CONTENT BASED IMAGE RETRIEVAL

By

Noor Naser Ahmed Al Aidy

Supervisor

Dr. Sami Serhan

Co-Supervisor

Dr. Moussa Habib

This Thesis was Submitted in Partial Fulfillment of the Requirements for the Master's Degree of Science in Computer Science

Faculty of Graduate Studies

The University of Jordan

November, 2004

This Thesis/Dissertation (Content Based Image Retrieval) was successfully defended and approved on 8-11-2004:

<u>Examination Committee</u> **<u>Signature</u>**

Dr. Sami Serhan, Supervisor
Assoc. Prof. of Computer Science -----------------------------------

Dr. Moussa Habib, Co-Supervisor
Assoc. Prof. of Computer Science -----------------------------------

Dr. Bassam Hasan Hammo, Member
Assistant Prof. of Computer Science -----------------------------------

Dr. Mohammed Sadeq Al-Rawi, Member
Assistant Prof. of Computer Science -----------------------------------

Prof. Saleh Al-Oqaili, External
Full Prof. of Computer Science
(*Al-Balqa Applied University*) -----------------------------------

Dedication

To My Father and My Mother

Acknowledgement

All praises to Allah who conveys me with the ability and the strength to make this thesis.

I would like to express my gratitude and sincerest appreciation to my supervisors Dr. Sami Serhan and Dr. Moussa Habib for their continuous support, advices, and for their patience in guiding me to the right approach for completing this work.

I would like to express my thanks and gratitude to the examination committee: Prof. Saleh Al-Oqaili, Dr. Bassam Hammo, and Dr. Mohammed Al-Rawi and to all my instructors at the University of Jordan.

TABLE OF CONTENTS

Subject	Page
Committee Decision	**ii**
Dedication	**iii**
Acknowledgment	**iv**
Table of Contents	**v**
List of Tables	**x**
List of Figures	**xii**
List of Abbreviations	**xiv**
Abstract	**xv**
Introduction	**1**
1 Problem Overview	1
2 System Outline	2
3 The General Structure of the Thesis	4
Literature Review	**5**
1 The Growth of Digital Imaging	5
2 History	6
3 What is CBIR?	8
4 CBIR User Categories	9
5 Professional groups making use of images	10
6 The Image Domain	12
7 Use and User: The Semantic Gap	13
8 Visual Feature Extraction	14
8.1 Color Feature	15
8.2 Texture Feature	16
8.3 Shape Feature	17
8.4 Color Layout	18
9 Segmentation: Grouping Data....	19
10 Query Formulation	20
11 Human in the Loop: Relevance Feedback	23
12 Image Retrieval Systems	24

CBIR Theoretical Approach **28**

1 Wavelet Transform 28

1.1 The Short Time Fourier Transform 29

1.2 Wavelet Analysis 30

1.3 Wavelet Properties 31

1.3.1 Orthogonal Wavelets 32

1.3.2 Quadrature Mirror Filter 32

1.3.3 A Band Pass Filter 32

1.4 The Scaling Function 33

1.5 Wavelet Transform in one Dimension 33

1.5.1 The Continuous Wavelet Transform 34

1.5.2 The Discrete Wavelet Transform 34

1.6 Wavelet Transform in Two Dimensions 35

2 Cross-Correlation 37

2.1 Cross-Correlation in One Dimension 38

2.2 Image Registration 38

2.3 Cross-Correlation in Two Dimensions 39

2.3.1 Un-Normalized Cross-Correlation in Two Dimensions 39

2.3.2 Normalized Cross-Correlation in Two Dimensions 39

3 Artificial Neural Networks 40

3.1 Perceptron and Backpropagation 42

4 Color Models 43

4.1 The RGB Color Model 43

4.2 The HSI Color Model 44

5 Gray scale Intensity Image 45

System Design and Implementation **46**

1 The Preprocessing Stage 46

1.1 Image rescaling (resizing) 47

1.2 Image Enhancement 48

1.2.1 Histogram Equalization 48

1.2.2 Multiplication of an image by a scalar 50

1.3 Image Partitioning (Division) 51

1.4 Converting an image from RGB to HSI color spaces and vice versa 53

1.5 Converting an RGB image to a Gray-scale image 54

2 Feature Extraction 55

2.1 Wavelet Decomposition 55

2.2 Coefficients Extraction 57

3 Similarity measure using Cross-Correlation 59

3.1 The Normalized Cross-Correlation on images 59

3.2 Cosine Theta Measure 60

3.2.1 The Reference Vector 61

3.2.2 The Use of Cosine theta 61

4 System Structure 62

4.1 Frequency Domain Models 63

4.1.1 Image Retrieval using cross correlation on approximation coefficients of Daubechies' Wavelets 63

4.1.1.1 A Variation to Model 1 67

4.1.2 Image Retrieval using cross correlation on approximation coefficients of Daubechies' Wavelets on the basis of sub-image matching 67

4.1.2.1 A Variation to Model 2 69

4.1.3 Image Retrieval using cross correlation on the reconstructed images on the basis of sub-image matching 69

4.2 Applying Cross-Correlation on Spatial Domain 70

4.2.1 Image Retrieval using cross correlation applied on spatial representation of sub-images 70

4.2.1.1 Variation I to Model 4 71

4.2.1.2 Variation II to Model 4 71

5 The Indexing Algorithm 72

Analysis of Results 74

1 Experiments' Details 74

2 Experimental Results 76

2.1 Hardware and Software 76

2.2 Database 76

2.3 The Proposed Models' Results 77

2.3.1 Frequency Domain Models 77

2.3.1.1 Image Retrieval using cross correlation on approximation coefficients of Daubechies' Wavelets 77

2.3.1.2 Image Retrieval using cross correlation on approximation coefficients of Daubechies' Wavelets on the basis of sub-image matching 85

2.3.1.3 Image Retrieval using cross correlation on the reconstructed images on the basis of sub-image matching 88

2.3.2 Spatial-Domain Models 88

2.3.2.1 Image Retrieval using cross correlation applied on spatial representation of sub-images 88

3 Discussion of Results 91

3.1 Frequency-Domain Models' Results 92

3.1.1 Image Retrieval using cross correlation on approximation coefficients of Daubechies' Wavelets Results 92

3.1.2 Image Retrieval using cross correlation on approximation coefficients of Daubechies' Wavelets on the basis of sub-image matching Results 95

3.1.3 Image Retrieval using cross correlation on the reconstructed images on the basis of sub-image matching Results 97

3.2 Spatial-Domain Models' Results 98

3.2.1 Image Retrieval using cross correlation applied on spatial representation of sub-images Results 98

3.3 The Enhancement of Retrieval time 102

Conclusions and Future Work 105

1 CBIR Models 105

1.1 Frequency-Domain Models 106

1.1.1 Image Retrieval using cross correlation on approximation coefficients of Daubechies' Wavelets …………………..… 106

1.1.2 Image Retrieval using cross correlation on approximation coefficients of Daubechies' Wavelets on the basis of sub-image matching …………………………………… 107

1.1.3 Image Retrieval using cross correlation on the reconstructed images on the basis of sub-image matching …………………………………...…………… 108

1.2 Spatial-Domain Models ……………………..………………….. 108

1.2.1 Image Retrieval using cross correlation applied on spatial representation of sub-images …………………...……... 108

1.3 The indexing algorithm ……………………………………...…... 109

2 Future Works ..………………………………..……………………. 110

References ……………………………………..……………………………. 111

Abstract (in Arabic Language) ……………………………………………. 115

LIST OF TABLES

Table Number	Table Title	Page
Table 1	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB db8A2mw'	78
Table 2	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB db8A3mw'	78
Table 3	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB sym8A2mw'	79
Table 4	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB sym8A3mw'	79
Table 5	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB db8A2sw'	80
Table 6	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB db8A3sw'	80
Table 7	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB sym8A2sw'	80
Table 8	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB sym8A3sw'	81
Table 9	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray db8A3s-sub'	86
Table 10	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray db8AD3s-sub'	86
Table 11	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray sym8A3s-sub'	87
Table 12	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray sym8AD3s-sub'	87
Table 13	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on model 3	88
Table 14	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB S64m'	89
Table 15	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB S32m'	89

Table Number	Table Title	Page
Table 16	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB S32s'	90
Table 17	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'HSI S32s'	90
Table 18	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray S64s'	91
Table 19	The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray S32s'	91
Table 20	Time (in seconds) taken to process user query on different models	102

LIST OF FIGURES

Figure Number	Figure Title	Page
Figure 1	Example queries for each of the six different query types and possible results	21
Figure 2	Short Time Fourier Transform basis functions and the coverage in time-frequency domain	29
Figure 3	Daubechies wavelet basis functions, time-frequency tiles, and coverage of the time-frequency plane	31
Figure 4	Image Decomposition Steps	37
Figure 5	General schematic of a feed-forward neural network	41
Figure 6	The RGB Color Cube	44
Figure 7	Preprocessing phase on an example image	48
Figure 8	The effect of histogram equalization	49
Figure 9	The Partitioning of an image to its five sub-images	52
Figure 10	Multi-scale structure in the wavelet transform of an image	57
Figure 11	The retrieval system using wavelet Coefficients	66
Figure 12	Target retrieved images in the top 20% returned images for a target query image using 'RGB db8A2mw'	82
Figure 13	Ranked images in the top 20% returned images of Category-search type using 'RGB db8A2mw'	83
Figure 14	Ranked images in the top 20% returned images using 'RGB db8A2mw'	84
Figure 15	The average number of hits for each category using 'RGB db8A3mw', 'RGB db8A3sw', and 'RGB sym8A3s'	94
Figure 16	Target retrieved images in the top 20% returned images from a target query image using 'RGB db8A3sw'	95
Figure 17	The average number of hits for each category using 'Gray db8A3s-sub', 'Gray db8AD3s-sub', 'Gray sym8A3s-sub', and 'Gray sym8AD3s-sub'	96
Figure 18	The average number of hits for each category using the third proposed model under the frequency domain category	98

Figure Number	Figure Title	Page
Figure 19	The average number of hits for each category using 'RGB S64m' and 'RGB S32m'	99
Figure 20	The average number of hits for each category using 'RGB S32s' and 'HSI S32s'	100
Figure 21	The average number of hits for each category using 'RGB S64m', 'RGB S32m', 'RGB S32s', and 'HSI S32s'	101
Figure 22	The average number of hits for each category using 'Gray S64s' and 'Gray S32s'	101
Figure 23	The ranked images in locations 8 to 16 using the indexing algorithm applied on 'RGB db8A3sw'	104

LIST OF ABBREVIATIONS

Abbreviation	**Meaning**
CBIR	Content Based Image Retrieval
DFT	Discrete Fourier Transform
FT	Fourier Transform
DWT	Discrete Wavelet Transform
CWT	Continuous Wavelet Transform
STFT	Short Time Fourier Transform
QBIC	Query By Image Content
ANN	Artificial Neural Networks
QMF	Quadrature Mirror Filter
TWT	Tree-structured Wavelet Transform
PWT	Pyramid-sructured Wavelet Transform
DWF	Discrete Wavelet Frame

Content Based Image Retrieval

By

Noor Naser Ahmed Al Aidy

Supervisor

Dr. Sami Serhan

Co-Supervisor

Dr. Moussa Habib

ABSTRACT

Recent years have witnessed a rapid increase of the size of digital image collections. Searching a digital library having large numbers of digital images or video sequences is in importance in this visual age. To make such a search practical, effective image coding and searching based on image semantics is becoming increasingly important.

In this thesis, four models were designed, implemented, and evaluated. These techniques are; image retrieval using cross correlation on approximation coefficients of Daubechies' wavelets, image retrieval using cross correlation on the coefficients of Daubechies' wavelets on the basis of sub-image matching, image retrieval using cross correlation on the reconstructed images on the basis of sub-image matching, and image retrieval using cross correlation applied on spatial representation of sub-images.

Such systems consist of three main stages: image preprocessing, image representation, and the matching process. In the proposed models, images are decomposed to a set of patches covering the entire image, as a step of the preprocessing stage. Three of the proposed models are working on images on their spatial-frequency representations by applying wavelet transform on the images. While the fourth model, is working on images on the spatial representations. To provide semantically meaningful image comparisons, the cross-correlation function is investigated in the matching process as the similarity measure. Several approaches for the proposed techniques were examined and compared. Moreover, to speed up the retrieval process, we proposed an indexing algorithm to be used to narrow the search process to the category that the query image belongs to.

The evaluation of the models is carried out using a general-purpose image collection. From the experiments, it is found that the proposed models are efficient in retrieving semantically similar images to the query image. Moreover, to illustrate the performance of the proposed models, the retrieval accuracies for target specific queries and category queries, on each of the proposed models, have been calculated which obtain promising results. The retrieval accuracy for the target specific queries was overwhelming beyond 95%. While for the category queries, the accuracy was totally dependant on the specific details of the images. The overall objective of such systems is to retrieve images that are semantically similar to the query image in the set of best matching images. We strongly believe that applying our models on a highly specialized database such as medical images database or video streams will achieve better results.

Introduction

Everyday, large numbers of people are using the Internet for searching and browsing through huge amount of information. However, they can not make use of the information unless it is organized so as to allow efficient browsing, searching, and retrieval. The goal of Content-based Image Retrieval (CBIR) is to search for images that are semantically similar to a query image given by a user.

1 Problem Overview

Searching for digital information, especially images or video sequences has become important in many areas such as hospitals, business, and government. The technology to access these images has accelerated phenomenally and at present surpasses our understanding of how users interact with visual information (Goodrum, 2000).

In a typical image retrieval application, the user has an image he or she is interested in and wants to find similar images from the entire database. Searching for images in a large library can be supported by the use of semantic information. However, the current state of computer vision does not allow semantic information to be easily and automatically extracted (Cox et al., 2000). In many current applications, the prevalent retrieval techniques involve the use of human supplied text annotation as the basis of the search process. Annotated text can describe some of the semantic content of the images. However, text-based search of annotated image databases has proved problematic for several reasons, including the user's unfamiliarity with specialized vocabulary and its restriction to a single language (Cox et al., 2000). As an alternative approach to the text-based annotation approach is the CBIR, by which images are indexed by their visual contents such as color, texture, shape, etc. Many research efforts have been made to extract these low-level image features, evaluate distance metrics, and look for efficient searching schemes (Tian et al., 2001).

2 System Outline

The operation of a CBIR system can be seen as a series of independent processing stages. As there exists multiple choices for each of these stages, a multitude of CBIR systems can be implemented by combining a set of common building blocks (Koskela et al., 2001). In this thesis, we developed four models to make semantically-meaningful comparisons of images efficient and accurate.

Many approaches are used to find better representations of images. Wavelet transform has received much attention as a tool for developing CBIR systems. In this thesis, four models for CBIR are designed and evaluated. The proposed models are categorized in spatial-domain and frequency domain models according to the form of images representations.

The first model based on applying wavelet transform on images to accurately encode semantic features of images. Image rescaling and histogram equalization techniques are used in the preprocessing stage. The approximation coefficients of the wavelet decomposition process are used as the images representations. In the matching stage, we use a multi-step metric to compute the distance between two given images. Cross-Correlation is applied on the images representations to measure how similar are the query image and a database image. As a result of this process, a matching vector is formed which corresponds to the tested database image. Next, cosine the angle between the so far formed matching vector and the prepared reference vector is computed. We called this technique Image retrieval using Cross-Correlation on approximation coefficients of Daubechies' Wavelets. The purpose of this model is to retrieve images that are semantically similar to the query image in the set of the best matching images.

In the second model, each image is decomposed to a set of sub-images, covering the entire image. This operation is applied as part of the preprocessing stage, in addition to the rescaling step. Then, wavelet transform is applied on each sub-image. This technique is called Image retrieval using Cross-Correlation on the coefficients of Daubechies' Wavelets on the basis of sub-image matching, because matching is applied on the basis of sub-images. The same processing steps are used like that in the previous technique.

In the third model, the same processing steps like the previous one are used. The main difference is in images representations. In this model, sub-images are decomposed using the wavelet analysis function. Then, they are reconstructed from only their second level of decomposition. The reconstructed images serve as the images representations. This model is called Image retrieval using Cross-Correlation on the reconstructed images on the basis of sub-image matching.

The last model differs from all the others in that it works on images on their spatial-domain representations. In this model, each image is, also, decomposed into a set of patches which cover the entire image. Preprocessing stage include the rescaling of images and the multiplication of the images by a scalar. Many approaches are tested including three representations of the images in the spatial-domain, which are, the RGB color space, the HSI color space, and the intensity representation of images. This model is called Image retrieval using Cross-Correlation applied on spatial representation of sub-images.

Finally, an indexing algorithm, to speed up the retrieval process, is proposed. When using it, only the category, in which the query image belongs to, is searched instead of the entire image collection.

3 The General Structure of the Thesis

In the Literature Review, a brief overview of Content-Based Image Retrieval and image processing is provided. The various visual features are presented. Finally, some related work in the field of CBIR is outlined.

In the CBIR Theoretical Approach, the wavelet transform techniques and the Cross-Correlation will be introduced. Moreover, a brief description of the RGB color model and the HSI color model is provided.

In the system Design and Implementation, the CBIR models design and implementations are presented. The preprocessing, images representations, and the matching stages are explained. The indexing algorithm is presented also.

In the Experiments and Analysis of Results, the proposed models are evaluated. Experiments and their results are introduced. Chapter five provides a conclusion for the thesis as well as some suggestions for the discussed approaches and ideas.

Literature Review

With the existence of the Internet, looking for information goes beyond physical libraries. Huge amounts of digital data are being generated everyday. Scanners convert the analog/physical data into digital form; digital cameras and camcorders directly generate digital data at the production phase (Rui et al., 1999). Because of these multimedia devices, information is currently in various media types, such as images, audio, video, graphics, in addition to the conventional text media type. Not only multimedia information is being generated at a rapid rate, it is also transmitted rapidly due to the expansion of the Internet. Experts say that the Internet is the largest library that ever existed; it is however also the most disorganized library ever (Rui et al., 1999).

The ability to access and manipulate remotely-stored images opens opportunities in many fields. Therefore, users in many professional fields are exploiting these opportunities in all kinds of new and exciting ways. While it is perfectly feasible to identify a desired image from a small collection simply by browsing, more efficient techniques are needed with collections containing thousands of items (Goodrum, 2000). The problems of image retrieval are becoming widely recognized, and the search for solutions an increasingly active area of research and development (Eakins and Graham, 1999).

1 The Growth of Digital Imaging

The twentieth century has witnessed a huge growth in the number of images in all fields of life. Images now are involved in diverse fields as medicine, journalism, advertising, design, education, and entertainment.

The real engine of imaging revolution has been the computer, which brings with it a range of techniques for digital image capture, processing, storage, and transmission. The involvement of computers in imaging can be dated back to 1965, with Ivan

Sutherland's sketchpad project, which demonstrated the feasibility of computerized creation, manipulation, and storage of images, though the high cost of hardware limited their use until the mid-1980s (Eakins and Graham, 1999). When the digital images become affordable, it is penetrated in many fields that depend essentially on images for communication, such as engineering, architecture, and medicine. Also, photograph libraries, art galleries, and museums saw the advantages of making their collections in an electronic form. The availability of the Internet in the early 1990s creates additional demands on the exploitation of digital images. According to a recent study (Lawrence and Giles, 1999), there are 180 million images on the publicly indexable Web, a total amount of image data of about 3Tb (terabytes), and an astounding one million or more digital images are being produced (Goodrum, 2000).

2 History

Image retrieval has been a very active research area since 1970's, with the thrust from two major research communities, database management and computer vision (Rui et al., 1999). These two communities study image retrieval from two different viewpoints; text-based and visual-based.

The text-based image retrieval can be traced back to the late 1970's. A very popular framework of image retrieval then was first annotate the images by text and then uses text-based database management system (DBMS) to perform image retrieval. That is, to assign keywords or textual descriptors to each image when it is first added to the collection, and use these terms as retrieval keys at search time. However, the process of manual indexing suffers from two major difficulties especially when the size of image collection is large (tens or hundreds of thousands). One is the amount of labor required for image annotation; labeling thousands of images is a cumbersome and expensive job to the degree that deployment of the economic balance behind the database is likely to decrease (Smeulders et al., 2000). To solve this problem, some systems like the one proposed by (Chang et al., 1997), use a program that explores the Internet, collects images, and insert them in a predefined taxonomy, depending on the text surrounding them.

The other difficulty, which is more essential, results from the rich content in the images and the subjectivity of human perception (Rui et al., 1999). Labeling is seldom complete, context sensitive, and, in any case, there is a significant fraction of requests whose semantics can't be captured by labeling alone (Armitage and Enser, 1997). The viewpoint of the cataloguer or indexer may be different from the perspective of the user, who him or herself may be an expert in the discipline (Eakins and Graham, 1999). An image may be perceived differently to different people, and may also mean different things to the same person at different times. For example, an image of people on a beach can signify "people," "a beach," "the ocean," "or pictures of San Diego" (Santini, 2001). A picture may not need so many words to describe it. A few numbers of words may be sufficient. However, the important issue is that those words vary from one person to another.

In the early 90's, with the huge growth of digital image collections, the two difficulties faced by text-based approach become more and more acute. To overcome these difficulties Content-Based Image Retrieval (CBIR) was proposed.

The earliest use of the term Content-Based Image Retrieval in the literature seems to have been by Kato in 1992, to describe his experiments into automatic retrieval of images from a database by color and shape features (Eakins and Graham, 1999).

The main idea behind Content-Based Image Retrieval is that instead of manually annotating images with keywords, images are indexed according to their contents, such as color, texture, shape…etc. Since then, many techniques in that research direction have been developed and many image retrieval systems, both research and commercial, have been built (Rui et al., 1999).

This approach has established a new framework of image retrieval from a new perspective. The advances in this research direction are mainly contributed by the computer vision community (Rui et al., 1999).

3 What is CBIR?

CBIR, standing for Content-Based Image Retrieval, is a term that is used to describe the process of retrieving desired images from a large collection on the basis of features (such as color, texture, and shape) that can be automatically extracted from the images themselves (Eakins and Graham, 1999). Retrieving of images on the basis of textual descriptors is definitely not a CBIR, even if these annotations describe image content.

CBIR differs from classical information retrieval in that image databases are essentially unstructured, since digitized images consist purely of arrays of pixel intensities, with no inherent meaning (Eakins and Graham, 1999).

Image databases differ fundamentally from text databases. In text databases, the raw material i.e., words stored as ASCII character strings, has already been logically structured by the author, while in image processing there is a need to extract useful information from the raw data before any reasoning is possible about the contents of the image.

Content-Based Image Retrieval inherits many of its methods from image processing and computer vision fields. It differs from these fields principally through its emphasis on the retrieval of images with desired characteristics from a collection of significant size (Eakins and Graham, 1999). On the other hand, image processing covers a wider field, including compression, image enhancement, transmission, and interpretation.

CBIR covers a range of topics that need research and development; some of the most important are (Eakins and Graham, 1999):

- Understanding image users' needs and information-seeking behavior.
- Identification of suitable ways of describing image content.
- Extracting such features from raw images.
- Providing compact storage for large image databases.
- Matching query and stored images in a way that reflects human similarity judgments.
- Efficiently accessing stored images by content.
- Providing usable human interfaces to CBIR systems.

4 CBIR User Categories

In the literature, there is a wide variety of Content-Based Image Retrieval systems and methods. Three broad categories of user aims are found in (Cox et al., 2000), when using the system:

1. *Target Specific search:* users are required to find a specific target image in the database; search termination is not possible with any other image, no matter how similar it is to the singular image sought (Cox et al., 2000). Target search may be for another image of the same object of which the user has an image (Smeulders et al., 2000). This is called *target search by example*. Target search may also be applied when the user has a specific image in his/her mind, and the target is searched interactively by looking if it is similar to a group of example images, for instance (Cox et al., 2000). These systems are suited to search for stamps, industrial components, and catalogues, in general (Smeulders et al., 2000). Target search

connects with the tradition of pattern matching in computer vision (Smeulders et al., 2000).

2. *Category search:* users search for images that belong to a prototypical category, e.g., "dogs", "skyscrapers", "kitchens", or "scenes of basketball games"; in some sense, when a user is asked to find an image that is adequately similar to a target image, the user embarks on a category search (Cox et al., 2000). Category search may be applied when the user has an example and the search is for other members of the same class. When the user has available a group of images and the search is for additional images of the same class, the search is called also category search.
3. *Open-Ended search / Browsing:* users of search by browsing have non-specific goal in mind when starting the search. Search by browsing often implies iterative refinement of the search, the similarity, or the examples within which the search was started (Smeulders et al., 2000). In a typical application, a user may start a search for a wallpaper geometric pattern with pastel colors, but the goal may changes several times during the search, as the user navigates through the database and is exposed to various options (Cox et al., 2000). Systems designed to meet user aims in this category are highly interactive, in which the result of the search can be manipulated interactively by *relevance feedback*.

5 Professional groups making use of images

The need to find desired images from a large collection is shared by many groups. Some of these groups use images in their jobs on daily basis, such as graphic designers, while others may never be required to use them, such as bank managers.

Between these two extremes, different groups of professionals make use of images such as medicine and law. Other groups of people may look for images not for themselves but for clients, such as librarians and museum curators. The following examples give a snapshot of the usefulness of CBIR in some fields:

- **Security and law enforcement:** The main application of content-based retrieval in this area is for retrieval of faces (Santini, 2001). The police use visual information to identify people or to record the scenes of crime for evidence; over the course of time; these photographic records become a valuable archive (Eakins and Graham, 1999). The photograph of the person will be digitized and linked with his textual record. For example, the metropolitan police force in London is involved with a project which is setting up an international database of the images of stolen objects (http://www.arttic.com/grasp/) (Eakins and Graham, 1999).
- **Medicine:** the medical and related health professions use and store visual information in the form of X-rays, ultrasound, or other scanned images, for diagnosis and monitoring purposes (Eakins and Graham, 1999). These images are kept with the patient's health record, which are stored most likely manually and organized using unique identifier. The role of content-based retrieval appears when images taken from many patients are collected in a database. The typical application is that of "clinical cases" databases, in which, given an image relative to a patient, the database retrieves the most similar images, so that the clinician can compare them together with the associated diagnoses (Santini, 2001). Keeping medical images digitized helps a lot in research and teaching purposes.
- **Architectural and Engineering design:** photographs are used in architecture to record finished projects, including interior and exterior shots of buildings as well particular features of the design (Eakins and Graham, 1999). Traditionally, these

photographs will be stored as hardcopy and organized by project number and name, and accessed by the architect to make presentation to the client or for teaching purposes. Larger architects' practices with more ample resources, have introduced digital cameras and the electronic storage of photographs. Because images are involved in many branches of engineering include plans, drawings, machine parts, many computer applications such as Computer Aided Design (CAD), are used in the design process. There is a need in many applications to make a use of the standard parts, in order to maintain competitive pricing. Hence, extensive design archive is existed in many engineering firms.

- **Historical research:** archaeologists rely heavily on images, since in some instances; visual record could be the only evidence. Where access to the original works of art is restricted or impossible, perhaps due to their geographic distance, ownership restrictions, or factors to do with their physical condition, researchers need to have an alternative form of photographs, slides or other pictures of the objects, which may be collected within a particular library, museum, or art gallery.

This is not the full of the story; these are examples of image use, and not a complete picture of the uses being made of visual information.

6 The Image Domain

When talking about image domain i.e., the set of images under consideration, there is a gradual distinction between narrow and broad domains.

At one end of the spectrum, there is the *narrow domain*. A narrow domain has a limited and predictable variability in all relevant aspects of its appearance (Smeulders et al., 2000). There is a limited variability in the contents of the images in the narrow domain. Also, when the object's appearance has limited variability, the semantic description of the image is generally well-defined and, by and large, unique (Smeulders et al., 2000). As an example of a narrow domain, is a set of frontal views of faces taken against a clear background. Although the visual details of each face differ largely from others, there are constraints governing the domain such as geometrical, physical, and color-related constraints. The domain will be wider if it will consider the faces photographed from a crowed or from outdoor scenes.

On the other end of the spectrum, there is the *broad domain*. A broad domain has an unlimited and unpredictable variability in its appearance even for the same semantic meaning (Smeulders et al., 2000). In broad domains, the semantics of the images are described only partially. The photo archives or large photo stocks are examples of broad domain. The broadest class of images available to date is the set of images on the Internet.

Image domain in many problems of practical interest lies in between these extreme ends of the spectrum. In a broad domain, the gap between the feature description and the semantic interpretation is generally wide (Smeulders et al., 2000). For narrow, specialized image domains, the gap between features and their semantic interpretation is usually smaller, so domain-specific models may help (Smeulders et al., 2000). Although there is a large field of applicability for restricted domain databases, there is a great interest in methods that can be applied to a large (theoretically unlimited) class of images (Santini, 2001).

7 Use and User: The Semantic Gap

The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation (Smeulders et al., 2000). In other words, when a user uses a content-based image retrieval system he/she seeks for semantic similarity, while the database can provide only similarity by data processing. This is because the database does not know what the user is looking for or what is the user thinking of the query. The use of keywords or captions as textual descriptors reduces content-based access to information retrieval. The common disadvantages of textual descriptors are the labor involved and the human subjectivity as discussed earlier.

Association of a complete semantic system to image data would entail at least solving the general object recognition problem from a single image (Smeulders et al., 2000). Since this problem is yet unsolved, research is focused on different methods to associate higher level semantics to data-driven observables (Smeulders et al., 2000).

The aim of content–based retrieval systems must be to provide maximum support in bridging the semantic gap between the simplicity of available visual features and the richness of the user semantics (Smeulders et al., 2000). But one does not expect the database to give the exact answer 100% of the time (Santini, 2001).

8 Visual Feature Extraction

In text-based retrieval systems, features can be keywords, phrases, or other textual descriptors. In contrast to text-based approach, features in visual-based systems are visual features such as color, texture, and shape. Visual features (content) extraction represents the basis of CBIR. Each image in the database is represented by a feature vector that reflects its characteristics according to the extracted visual feature and the selected representation for it. CBIR operates by retrieving stored images from a collection by comparing visual features that are automatically extracted from the images themselves.

Within the visual feature scope, features could be classified as global features and domain-specific features. The former include color, texture, and shape features, while the latter is application dependent. The domain-specific features are better covered in pattern recognition and may involve in much domain knowledge.

There does not exist a single best representation for a given feature. This is because the field is still under development and, more importantly, because of perception subjectivity; features are perceived differently by different people, and thus different representations cover most of preferences.

It is important to establish that content-based retrieval does not rely on describing the content of the image in its entirety (Smeulders et al., 2000). It may be sufficient that a retrieval system presents similar images, similar in some user-defined sense.

Some of the most common visual features that are used in image retrieval are described below:

8.1 Color Feature

One of the most widely used visual features in image retrieval is the color feature. It is relatively robust to background complication and independent of image size and orientation (Rui et al., 1999). Several color feature representations have been applied. The most common are color histogram, color moments, and color sets.

In image retrieval, color histogram is the most commonly used color feature representation (Rui et al., 1999). Color histogram shows the proportion of pixels within an image holding specific values (which are colors). The color histogram is computed for all images and stored in the database. When the user submits a query, the color histogram is computed for the query and compared with those stored in the database. To compute the similarity measure for the color histogram, histogram intersection can be used. The images with color histograms that are most closely match to the query color histogram are retrieved.

While histograms are useful because they are relatively insensitive to position and orientation changes, they do not capture spatial relationship of color regions and thus, they have limited discriminating power (Tian et al., 2001).

To overcome the drawbacks of color histogram, color moments approach was proposed as an alternative measure to color histogram. The mathematical foundation of this approach is that any probability distribution is uniquely characterized by its moments (Tian et al., 2001). Therefore, if the color distribution of an image is interpreted as a probability distribution, then it can be characterized uniquely by its moments. Furthermore, because most of the information is concentrated on the low-order moments, only the first moment (mean), the second and third central moments (variance and skewness) were computed as the color feature representation.

To facilitate fast search over large-scale image collections, (Smith and Chang, 1995) proposed color sets as an approximation to color histogram. They first transformed the (R, G, B) color space into a perceptually uniform space, such as HSI, and then quantized the transformed color space into M bins (Rui et al., 1999). A color set then, is the selection of the colors from that quantized color space.

8.2 Texture Feature

Texture refers to the visual patterns that have properties of homogeneity that do not result from the presence of only a single color or intensity (Rui et al., 1999). It is an innate property of virtually all surfaces, including clouds, trees, bricks, hair, fabric, etc (Rui et al., 1999). It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment (Haralick et al., 1973). The most popular texture representations are co-occurrence matrix, Tamura texture, and wavelet texture.

In the early 70's, (Haralick et al., 1973) proposed the co-occurrence matrix representation of texture feature. This approach constructed a co-occurrence matrix based on the orientation and distance between image pixels, and then extracts meaningful statistics from that matrix as the texture representation.

Tamura et al. (1978) explored the texture representation from a different angle. They developed computational approximation to six visual textual properties, which found, by psychological studies, to be important according to the human visual perception of texture. These six visual texture properties were coarseness, contrast, directionality, linelikeness, regularity, and roughness. One major distinction between the Tamura texture representation and the co-occurrence matrix representation is that all the texture properties in Tamura representation are visually meaningful whereas some of the texture properties used in co-occurrence matrix representation may not (for example, entropy) (Rui et al., 1999).

In early 90's, after wavelet transform was introduced, many researches began to study the effect of wavelet transform in texture representation (Rui et al., 1999). Many techniques have been built in which operations are done in time-frequency domain using the wavelet transform to obtain a multi-resolution analysis of the space.

8.3 Shape Feature

Queries for shapes are generally achieved by selecting an example image provided by the system or by having the user sketch a shape (Goodrum, 2000). In image retrieval, depending on the applications, some require the shape representation to be invariant to translation, rotation, and scaling; while others do not (Rui et al., 1999).

Shape representation is classified in general into boundary-based and region-based. In boundary-based, only the outer boundary is used as a representation of the shape, while the entire shape region is used in region-based shape representation.

Fourier descriptor and moment invariants are the most successful shape representation for the two classes, respectively. The idea behind Fourier descriptor is to use Fourier transformed boundary as shape feature. The main idea of moment invariants is to use region-based moments, which are invariants to transformations, as the shape feature (Rui et al., 1999).

8.4 Color Layout

Although the global color feature provides a discriminating power in image retrieval and is easy to calculate, using it in a large collection may retrieve many false positives, i.e. images, which have similar color composition but with a completely different content. Many results of researches proposed to use color layout; i.e. to combine color feature and spatial location, since it gives better results in image retrieval.

A natural approach to achieve this goal is to divide the image into sub-blocks and to compute color features from each of the blocks. Quad-tree based color layout is a variation of this approach. In quad-tree approach, the entire image is divided into quad-tree structure and each branch of the tree had its own color histogram. Although conceptually simple, this regular sub-block based approach can not provide accurate local information and is computation - and storage – expensive (Rui et al., 1999). Another approach is to segment the image into regions with salient color features by color set back projection, then to store color set together with the position to be used for later queries. The advantage of this approach is its accuracy while the disadvantage is the general difficult problem of reliable image segmentation (Rui et al., 1999).

Between these two extremes, several color layout representation were suggested. In the work done by (Pass et al., 1996), they classified each pixel of a certain color to either be coherent or incoherent, depending on whether it is a part of large similarity-colored region. By this approach, widely scattered pixels are distinguished from clustered pixels and thus improving the representation of local color features (Rui et al., 1999).

Along the same line of color layout feature, it's possible to construct the layout of texture and other visual features.

9 Segmentation: Grouping Data

Partitioning of the image aims at obtaining more selective features in a trade-off against having more information in features when no sub-division of the image is used at all (Smeulders et al., 2000). Shape feature and color layout feature depend on having good image segmentation. There are different levels of segmentation:

- If a user is interested in searching for an object, results will be better when doing complete object segmentation first. This form of segmentation is called *strong segmentation*, which means the division of image data into regions such that each region represents a complete object in the real world and nothing else. It should be noted immediately that object segmentation for broad domains of general images is not likely to succeed, with a possible exception for sophisticated techniques in very narrow domains (Smeulders et al., 2000).
- Because of the difficulty of achieving strong segmentation, researches tend to have *weak segmentation*. In weak segmentation, image data is grouped into regions that are internally homogeneous according to some criterion. Hopefully, each region is within the bounds of an object, but there is no guarantee that the region covers all objects' area.
- If the object has a nearly fixed shape, such as an eye, it is called *a sign*. Localizing signs is finding an object with a fixed shape and semantic meaning

(Smeulders et al., 2000). Signs are useful in CBIR, since they achieve unique and immediate semantic meaning.

- The last and weakest form of segmentation is *partitioning*, which means the division of the image data regardless of the data. A partition may be the entire image or we may do a conventional partitioning such as the central part of the image against the upper, right, left, and lower parts. The feasibility of fixed partitioning comes from the fact that images are created in accordance with certain canons or normative rules, such as placing the horizon about 2/3 up in the picture or keeping the main subject in the central area (Smeulders et al., 2000). Another way of partitioning is to split the image in tiles of equal size and to represent each tile with a set of features.

10 Query Formulation

Users must be able to express their needs easily and accurately in any retrieval system. Image retrieval is no exception to this, though it by no means obvious how this can be achieved in practice (Eakins and Graham, 1999).

A query falls in one of two categories: exact query; where the selected set to be closely related to the query satisfying a set of given criteria, and an approximate query, where the selected set of images is a ranking of the images that are most closely related to the query based on the used similarity measure.

Within each of the two categories, three subclasses can be defined depending on whether the query relates to the spatial content of the image, to the global image information, or to groups of images as depicted in Figure 1.

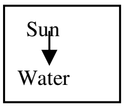
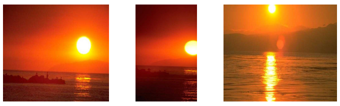
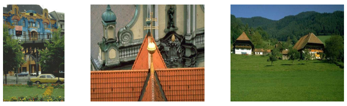
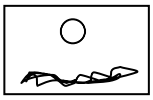
	Example query	Example query result
exact	Spatial predicate  	
	Image Predicate Amount of "sky" > 20% and amount of "sand" > 30%	
	Group predicate Location = "Germany"	
approximate	Spatial example 	...
	Image example 	...
	Group example positive negative	...

Figure 1 *Example queries for each of the six different query types and possible results*

For exact queries, the three subclasses are based on different predicates in which their results should satisfy:

- Exact query by spatial predicate: depends on the location of certain objects, homogeneous regions, or signs. To answer queries of this subclass, the system first must use an appropriate algorithm for segmentation, then to extract the domain dependent features. A web search system that allows the user to place icons that represent categories like human, sky, and water in the preferred spatial location is presented in (Lew and Sebe, 2000).
- Exact query by image predicate: specify the predicate on the basis of global image descriptions, often in the form of range predicates. Due to the semantic gap, range predicates on features are seldom used in a direct way (Smeulders et

al., 2000). One way to overcome this difficulty is to map ranges on color values into predefined predicates like "mostly blue" and "some yellow".

- Exact query by group predicate: to use this kind of queries, the database must be first partitioned into a set of categories. When using a query with an element that refers to a certain category, search happens only on that partition.

For approximate queries, the user specifies a single feature vector and it is anticipated that no image will satisfy the query exactly:

- Approximate query by spatial example: results in spatial structure that corresponds to literal image values and their spatial relationships. Pictorial specification of a spatial example requires a feature space such that feature values can be selected or sketched by the user (Smeulders et al., 2000). Low level feature selectors use color pickers or selections from shape and texture examples as in QBIC system (Flickner et al., 1995).
- Approximate query by image example: submits a complete array of pixels (image example) to the system and ask for the most similar images to the submitted query. This paradigm was first proposed by Chang and Fu in 1981 with their query by pictorial example (QPE) interface. Many systems followed this approach in designing the system's interface. If the target is to find an image of the same object as the example one or to find set of objects under different viewing conditions, then Query by image example is suited for such applications.
- Approximate image query by group example: feeds the system with a group of selected images which ensemble defines the goal. The user selects (m > 1) images from a palette of images presented to find images best matching the

common characteristics of the m images (Cox et al., 2000). This set of m-query is capable of determining the target more precisely.

11 Human in the Loop: Relevance Feedback

Early systems emphasize fully automated systems and focus on finding the single best feature. However, since few systems match the user needs from the first time round, recent research emphasis is given to interactive system and human in the loop or what is called *relevance feedback*.

Relevance feedback is the ability to automatically and iteratively refine the search in response to user indication of the retrieved images. It was developed originally for text retrieval and used recently for CBIR. In content-based retrieval, relevance feedback can be implemented e.g. by adjusting the weights of different textual terms when matching the query text with the documents of the database in a victorial form (Koskela et al., 2001). Other implementations include adding new terms or removing irrelevant ones in the query phrase, or modifying user profile.

The viewpoint of the system that adopted user interaction in CBIR is that any information obtainable from the user in the search process should be taken into consideration to provide the rich context required in reaching the optimal correspondence between the high level concepts people use and low-level features obtainable from the image. The correspondence between high level concepts and low level features is temporal and case specific. This means that, in general, every image query is different from others due to the hidden conceptions on the relevance of images and their mutual similarity (Koskela et al., 2001).

By involving the user in the retrieval process, the user can judge the relevance of the set of images displayed on the screen. For different search types; target, category, or open-ended search, different ways of user feedback have been considered. All are balancing between obtaining as much information from the user as possible and keeping the burden on the user minimal (Smeulders et al., 2000).

The simplest form of relevance feedback is to indicate the relevant images only as in (Cox et al., 2000). Another way requires the user to indicate explicitly both, relevant and irrelevant images.

If a search is dedicated for only a single image, then the method focus on target search. Bayesian framework (Cox et al., 2000) was proposed for target search. It was suggested to compute the likelihood of any image in the database being the target, given the history of user's relevance feedback actions. In each iteration, the user selects from the set of images displayed. Image pairs are formed by taking one selected image while the other is displayed but not selected image. The probability for an image to be the target is increased or decreased based on the similarity to the selected and non selected example in the pair.

Relevance feedback can be seen as a form of supervised learning to adjust the subsequent queries using the information gathered from the user's feedback (Koskela et al., 2001). This assists the system in next iterations of the retrieval process to closely converge the present needs of the user

Content-based image retrieval is only scalable to large data set when the database is able to anticipate what interactive queries will be made (Smeulders et al., 2000). A frequent assumption is that the image set, the features, and the similarity function are known in advance (Smeulders et al., 2000). When talking about a truly interactive system, these assumptions are no longer valid.

12 Image Retrieval Systems

Since the early 90's, image retrieval has become a very active research area. Recent studies have highlighted the fact that features like color, texture, shape, and spatial position indeed posses a very high semantic value and are effectively used in several CBIR systems (Ardizzoni et al., 1999). Many image retrieval systems, both commercial and research, have been proposed.

Queries can be formulated differently in different image retrieval systems. One way is by random browsing through the database one by one. Another way is to make a search based on text, in which the image is specified in terms of keywords. Yet another way is to provide an image or a sketch, in which a match is carried between database images' features and the features that are extracted from the query image.

IBM's QBIC system (Flickner et al., 1995), standing for Query By Image Content, is one of the first commercial image retrieval systems. QBIC supports queries based on example images, user-constructed sketches, and selected color and texture patterns from a sampler in which the percentage of the desired color in an image is adjusted by moving the sliders. When an image is added to the database, the system extracts and stores its color, shape, and texture features. QBIC allows text-based keyword search to be combined with content-based search.

Most systems are products of research, and therefore emphasize one aspect of content-based retrieval (Veltkamp and Tanase, 2002). Sometimes emphasize is on user interface, sometimes it is on forming feature vector, etc.

Color and texture features are used by many systems. However, this is not the case with the shape feature and layout. Shape feature is of little use. Moreover, layout is of less use than the shape feature. Using color feature for retrieval, results in images with similar colors. Retrieval on texture does not always yield images that have clearly the same texture, unless the database contains many images with a dominant texture (Veltkamp and Tanase, 2002).

It is difficult to evaluate how successful content-based image retrieval systems are, in terms of effectiveness, efficiency, and flexibility (Veltkamp and Tanase, 2002). Most of existing systems are good, but they are hard to verify. One reason is that the hyperlinks on the web are not active anymore, a design flaw of the web (Veltkamp and Tanase, 2002).

Another issue that must be considered is related to the retrieval systems. If the database contains only airplanes images, and the user asks for airplanes images, then the system will always satisfy the user needs. However, if the database is diverse and contains a single image of the airplanes, searching for more airplanes images will not satisfy the user needs. As the database grows, the chance to have images similar to the query image increases. Having a specific object in mind, looking for images with similar objects is a frustrating experience (Veltkamp and Tanase, 2002).

James Ze Wang et al. (1997) proposed a system called WBIIS, standing for Wavelet-Based Image Indexing and Searching. WBIIS system supports queries based on example images, user constructed sketches, as well as partial sketch images in which image contains non-specified regions are represented as black areas. The system allows queries to be performed on the basis of color layout which is encoded using Daubechies' Wavelet transform. However, the WBIIS system assesses the similarity between images by using whole pixel matrices, therefore operating in a very high dimensional space (Ardizzoni et al., 1999).

Stefania Ardizzoni et al. (1999) presented WINDSURF (Wavelet-Based Indexing of Images Using Region Fragmentation). The method uses Haar Wavelet transform to extract color and texture information from the image, and then a clustering algorithm is applied to partition the image onto homogeneous regions. Bhattacharyya distance is used to compare region descriptors, and then the results are combined at image level. The major limit of the presented approach is its low speed during the retrieval phase, since the regions of the database is sequentially scanned.

Q.Tian et al. (2001) developed a new technique to extract color and texture features in the locations given by salient points. The user is allowed to specify a query image by selecting an example image from the database. In order to extract features, a fixed number of salient points, according to their experiments, 50-100, for each image in the database are extracted using Haar Wavelet transform. The first three color moments for color and the Gabor moments for texture were extracted from the 3×3 and the 9×9 neighborhood of the salient points, respectively. The salient point approach proved to be robust to viewpoint change. It is also shown that the salient point approach works well in terms of the overall considerations of retrieval accuracy, computational complexity, and storage space of the feature vectors.

A recent algorithm was proposed by Mohammad F.A. Fauzi and Paul H. Lewis (2003) aimed to use a query image containing a single texture to retrieve images containing some area with similar texture to that in the query. The algorithm uses a multi-scale sub-image matching method. Three wavelet-based feature extraction methods are tested with the multi-scale technique, namely the pyramid-structured wavelet transform (PWT), the tree-structured wavelet transform (TWT), and the discrete wavelet frame (DWF). The study showed that the multi-scale sub-image matching method is an efficient way to achieve texture retrieval without any need for segmentation. The study showed that the retrieval accuracy of the algorithm is excellent and the computational load is also beneficial, as segmentation-based texture retrieval might take longer just to segment the image. It is also shown that the multi-scale TWT technique gives a slightly better result than the multi-scale PWT and multi-scale DWF.

CBIR Theoretical Approach

1 Wavelet Transform

Keeping signals in their raw format representation is not always the best representation for some signal processing related applications. In many cases, the information that can not be seen in the time domain can be seen in the frequency domain. Often times, the most discriminating information is hidden in the frequency representation of the signal. The frequency spectrum of a signal shows the frequencies that exist in that signal.

To obtain the frequency domain information, many mathematical transformations are applied. Localization, i.e., characterization of local properties, of a given basis functions in both time and frequency is the prime consideration when processing signals.

In our case, the signals we deal with are 2-D color images, in which the time domain is the spatial location of certain color pixels and the frequency domain is the color variation around a pixel. There is a variety of different transforms which can be used, but by far the most common one is the Fourier Transform (FT).

Fourier transform is a mathematical technique for transforming our view of the signal from time-based to frequency-based. It breaks down a signal into constituent sinusoids of different frequencies. A signal can be represented as a sum of sines and cosines according to the Fourier theory. FT is ideal for analyzing periodic signals, since Fourier expansions are periodic. However, the Fourier analysis has a serious drawback. The basis functions of the FT have good localization in frequency, but no localization in time. In other words, the FT tells us how much of each frequency exists in the signal, but it does not tell us when in time these frequency components exist. This drawback is not very important when the signal of interest is a stationary signal (periodic) i.e., the signal properties do not change much over time. However, for non stationary signals (signals that vary with time), if we want to know what spectral components occur at what time, then the FT is not the right transform to use.

To overcome this limitation, two mathematical models have been developed, the Short Time Fourier Transform (STFT) and the Wavelet Transform.

1.1 The Short Time Fourier Transform

The STFT analyzes the signal in both spatial and frequency domains simultaneously by encoding the signal through a scaled window related to both location and local frequency (Wang et al., 1997).

In STFT, the signal is divided into small segments, where each segment of the signal can be assumed stationary (Polikar, 2001). For this purpose, a time window function is proposed. Figure 2 shows a windowed FT where the window is a square wave (Graps, 1995). The square wave window truncates the sine or cosine function to fit the window of the selected width. Hence, the obtained information has a limited precision which is determined by the size of the selected window.

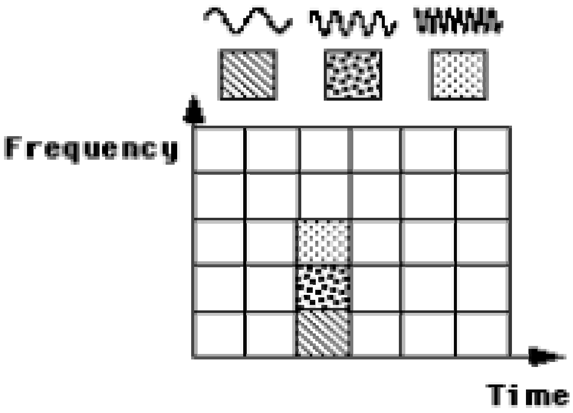


Figure 2 *Short Time Fourier Transform basis functions and the coverage in time-frequency domain*

However, the STFT has a drawback. Because a single window is used for all frequencies in the STFT, the resolution of the analysis is the same at all locations in the time frequency plane (Graps, 1995). Many signals require the ability to vary the window size in order to determine more accurately either time or frequency.

1.2 Wavelet Analysis

Wavelet analysis represents the next logical step; a windowing technique with variable-sized regions. They were invented to overcome the shortcomings of the FT. Wavelets are mathematical functions that cut up data into different frequency components, and then study each component with a resolution matched to its scale (Graps, 1995). They are capable of providing time and frequency information simultaneously. The fundamental idea is not only to analyze according to frequency, but also according to scale, which is not covered by the FT (Fauzi and Lewis, 2003). In addition, wavelet analysis provides the use of long time intervals where we want more precise low-frequency information, and shorter regions where we want high-frequency information. Therefore, local features can be described better with wavelets that have local extent.

Unlike the FT, whose basis functions are sinusoids, wavelet transforms are based on small waves, called wavelets, of varying frequency and limited duration (Gonzalez and Woods, 2002). Wavelets differ from the STFT in that the width of the window is changed as the transform is computed for every single spectral component.

Wavelets are basically based on the dilations and translations of one function that has special properties, this function is called the *mother wavelet*. The term mother implies that all functions, with different region of support, that are used in the transformation process are derived from one main function, which is the mother wavelet. Figure 3 shows the coverage in the time frequency domain plane with one wavelet function, the "Daubechies" wavelet (Graps, 1995).

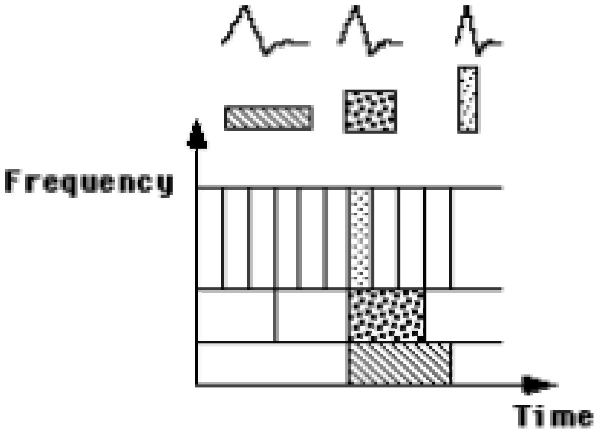


Figure 3 *Daubechies wavelet basis functions, time-frequency tiles, and coverage of the time-frequency plane*

The result of wavelet analysis is a number of time frequency representations of the signal with different resolutions; therefore it's called multi-resolution analysis. Approximations and details are used by the wavelet analysis to refer to the low frequency and high frequency contents of the signal, respectively. The approximations are the high scale, low frequency components of the signal and they carry the identity of the signal. On the other hand, details are the low scale, high frequency components of the signal and they carry the details information of the signal (Polikar, 2001).

One thing to remember is that the wavelet transform do not have a single set of basis functions like the FT, which utilizes just the sine and cosine functions (Graps, 1995). Instead, wavelets transform deals with the general properties of the wavelets; hence, infinite set of possible basis functions may be defined. Thus, wavelet analysis provides immediate access to information that can be obscured by other time-frequency methods such as Fourier analysis (Graps, 1995).

1.3 Wavelet Properties

The most important properties of wavelets are orthogonality, quadrature mirror filter, and filter bank.

1.3.1 Orthogonal Wavelets

Two functions *f* and *g* are said to be orthogonal to each other if their inner product is equal to zero:

$$\left\langle f(t), g(t) \right\rangle = \int_{a}^{b} f(t).g^{*}(t)\; dt = 0 \tag{3.1}$$

The mother wavelet and the father (scale) are orthogonal, since the inner product of wavelet coefficients and scale coefficients equals to zero.

1.3.2 Quadrature Mirror Filter (QMF)

Because of orthogonality, there exist a series of coefficients that can represent the decomposition of a signal. This set of coefficients is called the Quadrature Mirror Filter (QMF). When entering a signal *f(x)* to the QMF, two outputs will be generated. The first one is the output of the low pass filter, while the second one is the output of the high pass filter. The outputs of the QMF follow the orthogonality condition. In other words, what can be seen as output of the high pass filter can not be seen in the output of the low pass filter and vice versa.

1.3.3 A Band Pass Filter

Since signals have a finite spectrum, the translation of the wavelet will be limited by that duration. So, we have an upper boundary in the wavelets. However, we need a way to determine the number of scales needed to analyze the signals, that is, we need to get a lower bound for the translation of the wavelet (Valens, 2004). Looking at the wavelet as a band pass will solve the problem. In other words, the time compression of the wavelet by a factor of 2 will stretch the frequency spectrum of the wavelet by a factor of 2 and also shift all frequency components up by a factor of 2 (Valens, 2004). Every time the wavelet is stretched in the time domain with a factor of 2, its bandwidth is halved. If one wavelet can be seen as a band pass filter, then a series of dilated wavelets can be seen as a band pass filter bank.

1.4 The Scaling Function

The wavelet (mother) and scaling function (father) are the wavelet prototype functions required by the wavelet analysis. The mother wavelet produces the details of the wavelet decomposition; hence, it is determined by the high pass filter. While the father wavelet, which is also called the scaling function, produces the approximations of the wavelet decomposition, and so, it is determined by the low pass spectrum of the signal.

The width of the scaling function spectrum is an important parameter in the wavelet transform design. High scales correspond to a global information of the signal, they usually span the entire signal. While the small scales correspond to the detailed information. Moving from higher scale to smaller one is like zooming in.

To make the long story short, if we analyze a signal using the combination of scaling function and wavelets, the scaling function by itself takes care of the spectrum otherwise covered by all the wavelets up to the current scale, while the rest is done by the wavelets. The scaling function is defined as

$$\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k) \tag{3.2}$$

Here, k determines the location of $\varphi_{j,k}(x)$ along the x-axis, j determines the width of $\varphi_{j,k}(x)$, that is, how narrow or broad it is along the x-axis, and $2^{j/2}$ controls it's height. Since the shape of $\varphi_{j,k}(x)$ changes with j, $\varphi(x)$ is called a scaling function.

1.5 Wavelet Transform in one Dimension

In this section, the continuous wavelet transform and the discrete wavelet transform are defined.

1.5.1 The Continuous Wavelet Transform - CWT

To overcome the resolution problem of the STFT, the CWT was developed. The wavelet transform is calculated by continuously shifting a continuously scalable function over a signal and calculating the correlation between the two. The CWT of a continuous function f(x) relative to a wavelet $\psi(x)$ is defined formally as:

$$W_{\psi}(\mathrm{s},\tau) = \int_{-\infty}^{\infty} \mathrm{f}(\mathrm{x})\psi_{s,\tau}^{*}\ dx \tag{3.3}$$

Where * denotes complex conjugation. In this equation, the function f(x) is decomposed into a set of basis functions $\psi(x)$, called the wavelets. The variables s and τ are integers that scale and translate the function $\psi(x)$ to generate wavelets, such as a Duabechies wavelet family. These variables are the new dimensions after the wavelet transform. The function $\psi(x)$, is called the mother wavelet. The mother wavelet is defined as:

$$\psi_{s,\tau}(x) = \frac{1}{\sqrt{s}}\psi\left(\frac{x-\tau}{s}\right) \tag{3.4}$$

Where s is the scale factor and it indicates the wavelet's width, τ is the translation factor and it gives the position of the wavelet. The factor $\mathrm{s}^{-1/2}$ is for energy normalization across the different scales.

From the above equations, one can observe that the basis functions are not specified. Only a framework of the wavelets is defined within which one can design his preferred wavelets.

1.5.2 The Discrete Wavelet Transform - DWT

In many practical applications, the signal of interest is sampled. Therefore, a Discrete Wavelet Transform (DWT) is needed. It is important to remember that wavelets are not time discrete. Discrete wavelets are not continuously scalable and translatable but can only be scaled and translated in discrete steps (Valens, 2004). The DWT pair is defined as:

$$W_{\varphi}(j_0,k)=\frac{1}{\sqrt{M}}\sum_{x} f(x)\varphi_{j_0,k}(x) \tag{3.5}$$

$$W_{\psi}(j,k)=\frac{1}{\sqrt{M}}\sum_{x} f(x)\psi_{j,k}(x) \tag{3.6}$$

for j ≥ j_0 and

$$f(x)=\frac{1}{\sqrt{M}}\sum_{k} W_{\varphi}(j_0,k)\varphi_{j_0,k}(x)+\frac{1}{\sqrt{M}}\sum_{j=j_0}^{\infty}\sum_{k} W_{\psi}(j,k)\psi_{j,k}(x) \tag{3.7}$$

Where *f(x),*$\varphi_{j0,k}(x)$, and $\psi_{j,k}(x)$ are functions of the discrete variable $x = 0,1, 2, \ldots, M\text{-}1$. Normally, we let $j_0 = 0$ and select M to be a power of 2 (i.e., $M = 2^J$) so that the summations are performed over $x = 0, 1, 2, \ldots, M\text{-}1$, $j = 0, 1, 2, \ldots, J - 1$, and $k = 0, 1, 2, \ldots, 2^j - 1$ (Gonzalez and Woods, 2002). The coefficients defined in equation (3.5) are called approximation coefficients, and the coefficients defined in equation (3.6) are called detail coefficients.

1.6 Wavelet Transform in Two Dimensions

Thus far, we have discussed only one-dimensional data, which encompasses most ordinary signals. However, wavelet analysis can be applied to two-dimensional data like images. A two dimensional scaling function, $\varphi(x,y)$, and three two-dimensional wavelets, $\psi^H(x,y)$, $\psi^V(x,y)$, $\psi^D(x,y)$, are required as shown:

$$\varphi(x,y)=\varphi(x)\varphi(y) \tag{3.8}$$

$$\psi^{H}(x,y)=\psi(x)\varphi(y) \tag{3.9}$$

$$\psi^{V}(x,y)=\varphi(x)\psi(y) \tag{3.10}$$

$$\psi^{D}(x,y)=\psi(x)\psi(y) \tag{3.11}$$

$\psi^H(x,y)$, $\psi^V(x,y)$, *and* $\psi^D(x,y)$ are three wavelets that measure functional variations, that is, intensity or gray level variations for images, along different directions: ψ^H measures variations along columns (for example, horizontal edges), ψ^V responds to variations along rows (like vertical edges), and ψ^D corresponds to variations along diagonals (Gonzalez and Woods, 2002).

Extension of one-dimensional DWT to two dimensions is straight forward by using the above separable two-dimensional scaling and wavelet functions. The scaled and translated basis functions are defined as:

$$\varphi_{j,m,n}(x,y) = 2^{j/2}\varphi(2^j x - m, 2^j y - n), \tag{3.12}$$

$$\psi^i_{j,m,n}(x,y) = 2^{j/2}\psi^i(2^j x - m, 2^j y - n), \tag{3.13}$$

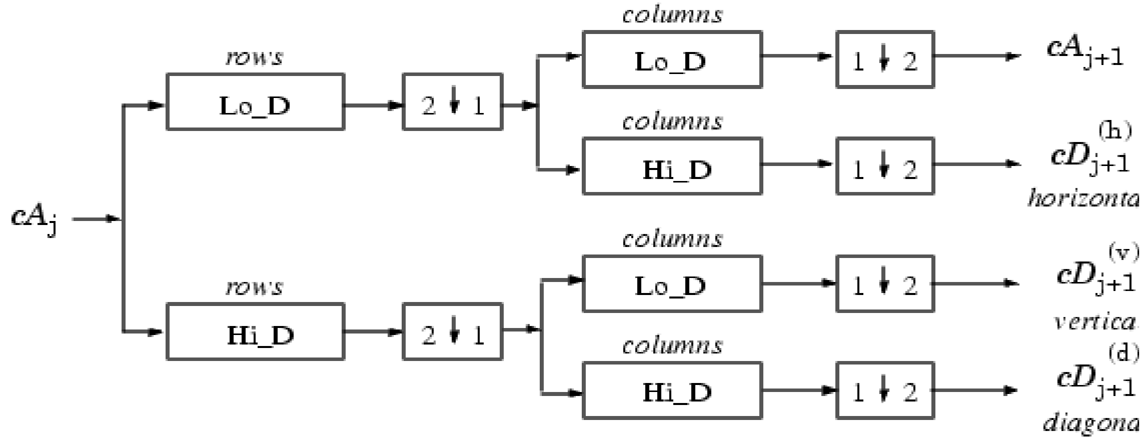
Where i is an index that identifies the directional wavelets in equations (3.9) to (3.11). Hence, i is a superscript that assumes the values H, V, and D. The DWT of a two dimensional function $f(x,y)$ of size M × N is then defined as

$$W_\varphi(j_0,m,n) = \frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{Y=0}^{N-1} f(x,y)\varphi_{j_0,m,n}(x,y) \tag{3.14}$$

$$W^i_\psi(j,m,n) = \frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\psi^i_{j,m,n}(x,y) \tag{3.15}$$

In away similar to the one-dimensional case, j_0 is an arbitrary starting scale, $W_\varphi(j_0,m,n)$ coefficients define an approximation to $f(x,y)$ at scale j_0, and the $W^i_\psi(j,m,n)$ coefficients add horizontal, vertical, and diagonal details for scales $j \geq j_0$. Normally, a selection is made to let $j_0 = 0$ and $N = M = 2^J$ so that j = 0, 1, 2, …, J-1 and m,n = 0, 1, 2, …, $2^j - 1$.

Given an image, the first step of decomposition produces four components: the approximation coefficients, and the details in three orientations (horizontal, vertical, and diagonal). The next step splits the approximation coefficients, cA1, in four components also, and so on. These vectors are obtained by convolving the approximation coefficients with the low-pass filter, Lo_D, for approximation, and with the high-pass filter, Hi_D, for detail, followed by dyadic decimation. Figure 4 describes the basic decomposition steps for images (Gonzalez, 2002).



Where

cAj: approximation coefficients at level j, and *cD*: wavelet coefficients

2↓1 Downsample columns: keep the even indexed columns.

1↑2 Downsample rows: keep the even indexed rows.

rows
X Convolve with filter X the rows of the entry.

X Convolve with filter X the columns of the entry**.**

Initialization **CA_0 =** s of the decomposition initialization.

Figure 4 *Image Decomposition Steps*

2 Cross-Correlation

In this section, Cross-Correlation in one dimension and in two dimensions are defined.

2.1 Cross-Correlation in One Dimension

Cross-Correlation is a standard method of estimating the degree to which two series are correlated (Bourke, 1996). The use of Cross-Correlation for template matching is motivated by the squared Euclidean distance measure (Lewis, 1995). If *x(i)* and *y(i)* are two series, where *i* = 0,1,2,…,N-1, then the Cross-Correlation series is defined as:

$$r(d) = \frac{\sum_i \left[(x(i) - \bar{x}) * (y(i-d) - \bar{y}) \right]}{\sqrt{\sum_i (x(i) - \bar{x})^2} \sqrt{\sum_i (y(i-d) - \bar{y})^2}} \qquad (3.16)$$

Where $\bar{x}$ and $\bar{y}$ are the means for the series *x(i)* and *y(i)*, respectively. The computed Cross-Correlation series *r(d)* is of twice the length as the original series.

In many signal processing applications the series is assumed to be circular in which case the out of range indexes are "wrapped" back within range, i.e., x(-1) = x(N-1), x(N+5) = x(5), etc (Bourke, 1996).

It is possible that the Cross-Correlation is used to test the correlation at short delays only. In this case, the range of delays *d*, and thus, the length of the Cross-Correlation series will be less than N. Therefore, the denominator of the previous formula is used to normalize the correlation coefficients to a maximum of a unit length, that is, -1 <= *r(d)* <= 1. The bounds of the output correlation coefficients mean the maximum correlation, while zero indicates no correlation. A high negative correlation value indicates a high correlation but of the inverse of one of the series (Bourke, 1996).

2.2 Image Registration

Image registration is the process of aligning two or more images of the same scene. In image registration, a comparison is done between one image that is considered as a reference, called the base image or template, and the other image, called the input image. One way to achieve image registration is by using Cross-Correlation.

2.3 Cross-Correlation in Two Dimensions

One way to identify a pattern within an image is to use Cross-Correlation of the image with a suitable mask. In this section, both, un-normalized and normalized Cross-Correlation are defined.

2.3.1 Un-Normalized Cross-Correlation in Two Dimensions

The form of un-normalized correlation coefficients is given by the following expression:

$$c(u,v) = \sum_{x,y} f(x,y) t(x-u, y-v) \qquad (3.17)$$

Where *f(x,y)* is the image and the sum is over *x,y* under the window containing the mask *t* positioned at *u,v*. However, there are several disadvantages to using the un-normalized formula for template matching. If the image energy $\sum f^2 (x,y)$ varies with position, matching using the above formula can fail (Lewis, 1995). Moreover, the range of *c(u,v)* is dependent on the size of the selected mask. To overcome these difficulties the normalized Cross-Correlation was proposed.

2.3.2 Normalized Cross-Correlation in Two Dimensions

In normalized Cross-Correlation the image and the used mask are normalized to unit length, yielding a cosine-like correlation coefficients (Lewis, 1995). Cross-Correlation in two dimensions is defined as;

$$\gamma(u,v) = \frac{\sum_{x,y}\left[f(x,y) - \bar{f}_{u,v}\right]\left[t(x-u, y-v) - \bar{t}\right]}{\left\{\sum_{x,y}\left[f(x,y) - \bar{f}_{u,v}\right]^2 \sum_{x,y}\left[t(x-u, y-v) - \bar{t}\right]^2\right\}^{0.5}} \qquad (3.18)$$

Where *f(x,y)* is the image, *t(x,y)* is the mask, $\bar{f}_{u,v}$ is the mean of *f(x,y)* in the region under the mask, and $\bar{t}$ is the mean of the mask. This formula is called normalized Cross-Correlation in two dimensions. If the pattern being sought and the mask are similar, the Cross-Correlation will be high. If we display the calculated normalized Cross-Correlation matrix as a surface plot, the peaks in the Cross-Correlation surface represent the positions of the best matches in the image of the mask. The mask is itself an image which needs to have the same functional appearance as the pattern to be found (Bourke, 1996).

It is important to note that the process can be extremely time consuming, the 2D Cross-Correlation function needs to be computed for every point in the image (Bourke, 1996). To calculate the Cross-Correlation function a N^2 operations are required.

To summary the above discussion, the main objective of using correlation is for matching. Suppose that *f(x,y)* is an image that contains objects or regions, and suppose we want to determine whether that image contains object or regions that we are searching for, we let a template image, *h(x,y)*, to be that object or regions. Then, if there is a match, the correlation of the two functions will be maximum at the location where *h* finds a correspondence in *f* (Gonzalez and Woods, 2002). Preprocessing, like scaling and alignment, is necessary in most practical applications, but the bulk of the process is performing the correlation (Gonzalez and Woods, 2002).

3 Artificial Neural Networks

The study of Artificial Neural Networks (ANNs) has been inspired in part by the observation that biological learning systems are built of very complex webs of interconnected neurons (Mitchell, 1997). In rough analogy, ANNs are built out of a densely interconnected set of simple units, where each unit takes a number of real-valued inputs (possibly the outputs of other units) and produces a single real-valued output (which may become the input to many other units) (Mitchell, 1997).

Neural networks consist of simple elements that are operating in parallel. Commonly, neural networks are adjusted or trained, so that association is established between a particular input and a specific target output. Many such input/output pairs are used, in this supervised learning, to train a network.

Neural networks have been involved in many applications to perform complex functions. These applications include pattern recognition, identification, classification, and vision.

One of the most commonly used network architectures is the multilayer feed-forward network. Figure 5 shows such a network. Feedforward networks often have one or more hidden layers of sigmoid neurons, which are non linear transfer functions, followed by an output layer of linear neurons.

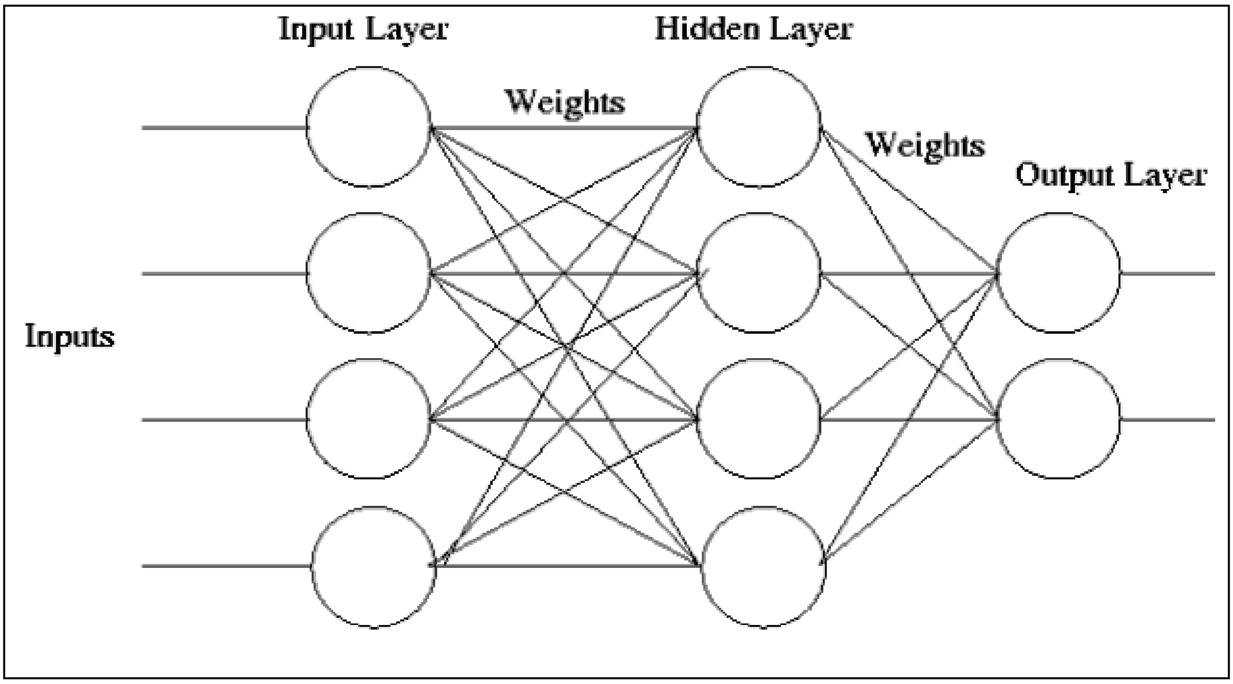


Figure 5 *General schematic of a feedforward neural network*

3.1 Perceptron and Backpropagation

The simplest neural network model is the perceptron model with a single neuron. A perceptron takes a vector of real-valued inputs; calculate a linear combination of these inputs, then outputs a 1 if the result is greater than some threshold and -1 otherwise (Mitchell, 1997). If the vectors are linearly separable, perceptrons trained adaptively will always find a solution in a finite time.

However, perceptrons have several limitations. The output values of a perceptron take only one of two values (1 or -1). Moreover, perceptrons can be used only to classify linearly separable set of vectors, while if the vectors are not linearly separable, learning will not reach a point in which all vectors are classified properly. For nonlinearly separable inputs, a multilayer network with a learning algorithm like *backpropagation* algorithm is suitable.

Multiple layers of neurons with non linear transfer functions in the hidden layer allow the network to learn non linear as well as linear relationships between input and output vectors. Moreover, the linear output layer allows the network to produce values outside the range -1 to +1.

The backpropagation is used to train a feedforward neural network. The backpropagation algorithm learns the weights for a multilayer network, given a network with a fixed set of units and interconnections (Mitchell, 1997). Since backpropagation is classified as a supervised learning algorithm, input vectors and target vectors are used to train a network, until it can associate input vectors with specific output vectors. Initially, weights are selected randomly, and then, at each iteration, a comparison is done between the network output values and the target values. If they differ, then weights are adjusted to minimize the squared error between the network output values and the target values for these outputs.

4 Color Models

Color models are three dimensional arrangements of color sensations (Sangwine and Horne, 1998). Each color is represented by a single point in these models. The purpose of color model is to facilitate the specification of colors in some standard, generally accepted way (Gonzalez and Woods, 2002). Two of the most popular color models in image processing community, the RGB and HSI color models, are defined in this section.

4.1 The RGB Color Model

The RGB, shorthand for Red Green Blue, model is the most commonly used color model for image processing. In the RGB model, each color appears in its primary spectral components of red, green, and blue (Gonzalez and Woods, 2002).

The color subspace of interest forms a cube as shown in figure 6. It is clear that the RGB values are at three corners; cyan, magenta, and yellow are at other three corners, black is at the origin, and white is at the corner that is farthest from the origin. All gray colors are placed on the line joining black and white points. Colors, in this color model, are represented by points and can be found on the surface or inside the cube. Representing each of the three primary color components by 8 bits yields a potential of 16 million colors.

An RGB image consists of three components images, one for each primary color. When fed into an RGB monitor, these three images combine on the phosphor screen to produce a composite color image (Gonzalez and Woods, 2002). Determining the color of each pixel, required a combination of the red, green, and blue intensities that are stored in each color plane at the pixel's location.

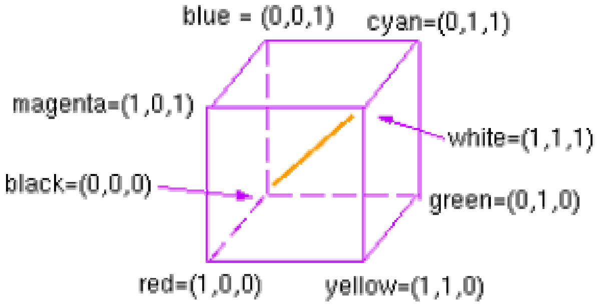


Figure 6 *The RGB Color Cube*

4.2 The HSI Color Model

Although RGB model matches nicely with the fact that human eye absorbs light with the greatest sensitivity in the blue, green, and red part of the spectrum, this model is not well suited for describing colors in ways that are practical for human interpretation. An inexperienced and not specially trained person can not give percentage estimates of blue, green, and red components. In the perception process, however, a human can easily recognize basic attributes of color; intensity (brightness, lightness), hue, and saturation (Sangwine and Horne, 1998). HSI, shorthand for Hue, Saturation, and Intensity, color model corresponds better to how people experience color than the RGB color space does.

The Hue is a color attribute that describes the pure color, such as pure yellow, orange, or red (Gonzalez and Woods, 2002). In other words, the hue represents the dominant wavelength in a mixture of light waves. Hue represents dominant color as perceived by an observer. Whereas saturation gives a measure of the degree to which a pure color (hue) is diluted by white light, hence, it is a measure of the purity of the color (Gonzalez and Woods, 2002). As the saturation values increase, the corresponding color (hues) vary from unsaturated (shades of gray) to fully saturated (no white components, and thus, pure spectrum colors). Intensity is roughly equivalent to brightness, as the intensity value increases, the corresponding colors become increasingly brighter.

However, brightness is a subjective descriptor that is practically impossible to measure. It embodies the achromatic (void of color) notion of intensity and is one of the key factors in describing color sensation (Gonzalez and Woods, 2002). Maximum intensity is sensed as pure white, while minimum intensity as pure black.

The HSI color model decouples the intensity component from the color-carrying information (hue and saturation) in a color image. As a result, the HSI model is an ideal tool for developing image processing algorithms based on color descriptions that are natural and intuitive to humans, who, after all, are the developers and users of these algorithms (Gonzalez and Woods, 2002).

5 Gray scale Intensity Image

If achromatic light (void of color) is considered, the only attribute of that color is its intensity, or amount. What viewers see on black and white television is achromatic light. Gray level refers to a scalar measure of intensity that ranges from black, to grays, and finally to white (Gonzalez and Woods, 2002).

An RGB colored image can be converted to a gray scale intensity image by using the standard perceptual weightings for the three color components, R, G, and B color components;

$$Gray\_\mathrm{Im}age = 0.2990 * R + 0.5870 * G + 0.1140 * B \tag{3.19}$$

Finally, intensity (gray level) is a most useful descriptor of monochromatic images (Gonzalez and Woods, 2002). The intensity is a measurable and easily interpretable quantity.

System Design and Implementation

Searching for digital information, especially images and video sequences has become important in this visual age. Among different features that are associated with CBIR, texture retrieval is one of the most difficult. Unlike color, texture can not be represented by a single pixel, making texture analysis a very challenging field (Fauzi and Lewis, 2003).

As discussed earlier, searching for desired images from a collection can be classified in three main categories: target-specific search, category search, and open-ended search. However, few systems take this classification into account when designing their systems. An implicit assumption made by (Cox et al., 2000) said that; systems which are optimized under a target testing condition also perform well in category searches and open-ended browsing.

Systems are implemented in MatLab, version 6.5, from scratch. Different functions from Image Processing, Wavelet, and Neural Network toolboxes are used. Such functions, like image decomposition, image resizing, and histogram equalization, are already built in MatLab and ready for use.

In this chapter, different techniques for CBIR, to retrieve similar images to a desired target image that is specified by the user, are designed and introduced. In addition, the preprocessing stage is described. Feature extraction stage, for models that are working by transforming images to the frequency domain, is presented. Moreover, the proposed CBIR techniques are defined. Finally, the indexing algorithm, which works with all the proposed models, is described.

1 The Preprocessing Stage

This section introduces the preprocessing techniques before the feature extraction stage. The preprocessing stage consists of different steps so that each model can use one or more of these steps. It's important to say that these steps are not necessarily to occur all together in a single model. We describe here the preprocessing steps for all models.

1.1 Image rescaling (resizing)

In the image dataset we had, images were large and of different sizes. The sizes we deal with are 678×435 and 435×678. Therefore, preprocessing step is needed to normalize the data. In this step, images were rescaled to common, relatively small sizes.

Bilinear interpolation is used for rescaling process. This method resamples the input image by overlaying the input image with a grid of number of points equal to the newly desired size of the image. Each point of the grid gives one pixel in the output image by calculating a weighted average of the input image pixels. At each grid point, sampling of the input image is needed to determine the pixel colors of the output image. An algorithm that shows the image rescaling is shown below.

```
Algorithm 4.1 Image rescaling
Input: I: Image Matrix; {input image}
Output: S: the rescaled image

Begin
S = image resize (I, [new_size], 'Bilinear Interpolation')
                    {Rescale the input image to a new size that is determined in the
            new_size variable using bilinear interpolation method}
End; {Algorithm}
```

An example of rescaling an image using bilinear interpolation is given in figure 7. This step is applied on images on all the proposed models.

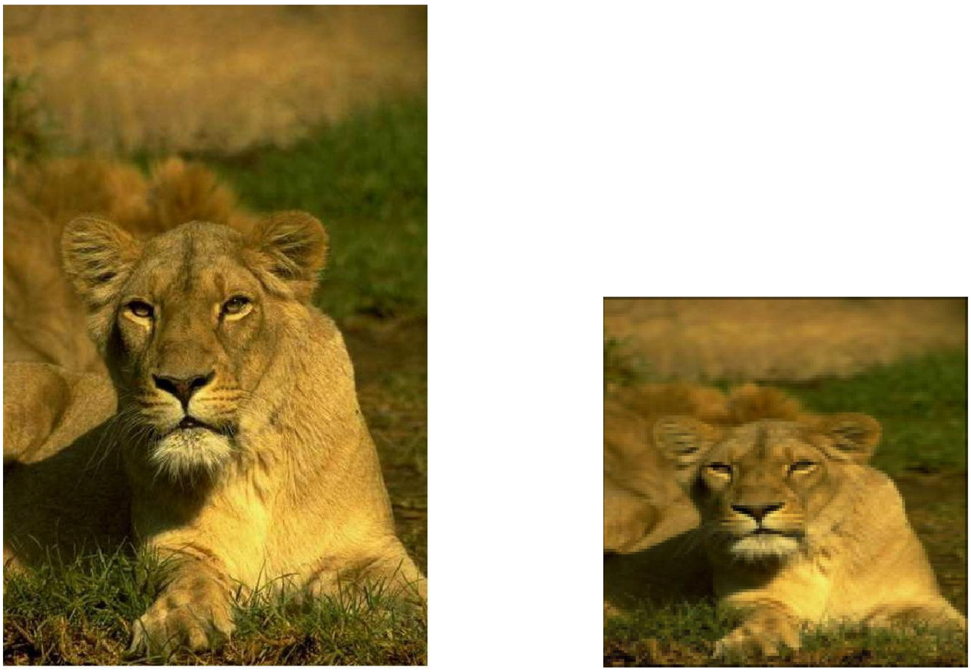


Figure 7 *Preprocessing phase on an example image: (a) original image (b) rescaled image using bilinear interpolation*

1.2 Image Enhancement

This step deals with enhancing the contrast of images. The principle objective of enhancement is to process an image so that the result is more suitable than the original image for a specific application (Gonzalez and Woods, 2002). Image enhancement can be done in frequency domain as well as in the spatial domain. Here, we apply image enhancement in the spatial domain in which images are directly manipulated. Enhancement of the contrast is carried out by histogram equalization and multiplication of images by a scalar.

1.2.1 Histogram Equalization

Histogram equalization is a gray scale manipulation method. The objective of histogram equalization is to spread the histogram of the input image so that the levels of the histogram-equalized image (output image) span a fuller range of the gray scale. As the dynamic range of gray levels is increased, the contrast range in an image is increased also. Figure 8 shows an example on the effects of histogram equalization.

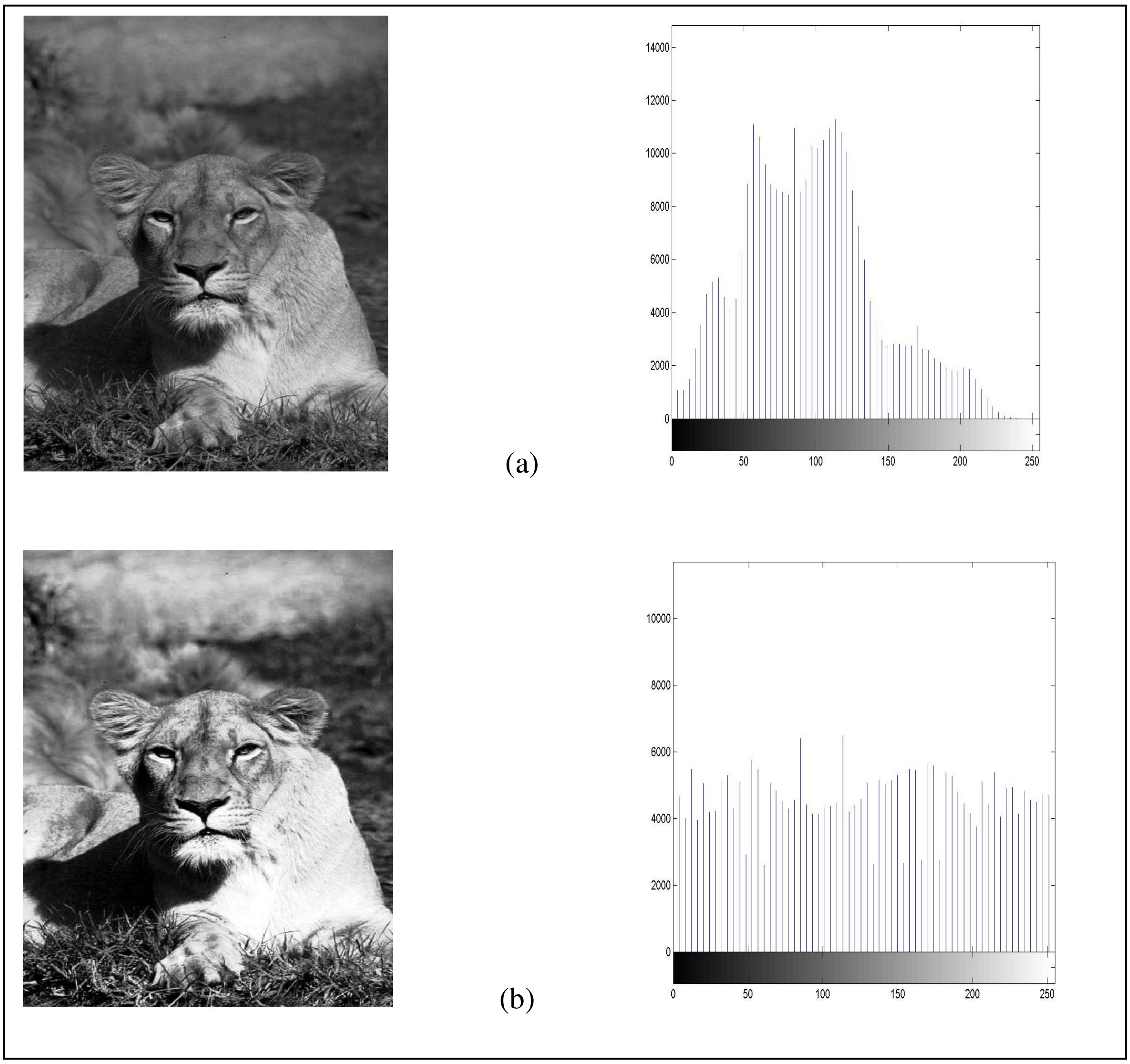


Figure 8 *The effect of histogram equalization (a) the original image and its histogram (b) the histogram equalized image and its histogram*

Algorithm 4.2(a) shows the histogram equalization process.

```
Algorithm 4.2(a) Histogram equalization
Input: I: the intensity (gray image) {it is the input image}
Output: E: the enhanced image

Begin
E= Histogram equalization (I);
                    {Apply histogram equalization on the input image I}
End; {Algorithm}
```

1.2.2 Multiplication of an image by a scalar

Another way to enhance the contrast in an image is by multiplying the whole input image by a scalar. By this way, we increase the differences between the pixels' values of the input image. We multiplied each of the three planes (R, G, and B) of the input image by a scalar, whose value is 255, to increase the differences in contrast between image pixels. Algorithm 4.2(b) shows the multiplication of an image by a scalar.

```
Algorithm 4.2(b) The multiplication of an image by a scalar
Input: S: the rescaled image {the output of algorithm 3.1}
Output: R, G, and B: the three enhanced planes of the image

Begin
R=255* S(:,:,1);
G=255*S(:,:,2);
B=255*S(:,:,3);
End; {Algorithm}
```

1.3 Image Partitioning (Division)

In some of the proposed techniques we applied image partitioning. The main purpose of image partitioning is to find locally representative features for different regions of an image. Image partitioning is used as an alternative of the global approach, in which features are obtained over the entire image, and thus, features do not take local properties into account. Moreover, by using image division, the probability of finding similar images to the query image is increased since regions are compared instead of the whole images. By using image partitioning approach, the computational speed is an advantage, since strong segmentation-based retrieval techniques could take longer time just to segment an image.

In our proposed models, sometimes we apply partitioning on the query image, sometimes we apply it on the database images, while in other times we apply it on both, query and database images.

Once an image is divided into its corresponding sub-images, a suitable feature extractor technique can be used then to find a suitable representation for each of the regions (sub-images).

Two ways of image partitioning are adopted in this thesis. In the first way, the image was partitioned into five sub-images. Each one of them is of size that is half of the size, in both dimensions, of the original image. This leads to segment the image into four partitions. The first partition lies on the upper left quarter of the original image, the second one lies on the upper right quarter of the image, the third one lies on the bottom left quarter, the fourth one lies on the bottom right quarter. Finally, the fifth one lies on the center of the image and it consists of the bottom right quarter of the first partition, bottom left quarter of the second partition, upper right quarter of the third partition, and the upper left quarter of the fourth partition, respectively. Figure 9 gives an example of image partitioning using the described algorithm.

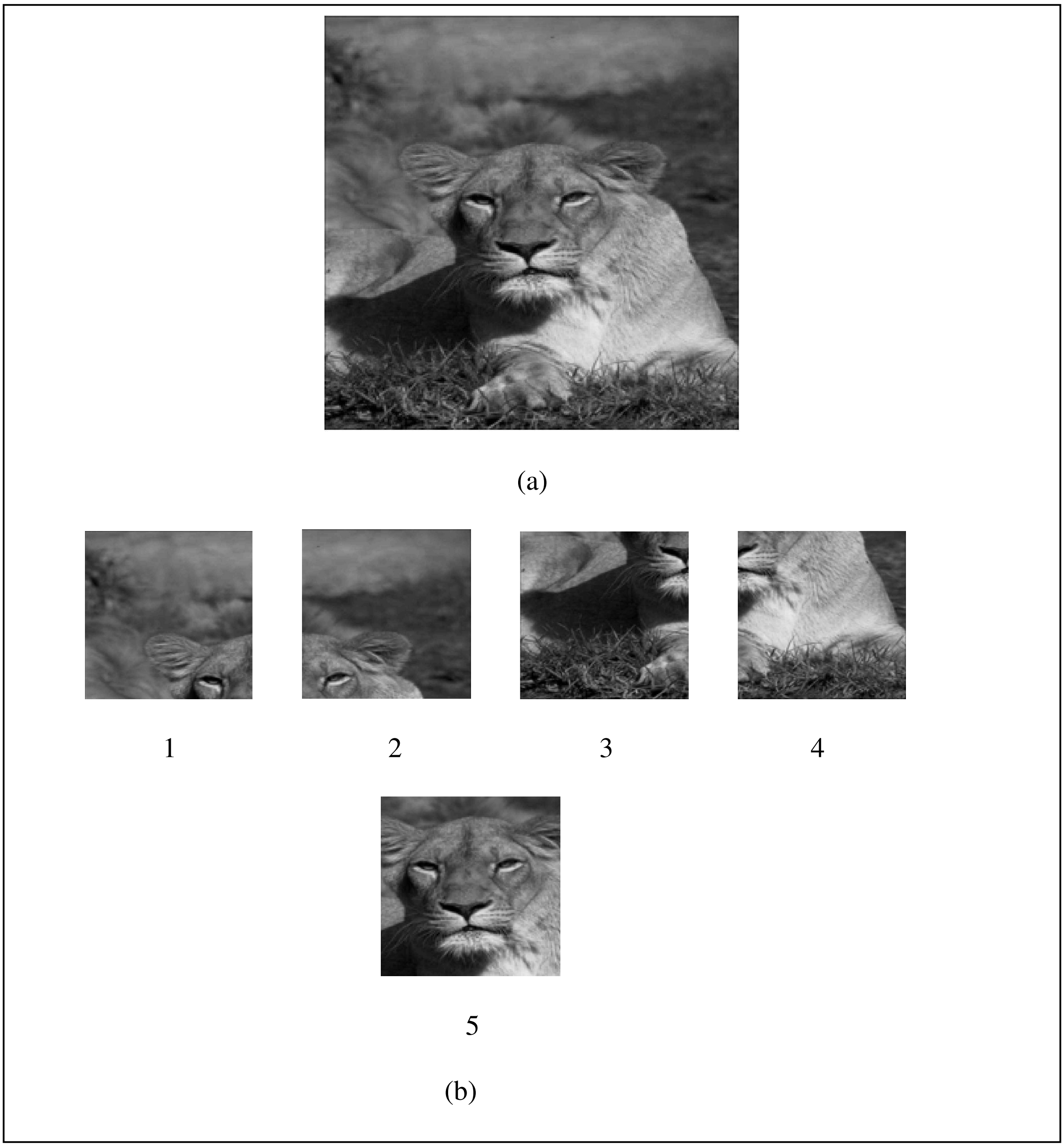


Figure 9 *The Partitioning of an image to its five sub-images. (a) The original image (b) The five partitions*

Algorithm 4.3 depicted the steps needed to achieve image partitioning. The second way of partitioning takes into account only the first four sub-images resulted from the first partitioning method.

```
Algorithm 4.3 The image partitioning (division)
Input: I: Image Matrix; {input image}
Output: sub_image1, sub_image2, sub_image3, sub_image4, and sub_image5: the five
output sub_images. These sub_images represent upper left quarter, upper right quarter,
bottom left quarter, bottom right quarter, and the center quarter of the image, respectively.

Begin
[x,y] = size(I);
                  {find the size of the input image, x is the number of rows and y is the
       number of columns}
factorx = x/2;
factory = y/2;
         {compute the factors by which the image will be divided}
sub_image1 = I(1:factorx,1:factory);
sub_image2 = I(1:factorx,factory+1: y);
sub_image3 = I(factorx+1:x,1:factory);
sub_image4 = I(factorx+1:x,factory+1:y);
sub_image5 = concatenate (
                 sub_image1 (factorx/2: factorx, factory/2: factory),
                 sub_image2 (factorx/2: factorx, 1: factory/2),
                 sub_image3 (1: factorx/2, factory/2: factory),
                 sub_image4 (1: factorx/2, 1: factory/2)
                                                                                    );
                  {Concatenate partitions of the sub_images that we found first}
End; {Algorithm}
```

1.4 Converting an image from RGB to HSI color spaces and vice versa

We test the HSI color space as a representative space for images in some of the proposed models. In more precise words, we convert the images to the HSI color space, then enhancement of the contrast of images using histogram equalization is applied, and then we convert them back to the RGB color space. Functions are available in MatLab to convert between these two spaces. These steps are described in Algorithm 4.4.

```
Algorithm 4.4 Converting an image from RGB to HSI color spaces and vice versa
Input: S: the rescaled image {the output of algorithm 3.1}
Output: E: the enhanced image in the RGB representation

Begin
HSI_Image = RGB-to-HSI (S);
                    {Convert image to the HSI color space}
HSI_Image((:,:,3) = histogram equalization (HSI_Image (:,:,3));
                    {apply histogram equalization on the Intensity matrix (V)}
E = HSI-to-RGB (HSI_Image);
End; {Algorithm}
```

1.5 Converting an RGB image to a Gray-scale image

In some cases, it would be sufficient to work with the intensity, rather than with the color information, of an image. We convert an RGB image to a gray-scaled one by using formula (4.20). As discussed earlier, R, G, and B are the Red, Green, and Blue components of the image.

$$Gray\_\mathrm{Im}age = 0.2990 * R + 0.5870 * G + 0.1140 * B \qquad (4.20)$$

Algorithm 4.5 shows the conversion of an image to a gray-scale one using the above expression.

```
Algorithm 4.5 Conversion of an image to a gray-scale
Input: I: Image Matrix; {input image}
Output: Gray_Image:Gray_scale image

Begin
R = I(:,:,1);
G = I(:,:,1);
B = I(:,:,1);
Gray_Image = 0.2990*double(R) + 0.5870*double(G) + 0.1140*double(B);
End; {Algorithm}
```

2 Feature Extraction

Once images are preprocessed, the next step is to find the suitable form to be used for images representations. In three proposed models, images are transformed to the frequency domain using the wavelet decomposition process. Images are represented using the wavelet coefficients resulted from the decomposition process.

In this section, the process of extracting images representations in the frequency domain is discussed.

2.1 Wavelet Decomposition

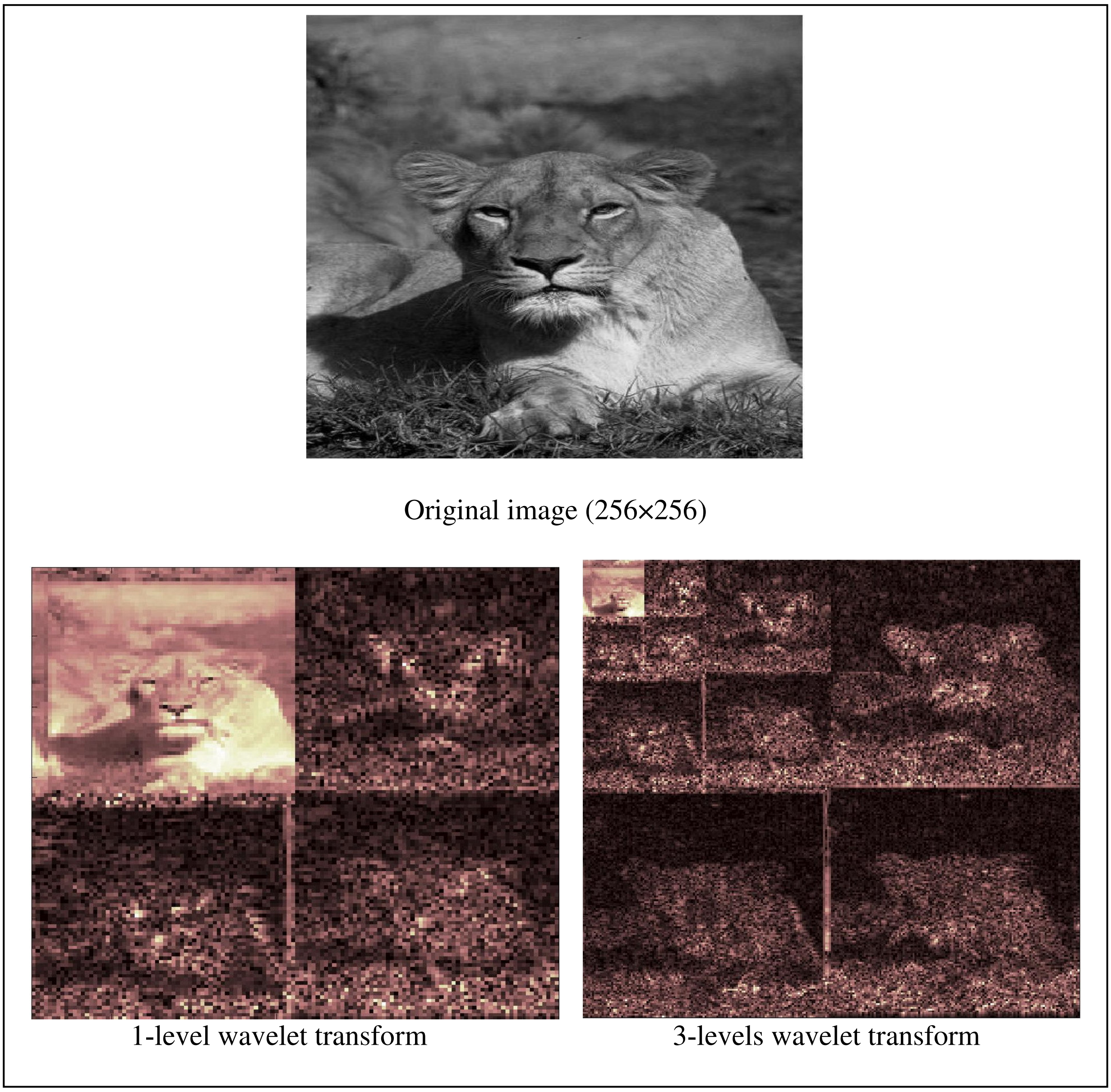
We used multi-scale 2-Dimensional wavelet decomposition function to decompose an image. This function returns the wavelet decomposition of the image at scale N, using the specified wavelet name.

Because the set of the wavelets is an infinity set, different wavelets may give different performance for different types of image (Wang et al., 1997). One should take benefits of this characteristic when designing an image retrieval system. We used a "Daubechies-8" and "Symmlet-8" wavelet for the wavelet decomposition process to match the characteristics of the 2-D signal (image) we are analyzing. They were chosen because they are compactly supported and well localized in both the time and frequency space. "Symmlets" were designed by Daubechies to be orthogonal, smooth, nearly symmetric, and non-zero on a relatively short interval (compact support) (Wang et al., 1997).

Wavelet subclasses can be classified by the number of vanishing moments which determine what does or does not the specified wavelet subclass represent. The number of vanishing moments for the subclass we considered is 8 for "Symmlet" and "Daubechies" wavelets. This means that our wavelet will ignore linear through eighth degree functions. An example of wavelet decomposition process is shown in figure10.

As discussed previously, the decomposition of an image at each scale gives four matrices. The upper left matrix is the lower frequency bands of the wavelet transform and gives the approximation coefficients. The other three matrices are the higher frequency bands of the signal which are arranged as: the upper right is the horizontal details, the lower left is the vertical details, and the lower right is the diagonal details coefficients, respectively.

Wavelets perform better than traditional layout coding because the coefficients in wavelet-created compression data actually contain sufficient information to reconstruct the original image at a lower loss rate using an inverse wavelet transform (Wang et al., 1997).



Original image (256×256)

1-level wavelet transform

3-levels wavelet transform

Figure 10 *Multi-scale structure in the wavelet transform of an image. "Daubechies-8" wavelet is used for this transform*

2.2 Coefficients Extraction

Once the wavelet transform is applied on the images, we want to extract coefficients that will be used as the images representations in the techniques that test images in their frequency-domain representation. The two types of coefficients are examined in our models; the approximation coefficients and the detail coefficients. The lower frequency bands, and hence the approximation coefficients, in the wavelet transform usually represent object configurations in the images while the higher frequency bands (the detail coefficients) represent texture and local color variation (Wang et al., 1997).

Sometimes we extract the approximation coefficients of the last level (scale) of decomposition. Another way is to extract horizontal, vertical, and diagonal detail coefficients of the last level of decomposition, in addition to the approximation coefficients. Algorithm 4.6 shows the decomposition of an image using wavelet transform and the extraction of coefficients.

Algorithm 4.6 The decomposition of an image using wavelet transform and the extraction of coefficients

Input: P: the preprocessed Image Matrix; {the preprocessed image that resulted from section1}

Output: A, H, V, and D: the wavelet coefficients of the last level of decomposition {A: the approximation coefficients. H, V, and D: the horizontal, vertical, and diagonal detail coefficients, respectively}

```
Begin
Determine wavelet name wname   {db8 (for "Daubechies-8") or sym8 (for "Symmlet-8")
Determine the number of levels of decomposition {N}
[C,S] = wavelet-decomposition (P, N, wname);
                         {Apply wavelet decomposition of the image at level N by
                 using wavelet name db8 or sym8; to produce Coefficients vector C
                 and Coefficients vectors size S}
Determine the desired level to extract features from it {L}
A = approximation coefficient(C, S, wname, L);
                         {extract approximation coefficients at level L from the wavelet
                 structure [C,S]}
H = detail coefficient ('Horizontal ', C, S, L);
H = detail coefficient ('Horizontal ', C, S, L);
V = detail coefficient ('Vertical' , C, S, L);
```

```
D = detail coefficient ('Diagonal' , C, S, L);
                              {Extract the horizontal, vertical, and diagonal detail coefficients,
                    (H, V, D), from the wavelet decomposition structure [C,S]}

End; {Algorithm}
```

3 Similarity Measure using Cross-Correlation

In all the models we proposed, Cross-Correlation is used as a similarity measure to indicate how much two images (or two sub-images) are similar. In this section, the use of normalized Cross-Correlation as a similarity measure and the computation of the final distance of a database image to the query image are discussed.

3.1 The Normalized Cross-Correlation on Images

The normalized Cross-Correlation of two images or two sub-images (regions) is computed using a built-in function in MatLab. This function computes the normalized Cross-Correlation of the two matrices, Template (the pattern we search for) and A (the searched image). The resulting matrix contains the correlation coefficients, which may range in value from -1.0 to 1.0.

In our experiments, we take the highest 20 values from the resulted Cross-Correlation matrix, to give us an indication of the degree of similarity. This is because images may not be 100% identical; instead, they may be similar to a high degree. The obtained values form a vector called the matching vector (MV). Algorithm 4.7 depicted the use of Cross-Correlation.

Algorithm 4.7 The use of Cross-Correlation

Input: Template, Input: the preprocessed Images (or sub-images) matrices; {the preprocessed images that resulted from section1} if the spatial representation is considered. If the frequency domain is considered then they are the wavelet decomposition coefficients that are resulted from Algorithm 3.6. *Template* is the pattern or image (sub-image) we are searching for, while *Input* is the tested image.

Output: MV: the Matching Vector

```
Begin
CC = Normalized_Cross_Correlation(Template, Input);
                {apply normalized Cross-Correlation on Template and Input, such that
     Template is smaller than or equal to the size of the Input. CC is the Correlation
     Coefficients matrix}
Sorted-CC = the sorted CC matrix in descending order.
MV = the first 20 values of the Sorted-CC matrix
End; {Algorithm}
```

3.2 Cosine Theta Measure cos(θ)

As a result of applying Cross-Correlation, a vector containing 20 values is formed. If an image is divided into partitions (sub-images), then each region has its own similarity vector.

To rank images according to their similarity to an image that is specified by the user, a scalar value, rather than a vector, is needed to be associated with each image. Therefore, we use cosine theta measure for that purpose. We assume that, each vector resulted from the Cross-Correlation function is a vector in our space.

However, a reference vector is needed to measure cosine the angle between that reference vector and the input vector. In this section, the reference vector and the use of cosine theta measure are explained

3.2.1 The Reference vector

Since our goal is to measure the distance between the query image and all the input images, the logical action is to build the reference vector from the query image or from its regions. The reference vector is formed by applying the Cross-Correlation function on the query image itself. That is, to let the query image to be the template as well as the input image, then to apply the Cross-Correlation function. This is of course if the query image is not divided into regions. However, if it is divided, this process has to be repeated for each region. In other words, each region of the query image has its associated reference vector that will be used to compute the distance to the corresponding regions of the input images as discussed in the next section. Reference vectors are formed from the highest 20 values of the resulted Cross-Correlation matrix.

3.2.2 The Use of Cosine theta

Now, we are ready to compute cosine the angle between the so far formed reference vector and the computed one for the input image in the 20 dimensional feature vector space. The cosine value of the angle can be obtained by computing vector dot product in a normalized vector space. This alternative measure, (alternative to the Euclidean distance) reduces the sensitivity to color or brightness shift (Wang et al., 1997).

Cosine the angle between the Reference vector and the Matching vector is computed using the following formula:

$$Cos(\theta) = \frac{\mathrm{Re}\,ference\_Vector \bullet Input\_Vector}{|\,\mathrm{Re}\,ference\_Vector\,| \quad |\,Input\_Vector\,|} \qquad (4.21)$$

If Reference_Vector = <a_1, a_2, a_3, ..., a_{20}> and Matching_Vector = <b_1, b_2, b_3, ..., b_{20}>, then

$$|\,\mathrm{Re}\,ference\_Vector\,| = \sqrt{a_1^{\,2} + a_2^2 + a_3^2 + \ldots + a_{20}^2} \qquad (4.22)$$

$$|\,Matching\_Vector\,| = \sqrt{b_1^{\,2} + b_2^2 + b_3^2 + \ldots + b_{20}^2} \qquad (4.23)$$

The values obtained here are between 0 and 1. As the output of this function becomes closer to zero, this means the corresponding image has a high degree of similarity to the query image, and hence, it will be one of the candidate images to be displayed first. Algorithm 4.8 explains the use of cosine theta measure.

Algorithm 4.8 The use of Cosine Theta Measure *cos(θ)*

Input: MV: the Matching Vector {the output of Algorithm 4.7}. MV = <b_1, b_2, b_3, ..., b_{20}>

RV: the Reference Vector {formed as discussed in section 3.2.1}

RV = = <a_1, a_2, a_3, ..., a_{20}>

Output: D: the scalar distance of the input image to the query image.

```
Begin

Product = Dot-Product(RV, MV);
                {apply dot product between the reference vector and the matching vector}
RV_Value = Square_Root($(a_1)^2 + (a_2)^2 + (a_3)^2 + ... + (a_{20})^2$);
MV_Value = Square_Root($(b_1)^2 + (b_2)^2 + (b_3)^2 + ... + (b_{20})^2$);
D = Product / (RV_Value × MV_Value);
Associate D with the tested image for the later ranking
End; {Algorithm}
```

4 System Structure

In this thesis, four models for CBIR, to retrieve images from a general-purpose image collection based on their similarity to the target image, are proposed. Three models based on transforming images to the frequency domain, while the last one works by keeping images in their spatial representation. Therefore, the proposed models are categorized into frequency domain models and spatial domain models. In the next sections, we present a full description of the proposed models.

4.1 Frequency Domain Models

Models under this class work by transforming images to the frequency domain using discrete wavelet transform. Descriptions of the proposed models are available in this section, in addition to the variations of these models.

4.1.1 Image Retrieval using Cross-Correlation on approximation coefficients of Daubechies' Wavelets

This model consists of three phases. The first phase is the preprocessing phase. In it, the image is rescaled to a new size of 128×128 pixels using bilinear interpolation as discussed earlier. Then, the image is converted to the HSI color space. A histogram equalization then is applied on the intensity matrix (V) to enhance the contrast of the image, we call the new intensity image V'. As a last step of this phase, the image is converted back to the RGB color space using H, S, and the new histogram equalized image (V').

The second phase is to find the image representation. We compute a 3-layer 2-D Wavelet transform on each of the three matrices of the image (R, G, and B) using Daubechies' Wavelets. Denote the three matrices obtained from the transform as $W_{PR}(1:128,1:128)$, $W_{PG}(1:128,1:128)$, and $W_{PB}(1:128,1:128)$, where PR, PG, and PB are Red Plane, Green Plane, and Blue Plane, respectively. Then, the upper left 16×16 corner of each transform matrix, $W_{Pi}(1:16,1:16)$, represents the lowest frequency band of the 2-D image in a particular color component for the level of wavelet transform we used, as discussed previously, it is called the approximation coefficients. The lower frequency bands in the wavelet transform usually represent object configuration. We take these three sub-matrices as the image representation.

Until this point, we have the approximation matrix of the third level wavelet decomposition as our image representation. This representation is computed for each image in the dataset as well as for the query image.

Now, we are ready for the Matching phase. We apply the Cross-Correlation function between the matrix obtained for an image in the dataset and the one obtained for the query on the corresponding planes. This process is repeated for all images in the dataset.

In order to cast the matching issue in more precise terms, we take the Red plane for a database image and the Red plane for the query image. This is of course after applying the discussed preprocessing operations for this model. We apply 3-layer 2-D Wavelet transform on both sides, the Red plane of the query and the database image. Then, we take $W_R(1:16,1:16)$ from both sides. Cross-Correlation, now, is applied on those matrices. This process is carried on for the Green and Blue Planes also. The output is three Cross-Correlation matrices for each database image, one for each plane.

As a result, a matching vector is formed from the highest 20 values of the obtained Cross-Correlation matrix. In other terms, for each image correlated with the query, three matching vectors are obtained, one for each plane MV_R, MV_G, and MV_B.

In the next step, the reference vectors are formed. As discussed earlier, they formed from the query image. Since the query image is represented by three color planes, three reference vectors are computed, one for each plane.

The reference vector is computed by taking the approximation matrix of the third level of decomposition of the query on each plane, W_{Pi} , and correlate it with itself. That is, we apply Cross-Correlation between $W_{PR}(1:16,1:16)$ and itself, this is become reference vector of the Red plane, RV_R, cross correlate $W_{PG}(1:16,1:16)$ with itself, this is become reference vector of the Green plane, RV_G, and finally, cross correlate $W_{PB}(1:16,1:16)$ with itself, this is become reference vector of the Blue plane, RV_B.

Now, we compute cosine the angle between a reference vector, RV_i , and a matching vector MV_K, for an input image, for sure in the corresponding planes, i.e., we compute cosine the angle between the two vectors such that $i = k$ where $i, k = \{R, G, B\}$.

As a consequence, three values are obtained; each one corresponds to a different plane (R, G, and B). Therefore, we sum these three values to be the final image score of similarity to the query image. This process is repeated for each image in our collection.

Finally, images are ranked and displayed in a descent order of their final scores. The model block diagram is shown in figure 11. Since the same stages are applied on the query and the database images, only one side (the database side) is shown in the figure, and the other side is generated in the same way.

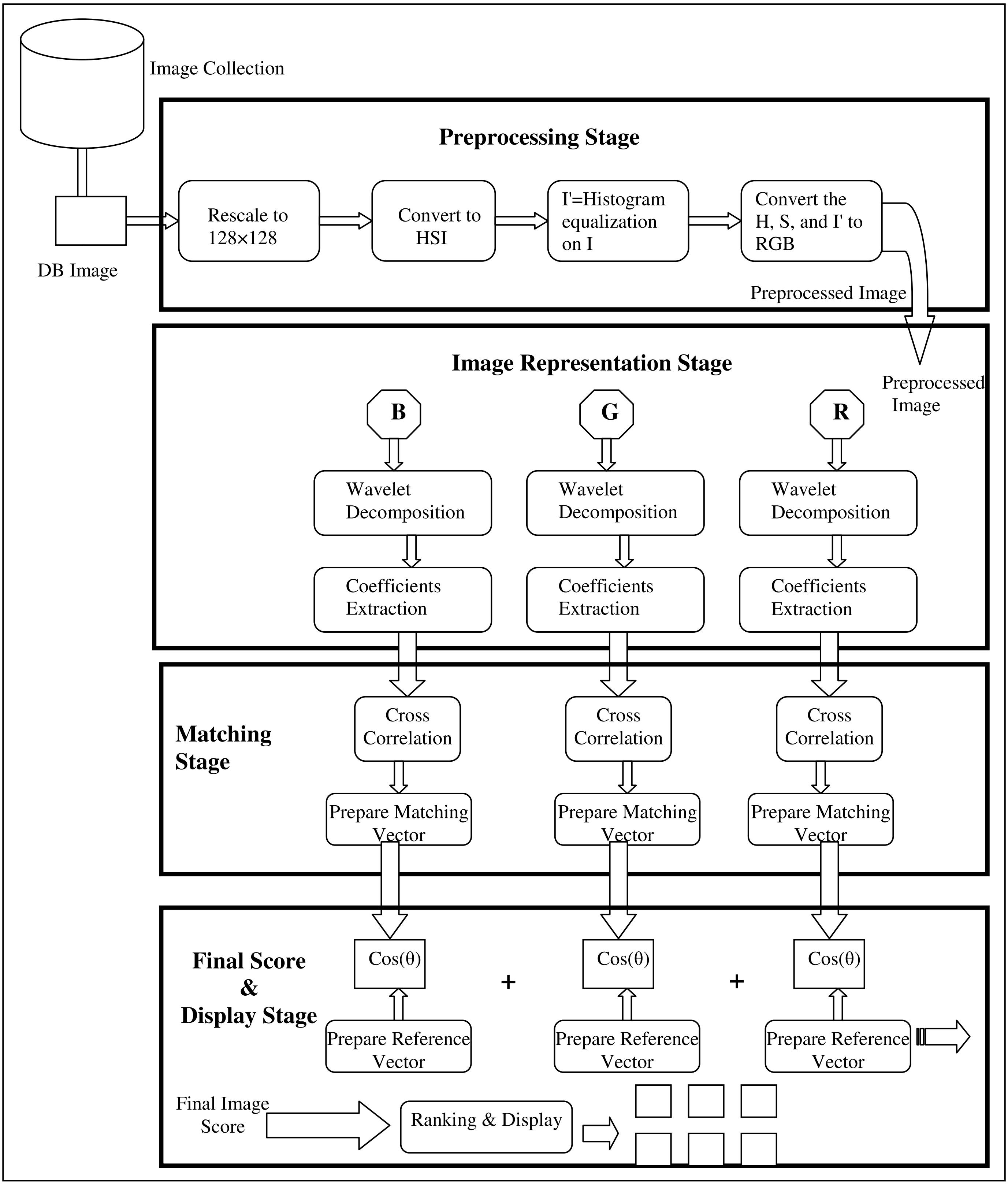


Figure 11 *The retrieval system using wavelet Coefficients*

4.1.1.1 A Variation to Model 1

A variation of this model is to form a single matching vector for the three planes for a database image as well as for the query image.

In other words, instead of having three matching vectors and three reference vectors, one for each plane, another way is to have a single reference vector and a single matching vector, both of 60 dimensions. The first 20 values of each vector represent the matching with the Red plane; the second 20 represent the matching with the Green plane; while the last 20 values hold the matching with the Blue plane.

By this way, we aggregate the matching vectors of different planes into a single representative vector. We compute cosine the angle between these two vectors i.e., the reference vector and the matching vector, this final score is the image similarity. Note that no summation, of cosine the angles in different planes, is required here.

4.1.2 Image Retrieval using Cross-Correlation on the coefficients of Daubechies' Wavelets on the basis of sub-image matching

In this model, we work on the intensity matrices of both, query and database images. The preprocessing stage consists of three steps. Firstly, the database and the query images are rescaled to a new size of 128×128 pixels. The images are then converted to their gray scale representation as discussed in section 1.5. Both the query and the database images are divided into five sub-images using Algorithm 4.3.

Now, we are ready for the image representation stage. This stage is similar to the one in the previous model, in terms of the way in which decomposition is applied, the type of wavelets used for the decomposition process, and in the matrix that is selected for the matching process, i.e., the approximation matrix of the third level of decomposition. However, it differs from the previous method in that; instead of applying the wavelet transform on the whole image at one time, we apply it on each sub-image that belongs to a particular image. Thus, five approximation matrices are obtained for each image, whether it is a database or a query image, each matrix corresponds to a sub-image of the original decomposed image.

The matching stage, between the query and a certain database image, of this model is achieved by applying the Cross-Correlation function five times, each time for one sub-image of the query and its counterpart from the database image.

In this model, we take the spatial layout information into consideration. That is, we are searching for images that are similar to the query not only on contents, but also in their spatial locations. Therefore, matching is done between the sub-images of a certain database image and their counterparts of the query image. In other words, the approximation matrix of subimage1 of the database image is cross correlated with the approximation matrix of subimage1 of the query image, the approximation matrix of subimage2 of the database image is cross correlated with the approximation matrix of subimage2 of the query image, and so on. As a result, five Cross-Correlation matrices are obtained.

Again, we take the highest 20 values of each resulted Cross-Correlation matrix. Thus, five matching vectors are obtained; each one corresponds to one region. To form the reference vectors, we again apply Cross-Correlation between the representations of the query sub-images (the approximation matrices), each one with itself. Hence, five reference vectors are also obtained; one for each region.

In the following step, cosine the angle between the matching vector of a particular region of a certain image and the reference vector for that region is computed. This process is computed for the five matching vectors. The result is five values that are added to each other to form the image final score of similarity to the searched query image.

After applying this procedure for all database images, and associating a scalar value with each image, the last step is to rank and display these images in a way according to their decreasing similarity scores.

4.1.2.1 A Variation to Model 2

Since the higher frequency bands represent texture and local color changes, a variation of the above model is achieved by using the whole third level decomposition, i.e., approximation and Horizontal, Vertical, and Diagonal details, not just the approximation matrix, as the image representation. These matrices are arranged in a single matrix so that, approximation and Horizontal matrices are on the left top and right top of the formed matrix, respectively, while Vertical and Diagonal matrices are in left bottom and right bottom, respectively. The resulted matrix from these four matrices that belongs to a certain query region is cross correlated with the corresponding one formed for the database image region. All other stages are hold in the same way as the previous model.

4.1.3 Image Retrieval using Cross-Correlation on the reconstructed images on the basis of sub-image matching

The first stage of this model is identical to the one of the model in section 4.1.2. Briefly, in the first stage, both the query and database images are rescaled to 128×128 pixels, converted to gray-scale representation, and divided into five partitions.

In the next stage, we compute a 2 layer 2-D wavelet transform on each of the five sub-images using Daubechies' Wavelets. Then, we reconstruct the sub-images back using only the second level of decomposition. Usually, the first level of decomposition contains noise and high frequency details. Ignoring it will not cause a significant loss of information.

In the matching stage, cross-correlation is applied between the reconstructed sub-images of the query and their counterparts of the database image regions. The reference vectors are composed here by cross correlating the reconstructed sub-images of the query with themselves.

Forming matching vectors, computing cosine theta, and ranking images are achieved by using the matching algorithm we discussed in section 4.1.2.

4.2 Applying Cross-Correlation on Spatial Domain

In this section, a model, in which Cross-Correlation is applied on spatial domain, is proposed.

4.2.1 Image Retrieval using Cross-Correlation applied on spatial representation of sub-images

In this model, the preprocessing stage consists of three steps. First, the image is rescaled to a new size of 64×64 pixels. Then, we multiply each of the three planes of the image (R, G, and B) by a scalar whose value is 255. As discussed earlier, the multiplication of an image by a scalar will highlight the different intensities that compose an image.

These two steps are applied to the query and database images. However, to make things appear simpler, the third step is applied only to the query image. In the third step, partitioning of the query image, to its four corresponding sub-images, is applied on each of the three planes.

Here, no transformation is applied on the images. Instead, template matching using Cross-Correlation is employed on spatial domain to identify patterns within an image. Hence, the next stage is to apply Cross-Correlation directly on the spatial representation of images.

In other words, we take the Red plane of the query sub-images and cross correlate it with the Red plane of an image selected from the dataset. We repeat this process for the Green and Blue planes of the query sub-images and the tested database image. Thus, for each region of the query image, three matrices are obtained, each from the Cross-Correlation in one of the three planes i.e., Red, Green, or Blue planes.

Forming the matching vectors is done in the same way as it's done in the model discussed in section 4.1.2. To prepare the reference vectors, we cross correlate each sub-image with itself. As a consequence, three reference vectors will be associated with each sub-image, one for each plane. This will results in 12 reference vectors (three for each sub-image); each one of them is of 20 dimensions.

The way to compute cosine the angle between the reference vector and the matching vector in a certain plane, and the computation of the final score are done as discussed previously in the model explained in section 4.1.2.

4.2.1.1 Variation I to Model 4

Instead of having three reference vectors for each sub-image, we did a variation of the previous model by combining the three representative vectors for a certain sub-image into a single vector of 60 dimensions. The first 20 values in that vector resulted from the correlation with the Red plane, the second 20 values resulted from the correlation with the Green plane, and the last 20 values computed by the correlation with the Blue plane. Except the above mentioned variation, all other steps of this model are the same as the model in section 4.2.1.

4.2.1.2 Variation II to Model 4

This model is the same as the model explained in section 4.2.1 except of the differences in the preprocessing stage. In the preprocessing stage, both the query and the database images are rescaled to a size of 32×32 pixels. Then, each plane (of the database images and the query image) is multiplied by 255. In the next step; we convert the images to their gray scale representations. Later, we divide the query image into four partitions (sub-images).

Again, to prepare the reference vectors, we cross correlate each partition with itself. Since the images are represented by a single matrix (Intensity), only a single vector is associated with each partition. Hence, just four matching vectors are associated with each database image. By keeping in mind that we are working in the intensity representation of images, we continue walking using the steps of the model discussed in section 4.2.1.

5 The Indexing Algorithm

In order to enhance the retrieval time needed to answer a user query, we developed an indexing technique using the backpropagation algorithm which works with all the proposed models. For that purpose, input vectors and the corresponding target vectors are used to train a feedforward network until input vectors can be classified in the right way.

In order to form the input vectors, images were rescaled to 128×128 pixels, and converted to their gray-scale representations. Then, a 3-layer 2-D wavelet transform is applied on all images. Input vectors are formed from only the third level of decomposition from all images in the dataset.

To achieve the target vectors, images were classified into seven classes: airplanes, big cats, monkeys, fireworks, flowers, sunset, and water sports. Thus, seven files are generated in which each file holds one class of the images. Target vectors are formed from the classes' numbers in which the corresponding images belong to. Classification of images into different categories depends on human subjectivity. However, categories that are used to classify images are very broad and not specific, which may limit the effect of human subjectivity problem. In addition, our main interest is on target-specific search in which it is possible to determine the classes of the images. To train the network, we selected fifteen neurons in the hidden layer. The transfer function we used is the tan-sigmoid, and the output layer transfer function is linear.

When it is the time of query, only the class, in which the selected image is a member in, is searched rather than the entire dataset. By using the proposed indexing algorithm, the retrieval time will be shorter than the case we don't use it. In addition, the quality of the retrieved images will be enhanced, since only related images to the one selected by the user are tested.

Analysis of Results

In this chapter, the CBIR models, which were proposed in the previous chapter, are evaluated. Moreover, many experiments are run on each of the proposed models as well as on their variations. Finally, experiments' results are analyzed.

Since the designed retrieval models are too many, their names were chosen to fulfill models' properties. The models are categorized into two: Spatial (S) and transformed domains (db for Daubechies and sym for Symmlet). The extracted coefficients are either the approximation values (A) or the detail values (D) or both (AD). The matching vector and the reference vector are either single space (s), in which a single vector is formed for the three color planes, or multi-space (m), in which three separated vectors are formed, one for each plane. The matching process is done either on the basis of the whole images' representations (w), or on the basis of sub-image matching (sub), in which the corresponding sub-images are matched. Images are represented either in the RGB color space (RGB), HSI color space (HSI), or in their gray scale representation (gray). For example, the model 'RGB db8AD3sw', means daubechies 8 (eight vanishing moments) with approximation (A) and details (D) extraction, and 3 layers decomposition applied on RGB images. In this model, the whole images' representations are matched and the vector matching is a single space (s). An example on the spatial domain abbreviations, the model 'RGB S32s', which means RGB 32x32 image in the spatial domain where the vector matching is a single space.

1 Experiments' Details

The performance of the proposed methods will be evaluated under two types of user aims when using a system which are target-specific search and the category search. For that reason, two measures are used to evaluate the performance of the proposed models; the retrieval accuracy of target-specific queries and the retrieval accuracy of category queries.

Basically, we will evaluate the ability of each algorithm to retrieve images that are adequately similar to the target image that is specified by the user. In other words, we will evaluate our models on the first type of user aims when using a system that we discussed in chapter two i.e., target search by example, rather than searching for images that share the same class with the query image. Images considered by this measure are used to compute the retrieval accuracy of target-specific queries on each of the proposed models. In addition, ranking of the retrieved images, that are in the same category as the query image, is used to compute the retrieval performance of the models for the category search queries type as discussed later in this section.

In our experiments, we considered a small database consisting of 144 images of seven classes such as airplanes (19 images), flowers (20 images), water sports (20 images), sunset (20 images), big cats (25 images), monkeys (20 images), and fireworks (20 images). All images in the database have been associated using one of these classes and this serves as the ground truth.

In order to test the retrieval accuracy and ranking results for each individual class using one of the proposed models, we randomly picked two images from each class and used them as queries. Those picked images are then used by each other models for later comparisons. For each individual class, we computed the retrieval accuracy as the average percentage of images of the same target or the same class, depending on the type of queries we are considering, as the query image that were retrieved in the top n images.

Although our main interest is in unique images target search, we also test the ability to retrieve images that are in the same category as the query image. For that purpose, three types of ranking are taken into consideration: high ranking, mid ranking, and low ranking. Those three ranking quantities represent the retrieval accuracy on category search queries.

The high ranking, mid ranking, and low ranking measurements, count the number of images that were retrieved such that they are in the same class as the query image. The difference between these three quantities is in the size and the location in the retrieved images list that are considered to compute the quantities from. To make things clear, the high ranking is computed from the top 20% retrieved images, mid ranking is computed from the next 50% retrieved images. This 50% is computed after truncation of the first 20%. Finally, low ranking is computed from the next 20 retrieved images. This latter percent is computed after truncating the first 20% and the next 50% retrieved images from the retrieved ranking images list.

2 Experimental Results

In this section, the significant simulation results are introduced. Experiments are run on all the proposed models and on their variations. It is important to say that all the experiments that are reported in this section are done without using the so far proposed indexing algorithm.

2.1 Hardware and Software

The models were implemented in MatLab version 6.5. The tests and experiments were run on a modern standard PC (2.40 GHz, 768 MB of RAM) running under Windows XP.

2.2 Database

The pictorial database was assembled using images from the greenstreet image collection CD. Images with a common theme such as flowers, sunset, water sports, etc, were selected to make experiments on them. This database was used in all the proposed models, where the user is required to select one image as a query to initiate the search process. The size of the dataset we deal with is 144 images, where images are classified into seven classes as we argued earlier.

All the experiments that are reported in this thesis were conducted with the color images. According to the model we are testing, we may perform different operations on images such as converting them to their gray-scale representations.

2.3 Results of the Proposed Models

In this section, the experimental results of all the proposed models are presented. As we explained in the previous chapter, the proposed models are classified into frequency-domain models and time-domain models.

2.3.1 Frequency Domain Models

Many experiments are run based on the approaches that are classified, in the previous chapter, under the frequency domain models.

2.3.1.1 Image Retrieval using Cross-Correlation on approximation coefficients of Daubechies' Wavelets

Many experiments are run on different versions of that model. Experiments are run on this model by varying three parameters: the type of wavelet used for decomposition, the number of levels of decomposition, and the way in which the matching vectors and the reference vectors is formed. Hence, eight versions of this model are tested. In the first version 'RGB db8A2mw', we use "Daubechies-8" wavelets and we decompose images up to two levels. In the second version 'RGB db8A3mw', we use "Daubechies-8" and we decompose images up to three levels. In the third version 'RGB sym8A2mw', we use "Symmlet-8" wavelet and the decomposition is up to two levels. While in the fourth version 'RGB sym8A3mw', the same wavelet used in the third version is used and the decomposition is up to three levels. The first four versions are tested using one matching vector and one reference vector for each plane. In other words, three matching vectors are associated with each database image, and three reference vectors are associated with the query image. Tables, from 1 to 4, show the results of experiments that are done on the above summarized versions of this model.

Table 1: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB db8A2mw'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	61	61	59	25
Water Sport	48	48	39	19
Sunset	50	50	73	20
Flower	100	45	61	33
Big Cat	100	38	42	29
Monkey	100	30	41	20
Firework	30	30	57	17

Table 2: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB db8A3mw'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	63	63	57	33
Water Sport	50	50	36	0
Sunset	63	63	53	71
Flower	100	48	69	0
Big Cat	100	40	49	19
Monkey	100	30	44	6
Firework	33	33	45	20

Table 3: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB sym8A2mw'*

	Target Specific Search	Category Search		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	24	24	69	13
Water Sport	30	30	61	39
Sunset	48	48	62	11
Flower	100	30	25	40
Big Cat	100	38	67	7
Monkey	100	45	73	0
Firework	23	23	58	9

Table 4: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB sym8A3mw'*

	Target Specific Search	Category Search		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	40	40	61	10
Water Sport	43	43	79	67
Sunset	43	43	64	7
Flower	100	25	24	22
Big Cat	100	38	40	12
Monkey	100	38	72	13
Firework	33	33	50	0

In the second four versions, the same parameters are set as the previous group except that experiments are run using one matching vector for each database image and one reference vector for the query image. Tables, from 5 to 8, show the results of experiments that are done on those four versions.

Table 5: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB db8A2sw'*

	Target Specific Search	Category Search		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	43	43	82	25
Water Sport	45	45	67	63
Sunset	48	48	57	38
Flower	100	43	57	54
Big Cat	100	48	67	25
Monkey	100	35	77	-
Firework	18	18	41	26

Table 6: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB db8A3sw'*

	Target Specific Search	Category Search		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	66	66	69	50
Water Sport	50	50	67	17
Sunset	68	68	31	0
Flower	100	48	60	40
Big Cat	100	42	50	0
Monkey	100	38	48	43
Firework	23	23	57	30

Table 7: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB sym8A2sw'*

	Target Specific Search	Category Search		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	61	61	75	-
Water Sport	38	38	70	9
Sunset	50	50	50	50
Flower	100	23	49	24
Big Cat	100	42	71	0
Monkey	100	45	91	50
Firework	45	45	74	0

Table 8: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB sym8A3sw'*

	Target Specific Search	Category Search		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	50	50	100	-
Water Sport	48	48	86	0
Sunset	50	50	53	7
Flower	100	30	61	19
Big Cat	100	40	49	31
Monkey	100	50	66	13
Firework	50	50	67	10

As we said earlier, we selected two images from each class to be our query images and did our experiments. The first column of each table shows the retrieval accuracy of the target-specific search query. This retrieval accuracy was given by the percentage of correct target images that were retrieved in the top 20% of all the retrieved images. In our case, we compute the accuracy from the first retrieved image up to the 29th retrieved image.

It is important to say that when computing the accuracy, we look at the unique target images that are similar to the query rather than on similar category images. However, in some classes such as airplane, water sport, sunset, and firework, no unique target images are exist in the image collection we had. Therefore, we compute the accuracy based on images in the same class as the query image. Consequently, it is the same as the high ranking.

The following two figures give examples on target-specific retrieval and category retrieval. In figure 12, images retrieved using target-specific query are shown using 'RGB db8A2mw'. The query image is selected from the flower class. Only five images for that specific flower are existed in the tested dataset, which are found by this version of that model. Figure 13 shows the results of category search query, this time we select the query image from the airplane class. It is obvious that in target-specific search we are interested in finding different images for exactly the same object(s) existed in the query image. While in category search, we are interested in retrieving other members of the same class as the query image.

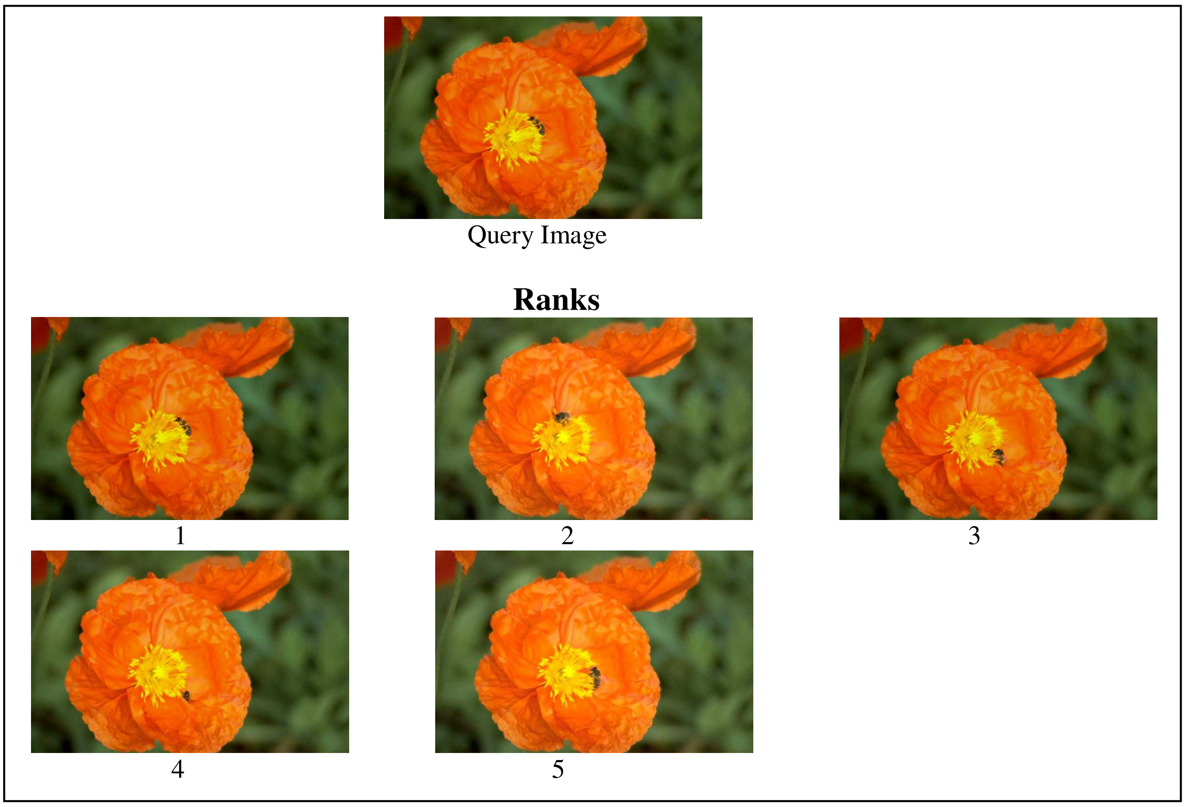


Figure 12 *Target retrieved images in the top 20% returned images for a target query image using 'RGB db8A2mw'*

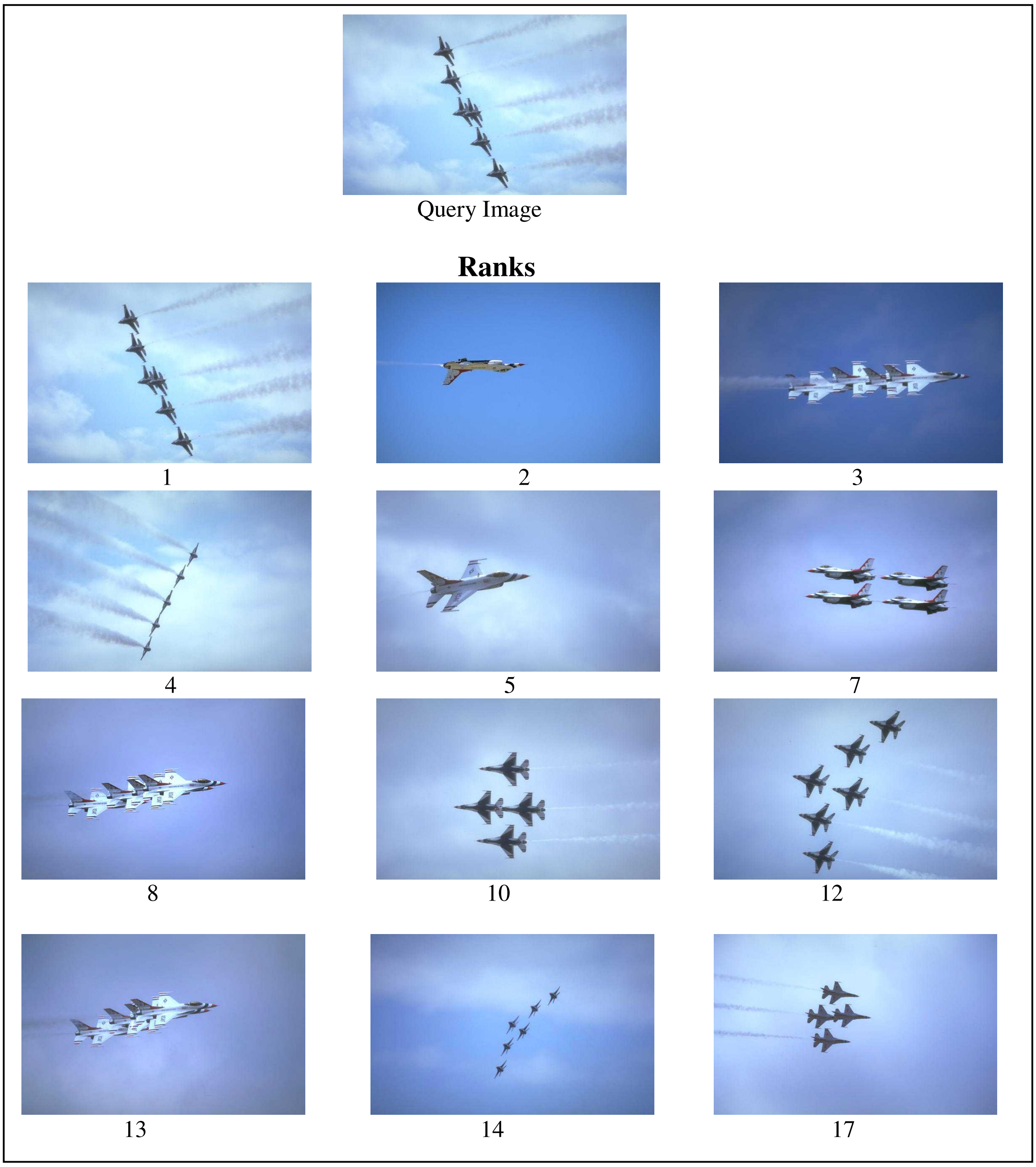


Figure 13 *Ranked images in the top 20% returned images of Category-search type using 'RGB db8A2mw'*

To highlight the difference between target-specific retrieval and category retrieval, figure 14 shows both, the target and the category ranked images in the top 20% retrieved images for a selected query image. Clearly, the first five retrieved images are target-specific images in which all of them are exactly for the same monkey submitted in the query image, while the remaining images share the same category with the query image; the monkey category.

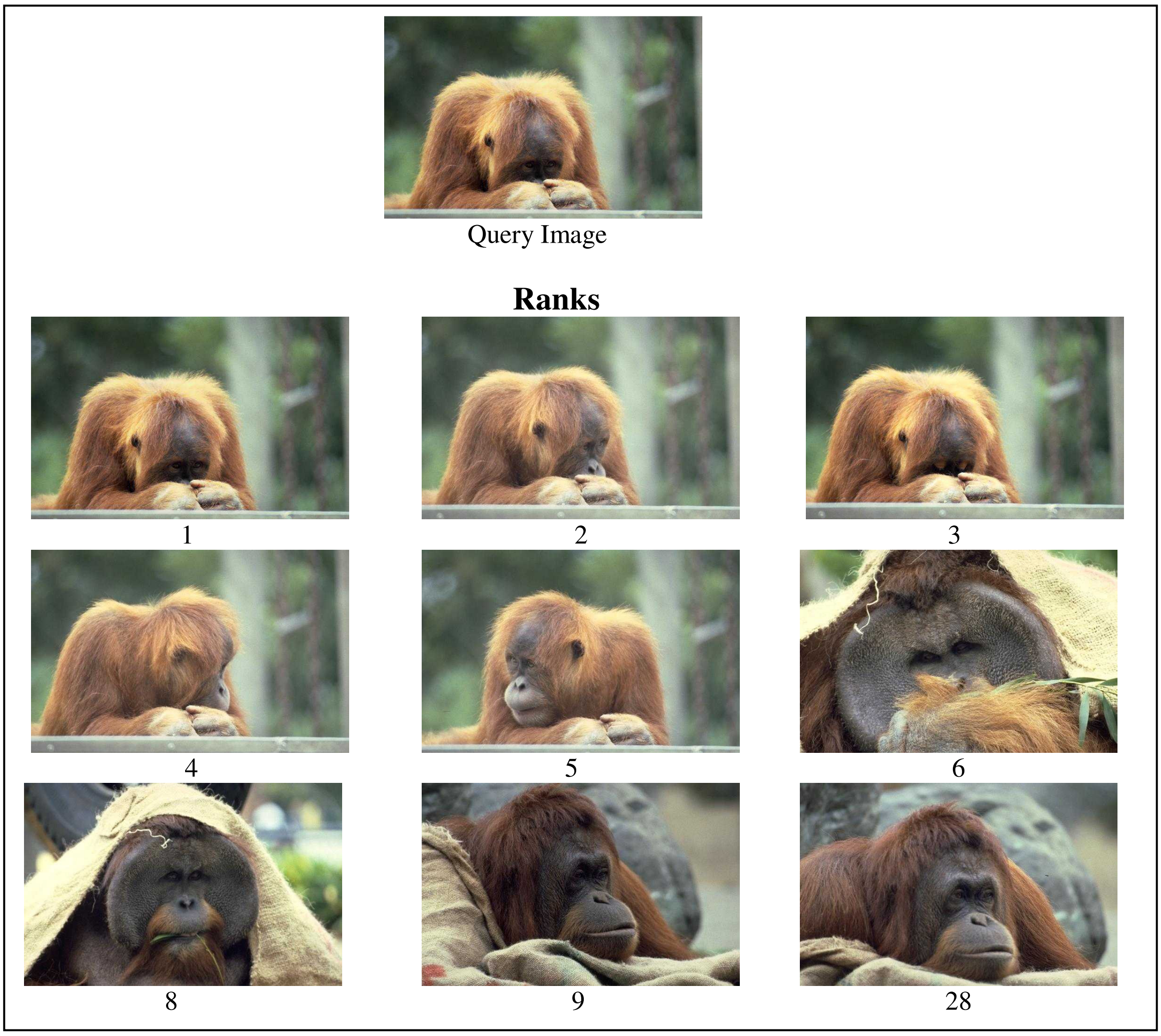


Figure 14 *Ranked images in the top 20% returned images using 'RGB db8A2mw'*

In high ranking, we look at the first 29th retrieved images, but this time we are interested in counting retrieved images that are in the same category as the query image. This percent is computed from the whole images in that category. For example, the value of high ranking for the airplane class in table one is computed for the first query image as the number of retrieved images that are in the same category as the query image divided by the total number of images in that class. That is, $HighRanking = \frac{12}{19}$.

The same percent is computed for the second query image. Finally, we take the average of those two values.

The mid ranking column shows, also, similar category images, but in the next 50% retrieved images. That is, ranked images in positions 30-87. To compute mid ranking percent, we subtract the number of images, that are in the same category as the query image, that were retrieved in the first 29th retrieved images from the total number of images in the specified category. The resulted size will serve as the new size for that category to compute the mid ranking from. For example, in order to compute the value of mid ranking for the airplane class in table one for the first query image, we have to find the new size of the category. Since 12 images were retrieved in high ranking, the new size will be 19-12 = 7. The number of images that were retrieved in the mid ranking is 3. Therefore, the mid ranking percent for the first query image on the specified class is $\frac{3}{7}$.

In the low ranking, we are, again, interested in retrieving images in similar class as the query but this time in the next 20% retrieved images. That is, ranked images in positions 88-98. Low ranking is computed in the same way as mid ranking. In other words, we subtract the number of images that are in the same class as the query image, which were used to compute the high ranking and mid ranking percents, from the total number of images of the specified category. We use the new size to compute the low ranking percent. In the recorded results for the low ranking, when the value is (-), this means that all the images of the specified class were retrieved in high ranking and mid ranking percents.

2.3.1.2 Image Retrieval using Cross-Correlation on the coefficients of Daubechies' Wavelets on the basis of sub-image matching

Experiments are run on this model by varying two parameters: the type of wavelet used for decomposition ("Daubechies-8" or "Symmlet-8"), and the representation of images that is used for the matching step i.e., the approximation matrix of the third level of decomposition or the approximation and details of the same mentioned level. Hence, four versions of this model are tested.

Table 9 and Table 10 show the results of experiments on two versions, 'Gray db8A3s-sub' and 'Gray db8AD3s-sub', of this model using "Daubechies-8" wavelets. Table 9 shows the results of experiments using only the approximation coefficients of the third level of decomposition, while Table 10 shows the results of using the approximation and detail coefficients of the third level of decomposition.

Table 9: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray db8A3s-sub'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	34	34	39	5
Water Sport	28	28	52	38
Sunset	63	63	78	0
Flower	100	28	49	30
Big Cat	77	30	32	0
Monkey	90	53	63	25
Firework	33	33	53	5

Table 10: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray db8AD3s-sub'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	77	77	90	-
Water Sport	43	43	79	25
Sunset	68	68	84	0
Flower	100	40	27	41
Big Cat	100	28	61	15
Monkey	100	38	29	14
Firework	63	63	55	10

Table 11 and Table 12 shows the results of experiments done on the other two versions, which are 'Gray sym8A3s-sub' and 'Gray sym8AD3s-sub'. In these two versions we used "Symmlet-8" wavelets instead of "Daubechies-8". Again, Table 11 shows the results of experiments using only the approximation coefficients of the third level of decomposition, while Table 12 shows the results of using the approximation and detail coefficients of the third level of decomposition.

Table 11: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray sym8A3s-sub'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	24	24	69	33
Water Sport	20	20	41	11
Sunset	48	48	60	44
Flower	100	30	50	23
Big Cat	72	42	38	12
Monkey	100	45	91	-
Firework	50	50	59	25

Table 12: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray sym8AD3s-sub'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	71	71	92	-
Water Sport	55	55	73	67
Sunset	75	75	70	75
Flower	100	43	38	16
Big Cat	100	36	44	39
Monkey	100	38	32	0
Firework	68	68	67	0

2.3.1.3 Image Retrieval using Cross-Correlation on the reconstructed images on the basis of sub-image matching

In this model, we use "Symmlet-8" wavelets to decompose images to their second level representations. Table 13 shows the results of experiments that are done on that model.

Table 13: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on model 3*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	27	27	36	22
Water Sport	33	33	68	25
Sunset	28	28	65	19
Flower	100	48	75	13
Big Cat	79	28	61	29
Monkey	100	45	63	38
Firework	50	50	55	44

2.3.2 Spatial Domain Models

This section shows the results of experiments that are run on the proposed models which work on images in the spatial-domain.

2.3.2.1 Image Retrieval using Cross-Correlation applied on spatial representation of sub-images

We did different experiments on two sizes of images, 64×64 and 32×32, to select the suitable size to rescale the images to. The results of the experiments that are run on the size 64×64 are shown in Table 14, we called it 'RGB S64m', while the results of the experiments that are run on the size 32×32 are shown in Table 15, and it is called 'RGB S32m'. In these experiments, we have three reference vectors for the query image, and three matching vectors for each database image; vector for each color plane.

Table 14: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB S64m'*

	Target Specific Search	Category Search		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	71	71	88	-
Water Sport	55	55	90	-
Sunset	53	53	52	20
Flower	100	60	95	-
Big Cat	100	50	65	8
Monkey	100	40	63	10
Firework	50	50	40	22

Table 15: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB S32m'*

	Target Specific Search	Category Search		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	47	47	72	-
Water Sport	48	48	88	-
Sunset	50	50	59	22
Flower	100	48	43	40
Big Cat	93	50	74	20
Monkey	100	50	70	38
Firework	50	50	41	44

Also, we did experiments on the first variation of this model that is mentioned in the previous chapter, which, briefly, said to have a single matching vector for each image in the dataset as a representative vector for the three planes. This strategy is applied also on the reference vectors. Table 16 shows the results of these experiments. In those recorded results, we considered the size of the query and each of the database images to be 32×32. We called this variation as 'RGB S32s'.

Table 16: The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'RGB S32s'

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	76	76	100	-
Water Sport	60	60	88	-
Sunset	50	50	61	0
Flower	100	43	40	15
Big Cat	100	52	69	29
Monkey	100	70	100	-
Firework	43	43	57	13

Moreover, we did similar experiments using the HSI color space as a representative color model for images instead of the RGB color model. Table 17 shows such results. We called this variation as 'HSI S32s'.

Table 17: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'HSI S32s'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	77	77	88	-
Water Sport	50	50	88	-
Sunset	23	23	33	20
Flower	100	30	48	20
Big Cat	100	52	54	17
Monkey	100	50	68	-
Firework	45	45	59	-

In addition, experiments are run on the second variation of this method, in which images are converted to their gray-scale representations as we explained in the previous chapter. Experiments are run on two sizes of images: 64×64 and 32×32. Tables 18 and 19, show the results of this variation on the two sizes. When the images are resized to

64×64 pixels; we called this version as 'Gray S64s', while when the resize is to 32×32 pixels; it is called 'Gray S32s'.

Table 18: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray S64s'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	71	71	92	-
Water Sport	55	55	72	-
Sunset	33	33	52	16
Flower	100	45	41	17
Big Cat	93	50	65	8
Monkey	100	43	65	27
Firework	33	33	53	30

Table 19: *The retrieval accuracy percent (%) of target-specific query and category query (Ranking) applied on 'Gray S32s'*

	Target Specific Search	**Category Search**		
Class	**Accuracy**	**High Ranking**	**Mid Ranking**	**Low Ranking**
Airplane	66	66	100	-
Water Sport	60	60	89	-
Sunset	48	48	59	42
Flower	100	33	56	7
Big Cat	93	48	66	50
Monkey	100	48	82	34
Firework	38	38	51	7

3 Discussion of Results

In this section, experimental results from the previous section are analyzed and discussed. The enhancement of retrieval time when using the proposed indexing algorithm is presented. Finally, a comparison between different versions of each of the proposed models is presented according to system accuracies.

In our comparisons of query results, we consider one retrieval model as better than another, if the number of best matching images is higher. We do not attempt to compare two images in which both of them are very similar to the query image; because it depends on human subjectivity. When several images are very close to the query image, it is meaningless to rank their similarities to the query image, since subjective opinions often dominate and distances are too close to make ranking orders simply based on sorting results.

3.1 Frequency Domain Models' Results

In this section, we analyze the results of the experiments that are run on the models that are proposed to work on images in the frequency domain.

3.1.1 Image Retrieval using Cross-Correlation on approximation coefficients of Daubechies' Wavelets Results

The results of the experiments that are run on the eight versions of this model are recorded in the previous section. In order to find the best version of this model, comparisons are done in two stages. In the first stage, we compare the first four versions in order to find the best one among them. At the same time, comparison is done between the last four versions and also we find the best one. Then, we compare the best two versions with each other to know which outperforms the other.

If we look at the first four versions, it is clear that 'RGB db8A3mw' has the best results, among the first four versions of this model, in six categories which are airplane, water sport, sunset, flower, big cat, and firework categories. This is because of two reasons: firstly; the use of "Daubechies" Wavelet in this model is better than the use of "Symmlet" Wavelets, and secondly; the matching step depends on a deeper level of decomposition i.e. the third level of decomposition. Moreover, in this version we have a separated vector for each plane. However, for the monkey category, 'RGB sym8A2mw' and 'RGB sym8A3mw' perform a little better than 'RGB db8A3mw'.

In the last four versions of this model, 'RGB db8A3sw' did the best work on five categories, which are airplane, water sport, sunset, flower, and big cat categories. However, for the monkey and firework categories, 'RGB sym8A3s' which uses "Symmlet" Wavelets, to decompose images to their third level representations, is the best one.

We compare the best version from the first group, with the best one from the second group, that is, we compare 'RGB db8A3mw' with 'RGB db8A3sw'. We found that 'RGB db8A3sw' of this model outperforms the other in four categories which are airplane, sunset, flower, and big cat. For the water sport category, both versions give similar retrieval results. The only difference between 'RGB db8A3sw' and 'RGB db8A3mw' is in the way in which the reference vectors and the matching vectors are formed. For the monkey and firework categories, 'RGB sym8A3s' outperforms all other versions. According to the results, using a single representative vector for the three planes instead of three vectors, one for each plane, give us better results. Figure 15 shows the average number of hits for each category using only the high ranking percents of the experiments done on 'RGB db8A3mw', 'RGB db8A3sw', and 'RGB sym8A3s'.

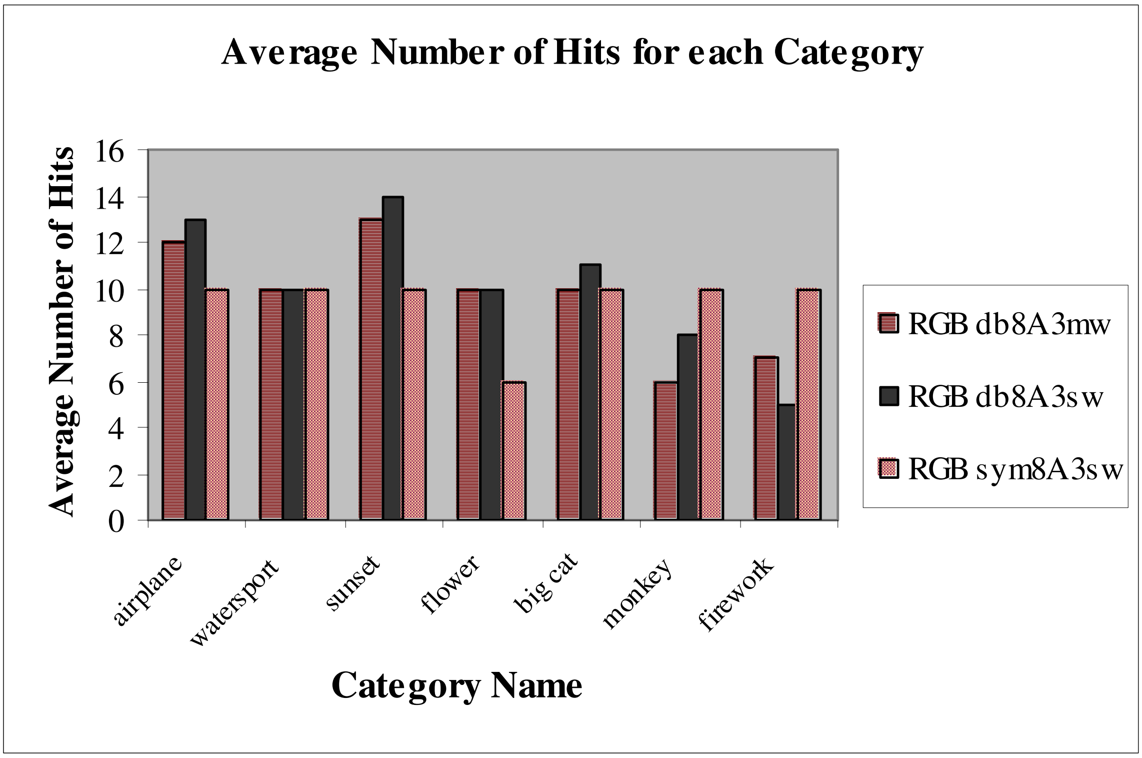


Figure 15 *The average number of hits for each category using 'RGB db8A3mw', 'RGB db8A3sw', and 'RGB sym8A3sw'*

To match the target-specific queries, the average accuracy of all versions of this model is 100%, which means that our proposed model can capture all the target images that are semantically similar to the one asked by the user. The use of this type of queries is clear in three classes, which are big cats, monkeys, and flowers. Figure 16 shows an example of using 'RGB db8A3sw' of this model to response to a target-specific query. In this figure, all the images in our database that are related to the same tiger are retrieved in the first 20% retrieved images. More precisely, they are the first eight retrieved images.

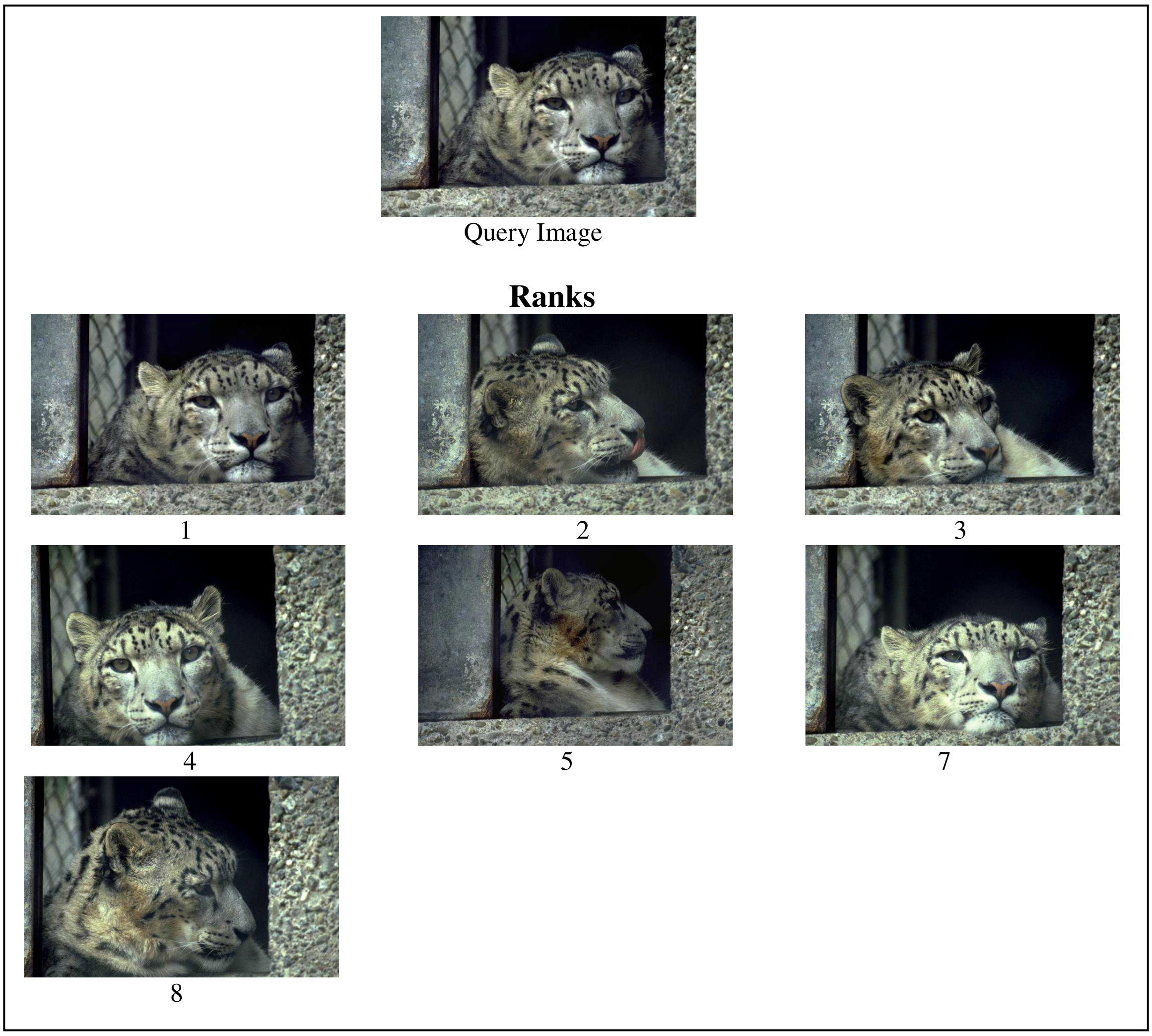


Figure 16 *Target retrieved images in the top 20% returned images from a target query image using 'RGB db8A3sw'*

3.1.2 Image Retrieval using Cross-Correlation on the coefficients of Daubechies' Wavelets on the basis of sub-image matching Results

Four versions of this model are tested in the previous section; in which two of them used "Daubechies-8" wavelets, while the other two used "Symmlet-8" wavelets. In order to find the best version of this model, we compare the results of the experiments that are done on the four versions.

Figure 17 shows the average number of hits using the high ranking results for 'Gray db8A3s-sub', 'Gray db8AD3s-sub', 'Gray sym8A3s-sub', and 'Gray sym8AD3s-sub'. As we discussed previously, 'Gray db8A3s-sub' and 'Gray db8AD3s-sub' are using

"Daubechies-8" to decompose images. While in 'Gray sym8A3s-sub' and 'Gray sym8AD3s-sub', we used "Symmlet-8" wavelets.

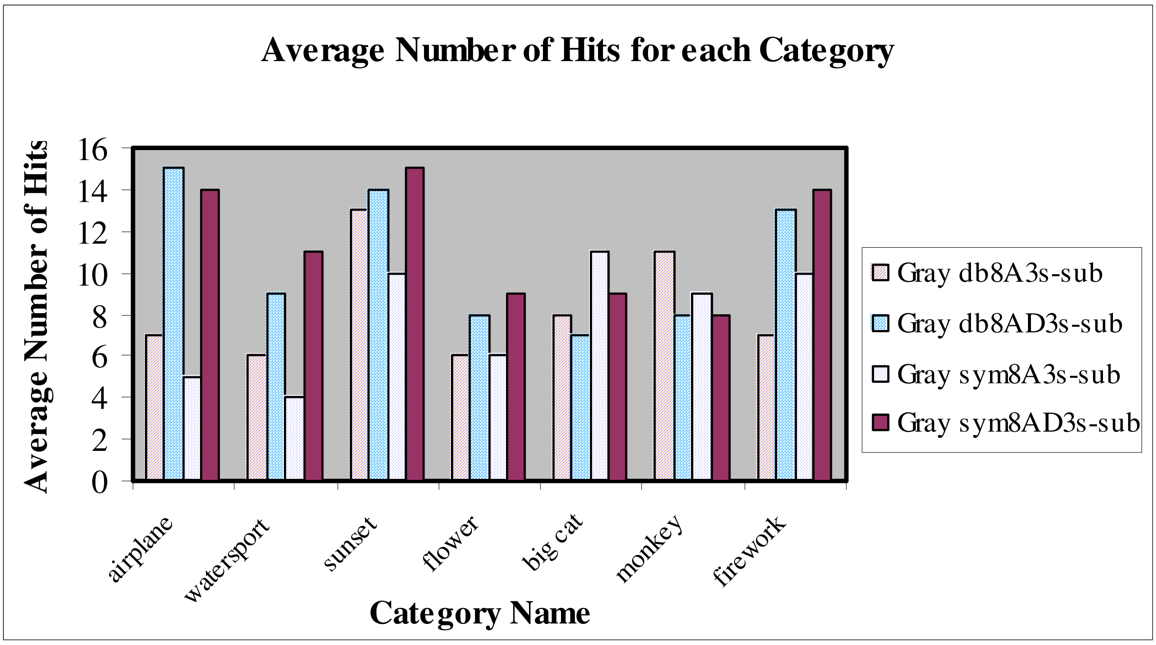


Figure 17 *The average number of hits for each category using 'Gray db8A3s-sub', 'Gray db8AD3s-sub', 'Gray sym8A3s-sub', and 'Gray sym8AD3s-sub'*

From the comparison of these four versions, it is clear that 'Gray db8AD3s-sub' and 'Gray sym8AD3s-sub' give the best results in airplane category, with a little improvement of results in 'Gray db8AD3s-sub'. 'Gray db8A3s-sub' performs the best one when working with monkey category. On the other hand, 'Gray sym8A3s-sub' performs a little better than 'Gray sym8AD3s-sub' when working with the big cat category. While in the remaining four categories, which are water sport, sunset, flower, and firework, 'Gray sym8AD3s-sub' performs the best one among all the versions of that model, followed by 'Gray db8AD3s-sub'. This is because, in 'Gray sym8AD3s-sub' and 'Gray db8AD3s-sub', we take into consideration the details information in addition to the approximation coefficients of the third level of decomposition.

It is important to note that 'Gray db8AD3s-sub' is using "Daubechies-8" while 'Gray sym8AD3s-sub' is using "Symmlet-8". Since each version yields to good results in certain categories, it is not easy to say which one is better than the other, in general. But, if we look at specific categories we can determine which one performs better than the others.

To compare the versions in terms of the retrieval accuracy for the target-specific queries, it is obvious that the versions that use "Symmlet" outperform the other two versions that use "Daubechies". Moreover, 'Gray sym8AD3s-sub' gives the most accurate results.

3.1.3 Image Retrieval using Cross-Correlation on the reconstructed images on the basis of sub-image matching Results

In this model, we use "Symmlet-8" to decompose images to their second level representations. Figure 18 shows the average number of hits using high ranking results of that model.

One can easily note that this model performs less than the other proposed models in many classes such as airplane, water sport, sunset, and big cat. However, the best achieved results are in the fireworks class. The average accuracy for target-specific queries, also, is less than the other models.

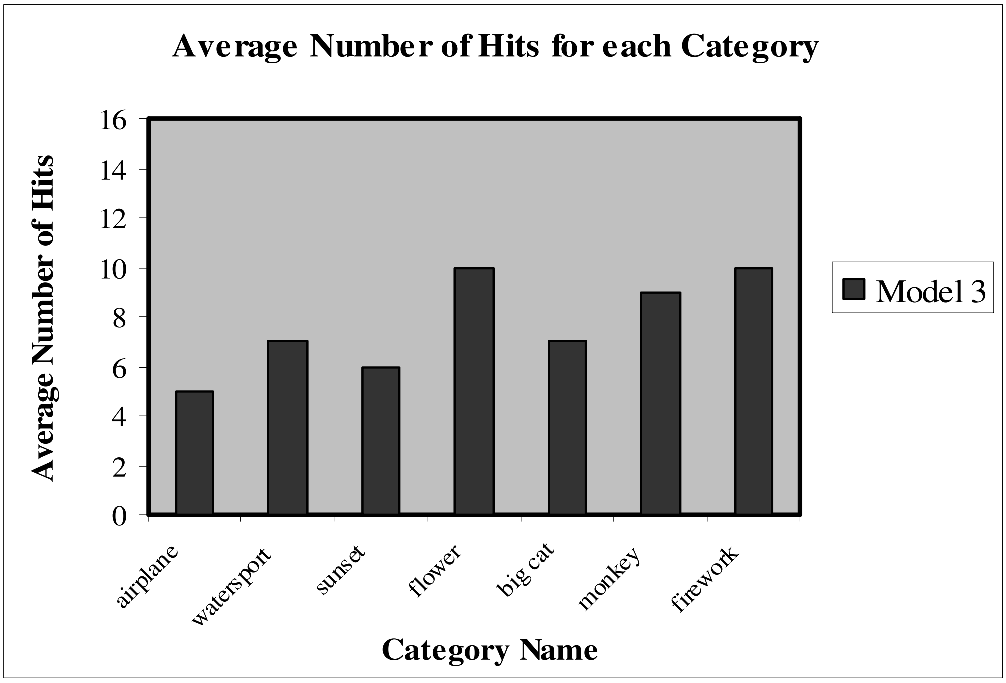


Figure 18 *The average number of hits for each category using the third proposed model under the frequency domain category*

3.2 Spatial Domain Model's Results

In this section, we discuss the results of the model that we proposed to work in images in their spatial domain representations.

3.2.1 Image Retrieval using Cross-Correlation applied on spatial representation of sub-images Results

Three sets of experiments are run on this model and on its variations. In the first set, we test the new size that will be used for the query and database images. Two sizes are considered 64×64 and 32×32. 'RGB S64m', which uses the size of 64×64, achieves better results in airplane, water sport, sunset, and flower categories. On the other hand, 'RGB S32m' performs little better than 'RGB S64m' in terms of high ranking on monkey category. Both of those versions have similar results on firework category. Moreover, both versions have similar results of high ranking on big cat category. However, 'RGB

S64m' yields to better retrieval accuracy results for target-specific queries that are applied on the big cat category.

Figure 19 shows the average number of hits using the two sizes. In these set of experiments, we have three reference vectors for the query image and three matching vectors for each database image; vector for each color plane. In general, 'RGB S64m', yields to better results in terms of retrieval accuracy for target-specific queries and retrieval accuracy for category queries.

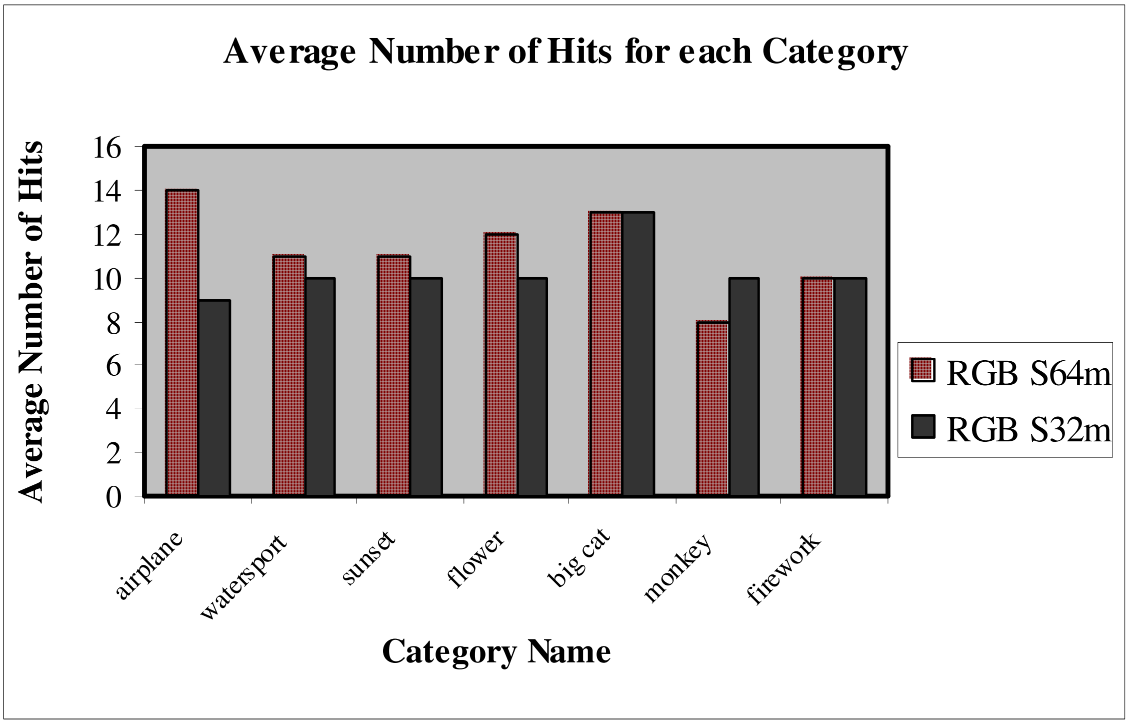


Figure 19 *The average number of hits for each category using 'RGB S64m' and 'RGB S32m'*

In the second set of experiments, we have a single reference vector for the query image and a single matching vector for each database image. In those experiments, images are represented in the RGB color space one time, while in the other time; they are represented in the HSI color space. In those two versions, 'RGB S32s' which works on RGB images has better results than 'HSI S32s' on four categories, which are water sport, sunset, flower, and monkey. Both versions give similar results on the other three categories, which are airplane, big cat, and firework categories. Figure 20 shows the average number of hits using 'RGB S32s' and 'HSI S32s'. In general, all the so far mentioned versions of this model yield to the same average retrieval accuracy.

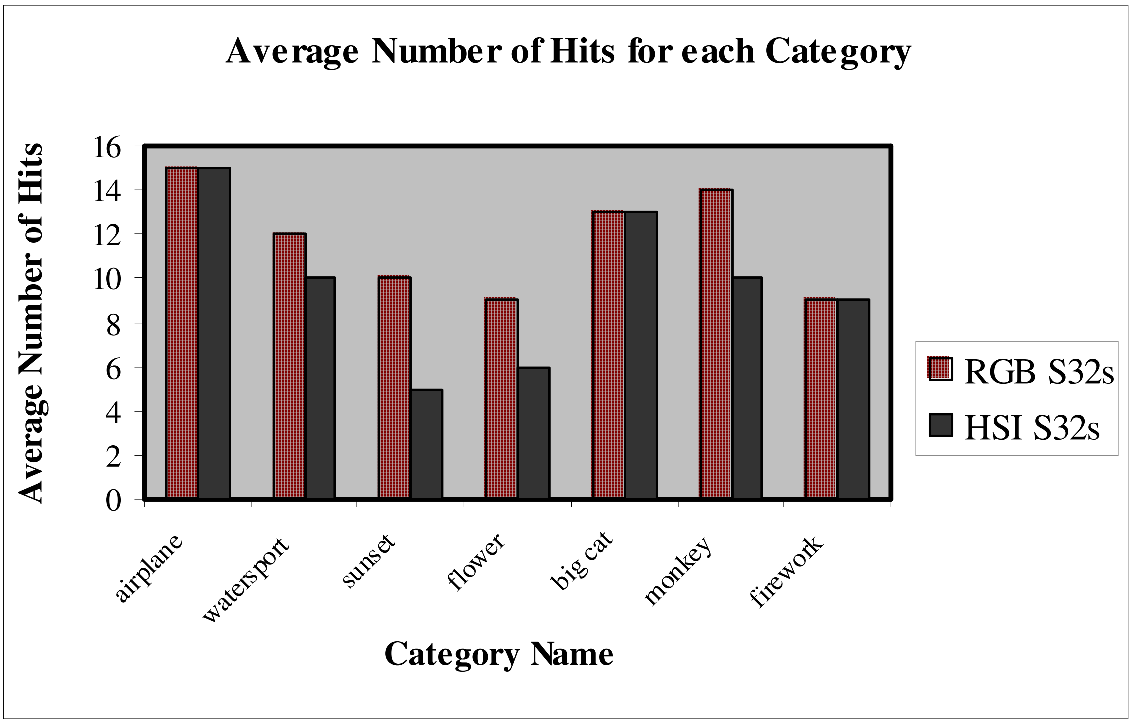


Figure 20 *The average number of hits for each category using 'RGB S32s' and 'HSI S32s'*

To make a clear distinction between the capabilities of the different discussed versions of that model, we plot the four versions, 'RGB S64m', 'RGB S32m', 'RGB S32s', and 'HSI S32s', all together in figure 21.

Finally, the third set of experiments run on the second variation of this model in which images are converted to their gray-scale representations. Two sizes are examined in our experiments which are 64×64 and 32×32. Figure 22 shows the average number of hits of high ranking for the two sizes. Although this variation performs well in many categories, the previous discussed versions of that model outperform 'RGB S32s' and 'HSI S32s'.

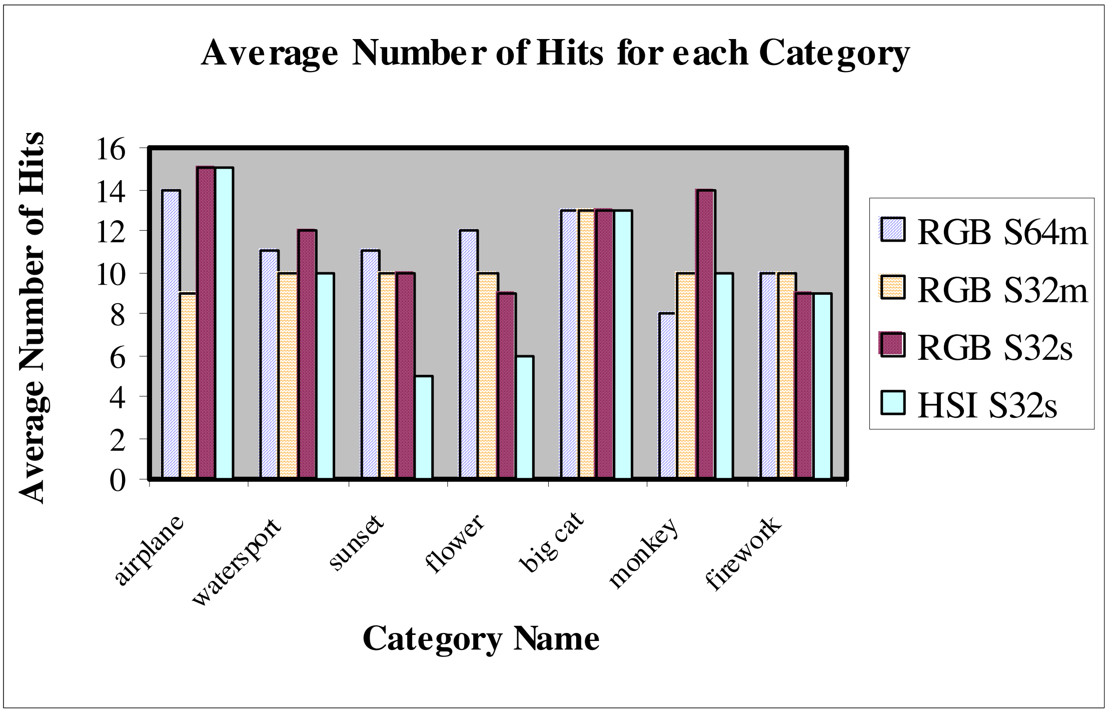


Figure 21 *The average number of hits for each category using 'RGB S64m', 'RGB S32m', 'RGB S32s', and 'HSI S32s'*

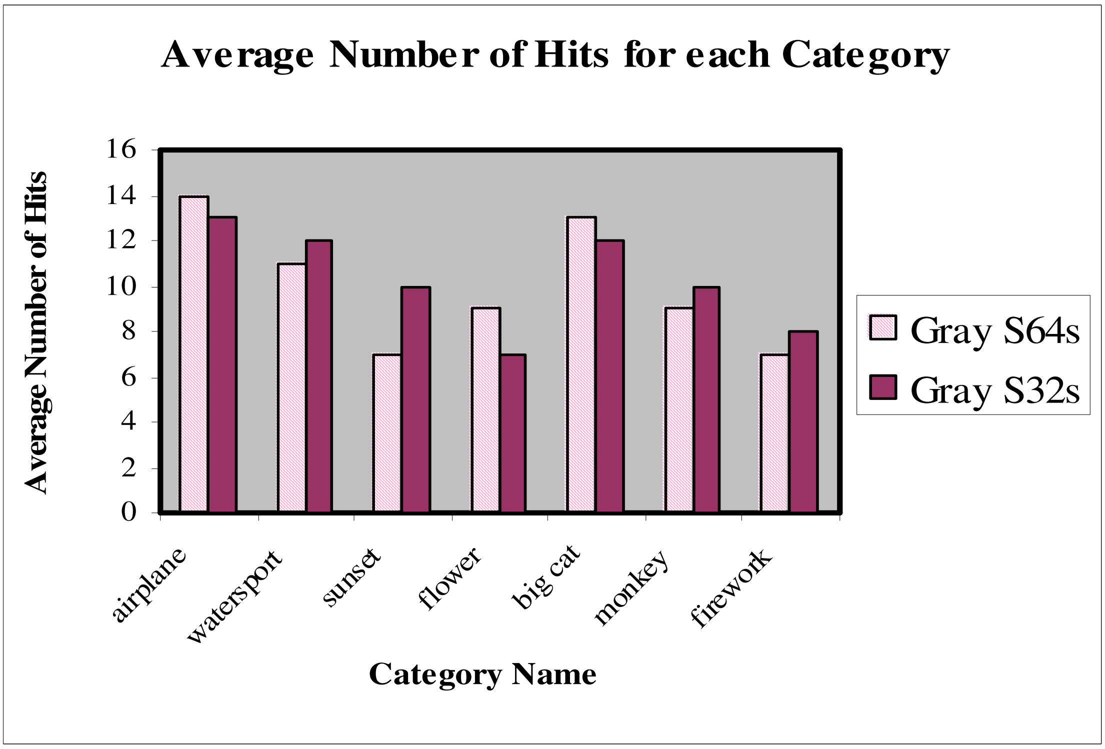


Figure 22 *The average number of hits for each category using 'Gray S64s' and 'Gray S32s'*

3.3 The Enhancement of retrieval time

All the experimental results that we discussed are achieved without the using of the proposed indexing algorithm. As we mentioned earlier, the main purpose of using the indexing algorithm is to enhance the retrieval time needed to answer the user query. In other words, to shorten the waiting time needed once a user makes a request to the time that the answer comes back. In addition, by using the indexing algorithm, all the returned images will be, in somehow, related to the query image. In other words, the retrieval accuracy of category search queries will be improved.

Table 20 compares the time needed to process a single user query with one version of each proposed model, with and without the indexing algorithm; we take the best version (if two or more are good, we select randomly one of them) of each model. Considering the time needed to process the query, using the indexing algorithm manage to achieve a reasonably fast response. This is because; all the computations of a certain model are applied only on one category; which is the same as the query image. To compute the time, we did our experiments on the category that contains the highest number of images, which is the big cat category, in order to compute the time needed in the worst case.

Table 20: *Time (in seconds) taken to process user query on different models*

Model Name	Without the Indexing Algorithm	With the Indexing Algorithm
RGB db8A3mw	186.9	36.2
Gray sym8AD3s-sub	214.1	40.1
Model 3	211.1	40.0
RGB S32s	127.4	25.0

It is obvious from the above table that the spatial domain model, regardless of the version, is the model that takes the least time to response to the user query, this is because processing is applied to images on their spatial representation, and hence, no transformation for images is required. The next model, that needs less time than the remaining models, to response to the user query, is the first one that is proposed to work on images in the frequency domain. This is because comparison is done on images without applying the partitioning procedure as a previous step. Clearly, the use of the indexing algorithm enhances the retrieval time by a high degree on all the proposed models. As an example on the effectiveness of the indexing algorithm, figure 23 shows the ranking results, in locations eight through sixteen, for the query image that was used in figure 16. We used 'RGB db8A3sw' in order to perform these experiments. The first seven retrieved images are the same as those in figure 16. Therefore, we displayed here the next nine retrieved images. As it is clear, all the returned images are related, in someway, to the submitted image.

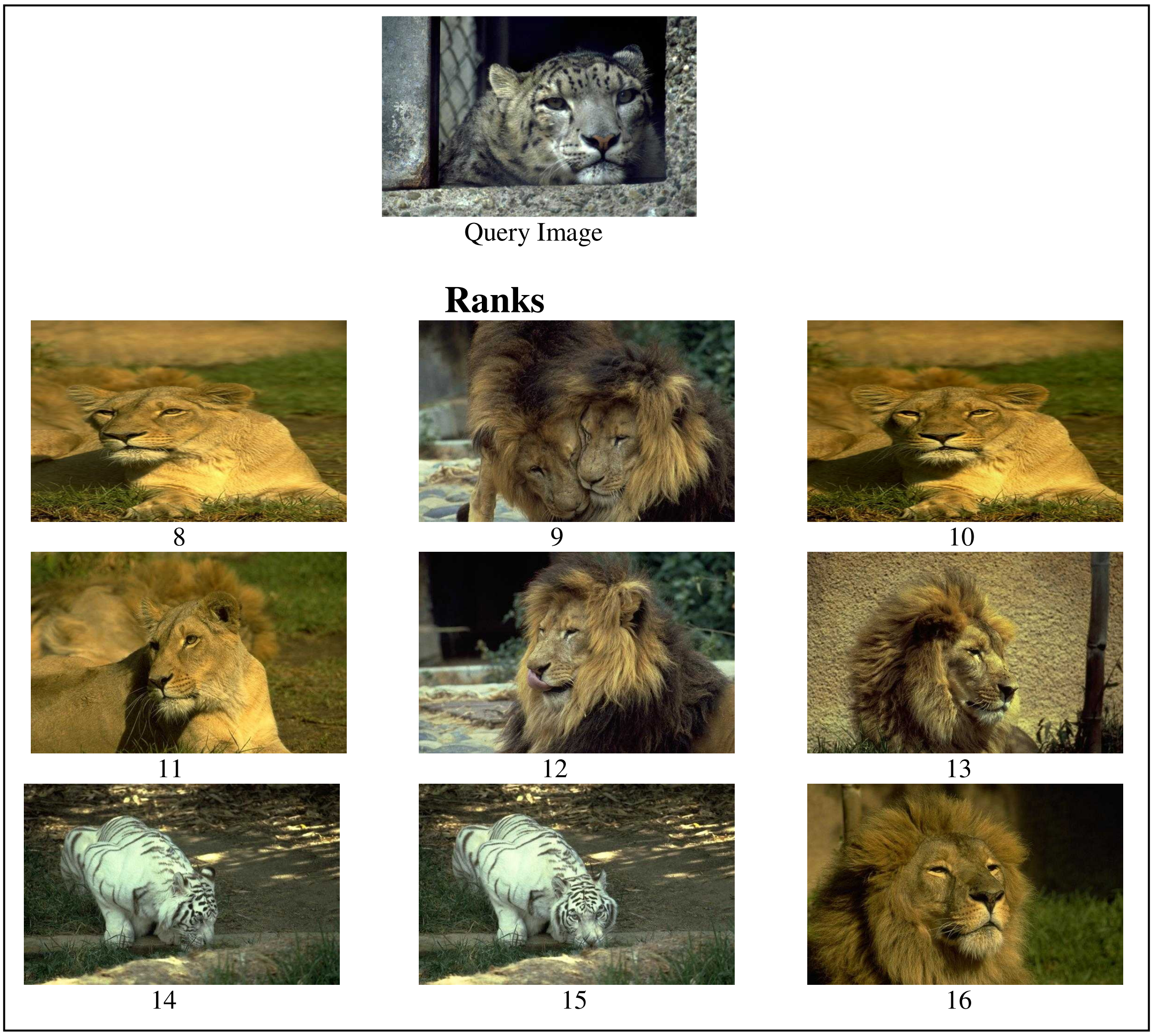


Figure 23 *The ranked images in locations 8 to 16 using the indexing algorithm applied on 'RGB db8A3sw'*

Conclusions and Future Work

This research has achieved very commendable results when compared with results obtained by other systems. Our proposed methods differ from other systems in making a clear distinction between target-specific queries search and category queries search which discriminate our models from others. A more discriminating sign of our models is that, the retrieval accuracy of target specific searches that is achieved by our models provides significantly improved results. In addition, we are working on color-texture template matching basis. This corresponds to consider both color and texture information to represent the image content, as opposed to many other approaches which consider color and texture as separate features.

In this thesis, we have presented and discussed four models for improving the accuracy of the CBIR. We have investigated and analyzed wavelet transform for the models that work on images on their frequency-domain representations. In addition, Cross-Correlation is investigated as the similarity measure on all the proposed models. This thesis has shown that the proposed models are very effective for CBIR in general, and specifically, for the target-specific queries search. We have examined and compared several versions of the proposed models. The evaluation of the models is carried out on a general-purpose image collection. Our models are achieved very promising results. We strongly believe that applying our models on a highly specialized database such as medical images database or video sequences database will achieve better results.

1 CBIR Models

Recent years have witnessed a huge growth of digital image collections, which motivates the research of image retrieval. Such CBIR methods like those proposed in this thesis, could assist on finding the desired images on short time and improve the retrieval accuracy. The proposed models are classified, according to the form of the images representations that they work on, into frequency-domain models and spatial-domain models. Our proposed techniques contributions, and their advantages and disadvantages are summarized in the next sections. Moreover, the proposed indexing algorithm is summarized also.

1.1 Frequency-Domain Models

We proposed three models to work on images on their spatial-frequency representations using the wavelet analysis functions.

1.1.1 Image Retrieval using Cross-Correlation on approximation coefficients of Daubechies' Wavelets

- The purpose of this technique is to retrieve images that are semantically similar to the query image in the set of the best matching images.
- Images are first preprocessed by rescaling them to a common smaller size. Also, histogram equalization is applied.
- Wavelet transform is applied on images and the approximation coefficients are returned as the images representations. Images are decomposed to two different levels: 2 and 3, to investigate which level yields to better retrieval accuracies.
- Reference vectors are formed from the processing on the query image itself.
- In the matching stage, cross-correlation is applied on the representation of a database image and the representation of the query image; this is applied on the three color planes. Matching vectors are formed from the resulted Cross-Correlation matrices. This process is applied for all the images in the database.
- Best results achieved when applying three levels of "Daubechies-8" and "Symmlet-8" wavelet decomposition functions.
- The main drawback of this model is time consuming on conventional machine.

1.1.2 Image Retrieval using Cross-Correlation on the coefficients of Daubechies' Wavelets on the basis of sub-image matching

- The purpose of this model is the same purpose for which the previous model is proposed.
- Images are preprocessed by rescaling them to a new smaller size, converting them to their gray-scale representations, and decomposing each image into a set of patches covering the entire image.
- Wavelet transform is applied on each partition of the query image as well as for the database images. The decomposition is applied to three levels. The representation of images is formed in two different strategies. In the first strategy, only the approximation coefficients of the third level of decomposition are returned. In the second strategy, the approximation and details information of that level are returned. In addition "Daubechies-8" and "Symmlet-8" wavelets decomposition functions are applied to investigate which one gives more accurate results.
- Reference vectors are formed from the representations of the partitions of the query image, with one reference vector for each partition.
- In the matching stage, Cross-Correlation function is applied between the representation of each partition of the database image with its counterpart from the query image.
- Best results are achieved with the extraction of the approximation and details information of the third level of wavelet decomposition.
- The main drawback of this model is the time needed to response to a user query.

1.1.3 Image Retrieval using Cross-Correlation on the reconstructed images on the basis of sub-image matching

- In this model, the same purpose and steps of processing used in the previous model are used here. The main difference between the two models is in the representations of images that are used by the matching step.
- After preprocessing, wavelet decomposition is applied on the images. Then, reconstruction of the partitions from, only, their second level of decomposition is performed. This strategy is applied on the query image partitions as well as on the database images partitions.
- This technique was efficient when used on target-specific queries type. However, on category queries, it is the model that gives the least high ranking percentage results. This is because when reconstructing from the second level of decomposition only, we ignore part of the information, which seems to be significant.

1.2 Spatial-Domain Models

We propose a model, with its variations, to work on images on their spatial-domain representations.

1.2.1 Image Retrieval using Cross-Correlation applied on spatial representation of sub-images

- In this model, two branches for the preprocessing stage of the images are investigated. In the first branch, images are rescaled to a new smaller size, and then, each image is decomposed into a set of patches covering the entire image.

- In the second branch, the same preprocessing steps of the previous approach are used, except that; after rescaling the images, they are converted to their intensity representations.
- Considering the first approach of preprocessing, all other steps are the same as model 2, except that comparisons are done on images partitions on their spatial-domain representations.
- When using the second branch of the preprocessing stage, the remaining steps are similar to those used by model 3.
- Best results are achieved with the one that used the first preprocessing stage in some categories. While for other categories, best results are achieved using the second branch of the preprocessing stage.
- The main drawback of this model is the time required to answer the user query.

1.3 The Indexing Algorithm

- The proposed algorithm works with all the models, to enhance the retrieval time needed to process the user query. In addition, all the retrieved images are similar, in someway, to the query image.
- A feedforward network using the backpropagation algorithm is used to classify images into different predefined categories.
- Once an image is requested, only the category it belongs to, is searched, which minimizes the waiting time needed to process the user request.
- The main drawback of this algorithm is that it requires a huge amount of data to train the network in order to find the right category of a non-trained query image.

2 Future Works

CBIR at present is still very much a research topic. Like most image retrieval researches, this thesis is widely open for improvement. Since large collections of digital images are being created in many fields of life such as commerce, government, and hospitals, research will and should continue into the future. As a future work, we recommend the following:

- Enhancing the retrieval time needed to process the user query. Although the use of the indexing algorithm enhanced the retrieval time, a further improvement is required on the time needed to process a user query.
- More data images to train the feedforward network of the indexing algorithm. Also, using more images to test the models.
- Investigating the use of strong segmentation on images instead of using the partitioning strategy.
- Developing the system to retrieve images that contain objects, but at different scales, like those that exist in the query image.
- Including the relevance feedback strategy in the system in order to enhance the interaction between the user and the system.
- Finally, doing experiments with our models on a highly specialized database such as video sequences database could be another interesting study.

REFERENCES

(Ardizzoni et al., 1999) Stefania Ardizzoni, Ilaria Bartolni, and Marco Patella. (1999). Windsurf: Region-Based Image Retrieval Using Wavelets, **Proceedings of the 10th International Workshop on Database & Expert Systems Applications IWOSS'99**, Florence, Italy. 01-03 September, 1999, 167-173.

(Armitage and Enser, 1997) L. Armitage and P. Enser. (1997). Analysis of User Need in Image Archives. **Information Science.** 23(4), 287-299.

(Bourke, 1996) Paul Bourke. (August 1996). Cross-Correlation. Retrieved April 19, 2004, from **http://astronomy.swin.edu.au/~pbourke/analysis/correlate/**

(Chang et al., 1997) S.-F. Chang, J.R. Smith, M.Beigi, and A. Benitez, (1997). Visual Information Retrieval from Large Distributed Repositories. **Communication ACM**. 40(12), 63-71.

(Chang and Kuo, 1993) Tianhorng Chang and C.-C. Jay Kuo. (October 1993). Texture analysis and classification with tree-structured wavelet transform. **IEEE Transactions on Image Processing**, 2(4), 429-441.

(Cox et al., 2000) Ingemar J. Cox, Matthew L. Miller, Thomas P. Minka, Thomas Papathomas, and Peter N. Yianilos. (2000). The Bayesian image retrieval system, PicHunter: Theory, implementation, and psychophysical experiments. **IEEE Transactions on Image Processing**. 9(1), 20-37.

(Eakins and Graham, 1999) John P Eaknis and Margaret E. Graham. (1999). Content-based Image Retrieval, A report to the JISC Technology Applications Programme (Electronic version). **Technical report, Institute for Image Data Research, University of Northumbria,** Newcastle, UK. (January 1999) Available at: **http://www.unn.ac.uk/iidr/report.html.**

(Fauzi and Lewis, 2003) Mohammad F.A. Fauzi and Pual H. Lewis. (2003). Texture-based Image Retrieval Using Multiscale Sub-image Matching. **In Proceedings of IS&T/SPIE's 15th Annual Symposium Electronic Imaging: Image and Video Communications and Processing 2003** (**SPIE**). 5022, Santa Clara, California, 407-416

(Flickner et al., 1995) M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. (1995). Query by Image and Video Content: The QBIC System. **IEEE Computer.** , 28(9), 23-32.

(Gonzalez and Woods, 2002) Gonzalez R. C. and Woods R. E. (2002). **Digital Image Processing,** (2nd ed). New Jersey: Prentice-Hall.

(Goodrum, 2000) Abby A. Goodrum. (2000). Image Information Retrieval: An Overview of Current Research. **Informing Science**. 3(2), 63-66.

(Graps, 1995) Amara Graps. (Summer 1995). An Introduction to Wavelets. **IEEE Computational Science and Engineering.** 2(2), pp 50-61.

(Haralick et al., 1973) Robert M. Haralick, K. Shanmugam, and Its'hak Dinstein. (1973). Texture features for image classification. **IEEE Transactions on Systems, Man, and Cybernetics.** SMC-3(6), 610-621.

(Koskela et al., 2001) Markus Koskela, Jorma Laaksonen, and Erkki Oja. (2001). Comparison of techniques for Content-Based Image Retrieval, **Proceedings of the 12th Scandinavian Conference on Image Analysis (SCIA),** Bergen, Norway. 2001, 579-586.

(Lew and Sebe, 2000) M.S. Lew and N. Sebe. (2000). Visual Websearching Using Iconic Queries. **Proceedings IEEE Conference on Computer Vision and Pattern Recognition**, 2, Hilton Head Island, USA. 13-15 June, 2000, 788-789.

(Lewis, 1995) J.P. Lewis. (1995). Fast Template Matching. **Vision Interface.** Retrieved March 21, 2004, from **http://www.idiom.com/~zilla/Work/nvisionInterface/nip.html.**

(Mitchell, 1997) Tom M. Mitchell. (1997). **Machine Learning**. WCB/McGraw-Hill.

(Pass et al., 1996) Greg Pass, Ramin Zabih, and Justin Miller. (1996). Comparing images using color coherence vectors**. Proceedings of the fourth ACM international conference on Multimedia**. Boston, MA, USA. 1996. 65-73.

(Polikar, 2001) Robi Polikar. (January 2001). The Engineer's Ultimate Guide to Wavelet Analysis: The Wavelet Tutorial. **Hosted by Rowan University**, Retrieved March 21, 2004, from **http://users.rowan.edu/~polikar/WAVELETS/WTtutorial.html.**

(Rui et al., 1999) Yong Rui, Michael Ortega, Thomas S. Huang, and Sharad Methrotra. (September 22, 1999). Information Retrieval Beyond The Text Document, **Appeared in Proceedings of Library Trends**, 48(2).

(Rui et al., 1999) Yong Rui, Thomas S. Huang, and Shih-Fu Chang. (March, 1999). Image Retrieval: Current Techniques, Promising Directions and Open Issues**, Journal of Visual Communication and Image Representation**, 10(1), 39-62.

(Sangwine and Horne, 1998) S.J. Sangwine and R.E.N. Horne. (1998). **The Color Image Processing Handbook,** (1st ed). Chapman & Hall.

(Santini, 2001) Simone Santini. (2001). **Exploratory image databases content-based retrieval,** (1st ed). Academic Press.

(Smeulders et al., 2000) Arnold W.M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. (DEC 2000). Content-Based Image Retrieval at the End of the Early Years**. IEEE Transaction on Patten Analysis and Machine Intelligence.** 22(12), 1349-1380.

(Smith and Chang, 1995) John R. Smith and Shih-Fu Chang. (1995). Single color extraction and image query, **Proceedings of the IEEE international conference on Image Processing,** 3, Washington D.C., USA. 23 – 26 October, 1995, 3528-3531.

(Smith and Chang, 1994) John R. Smith and Shih-Fu Chang. (1994). Transform features for texture classification and discrimination in large databases, **Proceedings of the IEEE International Conference on Image Processing (ICIP),** 3, Austin, Texas, USA. November 1994, 407-411.

(Tamura et al., 1978) Hideyuki Tamura, Shunji Mori, and Takashi Yamawaki. (1978). Texture features corresponding to visual perception. **IEEE Transactions on Systems, Man, and Cybernetics.** SMC-8(6), 460-473.

(Tian et al., 2001) Q. Tian, N. Sebe, M.S.Lew, E.Loupias, and T.S.Huang. (October 2001). Content-Based Image Retrieval Using Wavelet-based Salient Points. **Journal of Electronic Imaging**. 10(4), 835-849.

(Valens, 2004) C. Valens. (2004). A Really Friendly Guide to Wavelets. Retrieved March 21, 2004, from **http://perso.wanadoo.fr/polyvalens/clemens/wavelets/wavelets.html#note1.**

(Veltkamp and Tanase, 2002) Utrecht University. (October 28, 2002). Content-Based Image Retrieval Systems: A Survey. A revised and extended version of **Technical Report UU-CS-2000-32.** The Netherlands. Remco C. Veltkamp and Mirela Tanase: Authors.

(Wang et al., 1997) James Ze Wang, Gio Wiederhold, Oscar Firschein, and Sha Xin Wei. (1997). Content-based image indexing and searching using Daubechies' wavelets. **International Journal on Digital Libraries**. 1(4), 311-328.

إسترجاع الصور بالاعتماد على المحتويات

إعداد
نور نصر أحمد العايدي

المشرف
الدكتور سامي سرحان

المشرف المشارك
الدكتور موسى حبيب

ملخص

شهدت السنوات الأخيرة زيادة متسارعة في حجم مجموعات الصور الرقمية. البحث من خلال مكتبة رقمية والتي تحتوي على عدد كبير من الصور الرقمية أو سلسلة متعاقبة من الفيديو هو أمر في غاية الأهمية في هذه الحقبة الزمنية من عمر الصور الرقمية. لجعل بحث من هذا النوع عملي، هناك اهتمام متزايد لإيجاد طرق لترميز الصور بطريقة فعالة وإتمام عملية البحث بالإعتماد على المعنى الذي تحمله الصورة.

تم في هذه الرسالة تصميم، بناء، وتقييم أربعة تقنيات هي: استرجاع الصور باستخدام (cross-correlation) على المعاملات المقربة الخاصة بمويجات (Daubechies)، استرجاع الصور باستخدام (cross-correlation) على معاملات مويجات (Daubechies) بالاعتماد على تطابق أجزاء الصور، استرجاع الصور باستخدام (cross-correlation) على الصور المعاد بناؤها بالاعتماد على تطابق أجزاء الصور، واسترجاع الصور باستخدام (-cross correlation) مطبقة على التمثيل الفضائي لأجزاء الصور.

كل نظام من هذا النوع يتكون من ثلاث مراحل أساسية هي: مرحلة تحسين الصور، مرحلة تمثيل الصور، ومرحلة مطابقة الصور. في هذه التقنيات، وكخطوة من مرحلة تحسين الصور، تم تقسيم الصور الى مجموعات من الحزم والتي تغطي كل صورة بكاملها. ثلاثة من

هذه التقنيات تعمل على الصور وهي ممثلة في المجال الترددي باستخدام تحليل الصور بطريقة (Wavelet). في حين أن التقنية الرابعة تعمل على الصور في تمثيلها الفضائي. تم استخدام (Cross-Correlation) في مرحلة تطابق الصور على أنه مقياس للتشابه بين الصور. عدة إتجاهات تم فحصها و مقارنتها لكل من التقنيات المذكورة. لتسريع عملية الاسترجاع، تم اقتراح خوارزمية للفهرسة وذلك لتقليص عملية البحث لتكون فقط في نفس الصنف الذي تتبع له الصورة المستعلم عنها.

تقييم هذه التقنيات تم باستخدام مجموعة من الصور متعددة الأغراض. من التجارب، وُجد أن التقنيات المقترحة فعّالة في استرجاع صور تحمل نفس معنى الصورة المستعلم عنها. إضافة إلى ذلك، ولتوضيح أداء التقنيات المقترحة، تم حساب دقة الإسترجاع لكل من حالات الإستعلام عن أهداف محددة وحالات الإستعلام عن صور من نفس الصنف الخاص بالصورة المستعلم عنها لكل تقنية مقترحة، وقد أظهرت الحسابات نتائج واعدة. دقة الاسترجاع للإستعلام عن أهداف محددة كانت أكثر من ٩٥%. بينما لاسترجاع صور من نفس صنف الصورة المستعلم عنها، فقد كانت دقة الإسترجاع تعتمد بشكل كلي على التفاصيل المحددة الخاصة بكل صورة لوحدها. الهدف من هذا النظام هو استرجاع الصور والتي تحمل معنى مشابه للصورة المستعلم عنها في مجموعة أفضل صور مطابقة. نعتقد، وبشدة، أن تطبيق هذه التقنيات على قاعدة بيانات متخصصة مثل قاعدة بيانات خاصة بالصور الطبية أوسلسلة متعاقبة من أشرطة الفيديو ستؤدي الى الحصول على نتائج أفضل.