Optimal Feature Selection and Fusion for Content-Based Image Retrieval Using Computational Intelligence

Submitted by

Sumaira Farid

Reg # 57 FBAS/PHDCS/F09

Supervisor: Dr. Ayyaz Hussain

Department of Computer Science and Software Engineering,
Faculty of Basic & Applied Sciences,

International Islamic University, Islamabad, Pakistan

(2016)

International Islamic University Islamabad

Faculty of Basic & Applied Sciences

Department of Computer Science and Software Engineering

Final Approval

Dated: 22/02/2016

It is certified that we have read the thesis, entitled "**Optimal Feature Selection and Fusion for Content-Based Image Retrieval Using Computational Intelligence**" submitted by Ms. Sumaira Farid, Reg. No. 57- FBAS/PHDCS/F09. It is our judgment that this thesis is of sufficient standard to warrant its acceptance by the International Islamic University Islamabad for PhD Degree in Computer Science.

PROJECT EVALUATION COMMITTEE

External Examiners:

Dr. Hassan Mujtaba Kayani, Associate Professor,
National University of Computing and Emerging Sciences,
NUCES, FAST Islamabad.

Dr. Zahoor Jan, Chairman
Department of Computer Science,
Islamia College Peshawar.

Internal Examiner:

Dr. Ali Daud, Assistant Professor
Department of Computer Science & Software Engineering
International Islamic University
Islamabad Pakistan

Supervisor:

Dr. Ayyaz Hussain, Assistant Professor
Department of Computer Science & Software Engineering
International Islamic University
Islamabad Pakistan

International Islamic University Islamabad

Faculty of Basic & Applied Sciences

Department of Computer Science and Software Engineering

Final Approval

Dated: 22/02/2016

It is certified that we have read the thesis, entitled "**Optimal Feature Selection and Fusion for Content-Based Image Retrieval Using Computational Intelligence**" submitted by Ms. Sumaira Farid, Reg. No. 57- FBAS/PHDCS/F09. It is our judgment that this thesis is of sufficient standard to warrant its acceptance by the International Islamic University Islamabad for PhD Degree in Computer Science.

Head of the Department

Dr. Hussnain Abbas Naqvi

Department of Computer Science and Software Engineering,

Faculty of Basic and Applied Sciences,

Dean

Prof Dr Muhammed Sher

Faculty of Basic and Applied Sciences,

International Islamic University Islamabad

ACKNOWLEDGMENTS

Inestimable thanks to Almighty ALLAH, whose boundless and unpredictable sources of help have made me able to win honors in life. I am also enormously grateful to my supervisor Dr. Ayyaz Hussain for his advice, valuable comments, and assistance during the course of my study. His keenness and useful views on research have made a deep impression on me. I owe him my sincere appreciation for showing me the way to conduct research.

I would also like to thank my co-supervisor Imad Fakhri for his assistance during my research study. He guided me with expert advice and always provided helpful ideas. His constant encouragement gave me great energy and motivation at every step.

My cordial appreciation extends to Dr. Muhammad Sher for providing constructive suggestions and inspiration. I also owe my thanks to the other faculty members, Dr. Naqvi, Dr. Ali Daud and Dr. Jamal Nasir for their support at various phases of my work.

Finally, without the support and encouragement of my parents, husband, other family members and friends, their understanding and patience, it would have been impossible for me to complete this study. Their never-ending care helped me a lot getting through hard times.

ABSTRACT

Efficient and effective methods are required for the retrieval of relevant data from data stores. It is not only text documents that are kept in these databases, but also non-textual data, which has gained popularity in the field of information management. Non-textual data is mostly comprised of images. These images are saved in clusters and retrieval of required images from these repositories is of great importance. The need for a vigorous method for the retrieval of images from such systems has greatly increased research interest in the field of image retrieval.

Two different approaches used for image retrieval are: (1) text-based and (2) content-based. Text-based image retrieval is based on the keywords provided by users describing the required image. Content-based image retrieval (CBIR), on the other hand, depends on the content of the image, which includes visual features such as color, texture, and shape, and is of great importance in CBIR systems. Content-based image retrieval can be performed at two different levels: the local level and the global level. In the local method, the image is first divided into different parts and then features are extracted from each part separately, while the global technique extracts features from the whole image. We have proposed new versions of both approaches utilizing all vital image features (color, texture, and shape), and promising results were obtained compared to some recent CBIR techniques.

Image features play a crucial role in CBIR systems, leaving a great impact on the retrieval process. Our proposed techniques utilize all the three core image features: color, texture, and shape. Furthermore, Fourier Descriptors (FDs) and Edge Histogram Descriptors (EHDs) were calculated and added to the feature vector, resulting in increased retrieval precision. An important texture feature - Local Binary Patterns (LBP) has also been integrated into the system. The proposed method is based on the identification of several local binary patterns, called uniform local binary patterns. These are indispensable properties of image texture and their histogram has been verified to be a dominant texture feature. A generalized gray-scale and rotation invariant operator was constructed to identify uniform patterns for each spatial resolution and for all

quantization of the angular space. Euclidean distance was used as a similarity measure to find the distance between the query and database images.

Feature fusion and effective relevance feedback methods can contribute extensive benefits to content-based image retrieval. Feature fusion combines different image features in such a way to obtain a single feature vector for all of them; however, combining different image features is not always beneficial. This problem is addressed in this study by acquainting multifarious features, where image features are combined with similarity measures. Genetic programming was also utilized in this study to calculate a single distance measure and the semantic gap was reduced by providing user's feedback after the retrieval of images. This was an iterative process eventually leading to the definition of a set of relevant images. A genetic algorithm based search was also applied to a series of weighting functions, which helped to maximize the fitness function.

The experiments in this study were performed on natural images of diverse semantics from a Corel image database, and showed obvious improvement in results compared to several noble systems in the literature. The same image was selected from each category for experimentation to ensure fair comparison. Precision and recall were used as a metrics to find the retrieval efficiency of the proposed system. The average precision of the proposed methods was greater than 50%, showing improvement in results compared to other current CBIR methods reported in the literature. Finally, comparisons were made between local and global techniques, with both techniques having a reasonable number of image features.

Main contributions of the thesis

- Comparison of the two main approaches to CBIR is made by introducing two different techniques where one is based on the local approach and the other is based on the global approach. Effect of segmentation on CBIR systems has also been evaluated through comparison of local and global techniques
- Image features play an important role in the retrieval process of CBIR systems. It is realized in the study that some image features such as FD and EHD when used in the calculation of feature vector improve the retrieval efficiency of CBIR systems. Uniform local binary patterns are calculated that are the fundamental properties of image texture and are invariant to grays scale and rotation. The operator is further improved by adding a contrast invariant operator jointly used to form a powerful tool for texture analysis.
- Multifarious features are introduced and their effect on the retrieval system has been analyzed through different experiments. Different image features are combined with different similarity measures based on genetic programming technique.
- At times the retrieved results might not be relevant to the query provided by the user. This is mainly because of the semantic gap between the query image and database images. A technique is proposed to reduce the semantic gap between low level image features and high level user's expectations. Genetic algorithm is incorporated to acquire k nearest neighbors of the query given.

Acronyms

CBIR	Content-Based Image Retrieval
FD	Fourier Descriptor
EHD	Edge Histogram Descriptor
LBP	Local Binary Pattern
GA	Genetic Algorithm
GP	Genetic Programming
SVM	Support Vector Machine
ANN	Artificial Neural Network
NN	Neural Network
RF	Relevance Feedback
WWW	World Wide Web
GLCM	Gray Level Co-occurrence Matrix
BAS	Beam Angle Statistics
PCA	Principal Component Analysis
LDA	Linear Discriminant Analysis
ICA	Independent Component Analysis
ACT	Advanced Color and Texture
UM	Universal Model
NFA	Novel Fusion Approach
IRM	Integrated Matching
UFM	Unified Feature Matching
LDP	Local Derivative Pattern
ICLTP	Inter Color Local Ternary Pattern
LTrP	Local Tetra Pattern
MF	Multifarious Features

Table of Contents

List of Tables

List of Figures

Chapter 1

1 Introduction

Content-Based Image Retrieval (CBIR) systems have attained great popularity in recent years, attracting much contemporary research. It is crucial to debate and disseminate a wider understanding of it. Its importance is increasing with the rise of digital data saved in the form of images. These digital image repositories need an efficient system to retrieve required images from them. This chapter introduces the CBIR system and its two distinct approaches with appropriate illustrations from the literature. A brief portrayal of three important techniques of computational intelligence, namely genetic programming, neural networks, and fuzzy logic are also briefly described. There is always a distance between high level user expectations and low level image features, known as a semantic gap, and this requires proper handling to obtain improved results. Relevance feedback is a way of reducing the gap by involving the user in providing feedback to the system. A milieu of the proposed work is given in the end, followed by the objectives of the work done.

1.1 Motivation behind the research

It is a need to create large data sets because of the advances in data storage and image acquisition technologies. The need for keeping these databases is growing because of the increasing amount of digitally produced images in areas like medicine, journalism, and private life. An efficient way is therefore required to manage these databases.

The process of retrieving desired images from a large collection of images (image database) is of great importance in computer vision. Its application in almost every field is increasing for the ease and convenience of users. The process of image retrieval is based on the features that can be automatically extracted from the images themselves.

The main motivation for us behind this research was to propose content based image retrieval system that could work effectively with better precision values for large databases of natural images. We have incorporated an evolutionary computation technique i.e. genetic programming (GP) for the optimal selection and fusion of image features with different similarity measures. Literature shows that GP framework is suitable for the design of effective combinations functions. GP has been successfully employed in different machine learning techniques and can provide better results for pattern recognition in comparison with other classical methods such as support vector machine (SVM).

1.2 Introduction to Content-based image retrieval

Latest developments in data storing and image acquirement methods have created an inevitable demand for large data sets. The need for keeping such databases has emerged due to the growing volume of digitally formed images in fields like medicine, journalism, and private life. A skillful way is needed to manage these databases [1]. The practice of retrieving required images from a huge collection (image database) is of great importance in the computer world. Its application in almost every field is increasing to the convenience of users. The process of image retrieval is built on certain image features that can be automatically extracted from the images. These systems are thus named as CBIR (Content-based Image Retrieval) systems and are of great consideration in the work of image data retrieval. CBIR systems have three main tasks: extraction, selection, and classification [2]. In a large collection of images, it is possible that duplicates of images exist. They would have similarity in content and share some regions with possibly different colors, contrast, shape, and position. These similar images are called image families. Object recognition and image retrieval are very important in such images [3].

With the advancement of the World Wide Web (WWW), developments in computer technology and information bursts in software have created a large number of digital data archives in various fields of life including education, entertainment, military, commerce, biomedicine, web image organization, and image searching [4]. For example, CBIR is of great importance in the medical field. Doctors access large numbers of images daily. These images, being kept in large databases, require a quick and efficient retrieval method. [5]. People have also developed their own personal databases where they keep thousands of images [6]. Journalism is another field of life in which image saving and retrieving is of great importance. Thus, there is a need in this field for new image searching techniques that produce more efficient and accurate results [7]. There are also certain websites that share photos. This gives ease to users in storing images and to others in accessing and viewing the available images. Flickr and Google Picassa are examples of such sites where photo sharing can be done [8].

Many technologies are being established for quick indexing retrieval and management of digital images. Traditional methods for image retrieval comprise of text-based image retrieval methods; however, their performance is not optimum because of the inaccurate language used to describe image content by humans. Also, manual gloss of images is a time consuming task because of large database sizes. As a solution to such problems, CBIR systems were introduced in the early 1990s. CBIR greatly augments the accuracy of information retrieval and is an important substitute to traditional text-based image retrieval [9].

Two different approaches towards CBIR include the discrete method and the continuous method. The discrete method is stimulated by retrieval of textual information. It needs all the features to be drawn to discrete features. The existence of each feature of an image is treated just like the existence of a word in a text document. Systems that follow this approach are VIPER and GIFT. The continuous approach to CBIR is inspired by the concept of nearest neighbor arrangement. Every image is characterized by a feature vector and these are compared by using several distance methods. Images that have minimum distances have maximum scores in image retrieval practice. A comparison of these approaches has been presented in [10]. The discrete approach has an advantage in the sense that in a method based on textual information retrieval can easily be transmitted, e.g., user interaction and storage management. However, most image retrieval methods follow the continuous approach. The continuous approach has strong relations with nearest neighbor classification in pattern recognition. The relationship between general pattern recognition, classification, and image retrieval has been investigated in [11, 12].

The visual features of an image are used to illustrate the image's content in CBIR. These features can either be local or global. Typical global features include shape, color, and texture features, while for local features, extraction is done on the local part after image segmentation. In medical and satellite images, segmentation is a difficult task as there is no clear boundary in such images [13]. It should be noted that the major problem in the field of image retrieval is determining the similarity of two images. Color, texture, and shape are the properties that are most commonly used to measure similarity in these CBIR systems [14].

1.3 Computational Intelligence Techniques

Computational intelligence techniques are nature-inspired techniques dealing with the multifarious problems of the real world. Computational intelligence includes fuzzy logic, neural networks, and evolutionary computations.

1.3.1 Evolutionary Computations

Evolutionary computation is a form of simulated intelligence. It deals with the problem of finding an optimal object from a finite set of objects. This technique is inspired by biological evolutionary processes. Evolutionary algorithms are an important subset of evolutionary computation. Genetic algorithms (GA) and genetic programming (GP) are the sub fields of evolutionary computation. In both these subfields, a population of individuals is taken and the fittest among them are found. This group is selected for the next generation, and hence in the end gives an optimized solution for the given problem. Cross over, transmutation, and recombination are the techniques by which assortment in population is achieved. Selection is the process that selects the best individual for the next generation. GA provides a complete adaptive search procedure that is based on natural selection. It has been employed successfully to put into practice feature selection and weighting on different dissimilar functions [15]. GP has also been successfully employed in different machine learning techniques and can yield better results for pattern recognition in comparison with other classical methods such as the Support Vector Machine (SVM). The key difference between GP and GA lies in their data structure or internal representation of an individual. Generally, individuals in GA applications are of fixed length/bit string or fixed length order of real numbers. However, in GP, more complex data structures are used; for example, trees, stacks, or linked lists.

1.3.2 Fuzzy Logic

Fuzzy logic was introduced in 1965 by Lotfi Al Zadeh with the application of fuzzy set theory. Fuzzy logic is a multivalued logic and has the ability to deal with approximate values rather than fixed and absolute values. It avoids crisp decisions by deciding something between 0 and 1, true and false. It uses incessant variables, not discrete. The concept of fuzzy set membership theory is used in fuzzy sets, which determines how much a variable is in a set. Fuzzy membership functions are premeditated and used in different problems. These functions either take fixed and static values for all cases or are adaptive in nature. The comprehensive concept of fuzzy logic is to cope with the idea of partial truth, where the value of truth can be entirely true or entirely false. It is further added that when semantic variables are used, these can be controlled by certain functions. Fuzzy logic and probabilistic logic are statistically corresponding. Both logics have true values between 0 and 1, but theoretically they are not similar due to different elucidation. Fuzzy logic matches to "degrees of truth", whereas probabilistic logic relates to "probability or likelihood". Fuzzy logic is a computing method that depends on the concepts of fuzzy set theory. It has four main components: knowledge base, fuzzification interface, inference engine, and defuzzification interface. The knowledge base consists of fuzzy rules and a database, which have definitions of linguistic terms for the input and output variable. The fuzzification interface takes the fixed input value and transforms that into fuzzy value and the defuzzification interface calculates the output value by combining the output of rules and performing transformations [16].

1.3.3 Neural Networks

Artificial neural networks are scientific models that are stimulated by genetic neural networks. This technique is used in the fields of optimization, classification, and control theory, and for finding solutions for waning problems. Neural networks are very active for classification problems where they are required to detect and recognize objects effectively. A neural network is a combination of artificial neurons that are interconnected in the form of groups. Information is processed through these neurons. They are designed into many layers. The first layer is the input layer and the last layer is the output layer. There are many layers in between these two layers; these are called concealed layers. The more hidden layers in a system, the more the system is convoluted. Weights are assigned to the interconnections between different neurons. The success of a neural network lies in its vibrant nature, which is consummated by adjusting weights based on the final output and applied input data. Weights are adjusted iteratively till the desired output is attained. The process of adjusting weights in networks is called 'learning of the neural network' [17].

1.4 Semantic gap and relevance feedback

Regardless of the indispensable information acquired from database images, results presented by CBIR systems are flawed. This is mainly because of the minimum association between high level image semantics and low level image features. This absence of correspondence between user expectations and low level image features is called the semantic gap. The problem of the semantic gap in image retrieval is caused by assuming that all image features are equally related during the process of retrieving relevant images from the database. However, some features of images can be germane for certain query images but irrelevant for other query images. Furthermore, some features are deprived of semantic meaning, while others are good in capturing semantics in a given context of CBIR [15].

Relevance feedback methods have been suggested to diminish these semantic gap problems in CBIR systems. RF methods provide a mechanism for CBIR systems to learn which features are best able to capture users' interest [15].

General steps for RF in CBIR are as follows:

- Results are displayed to the user based on the query provided.
- The user provides a decision on the results displayed as to whether and to what extent they are relevant to the query given to the system.
- System gets user's feedback and repeats the second step [18].

RF becomes a classification problem when both positive and negative examples are presented. Such methods have a small sample where the number of training images is small compared to the size of the feature space. It is difficult to get significant results for such small sample sizes from some of the existing learning machines, e.g., support vector machines. RF algorithms should be quick enough to complete the required computations on the data set, as the users' interaction with the machine is in real time.

A number of techniques based on relevance feedback have been suggested using different hypotheses and problem settings. These techniques can be separated into two main classes: i. wraps different aspects of user behavior, ii. include algorithmic assumptions. Regarding the first approach, a user might not be looking for a particular image or target, but the positive rank given by the user indicates that the image is closer to the query image and is not the actual target. It is difficult to produce the best set of images for a large database. Some RF algorithms presume that the user will give binary feedback for both positive and negative examples. Some will take only positive examples and some will take both negative and positive examples with a degree of relevance or irrelevance. The "second approach" takes image features into account. Important features are selected by the implementation of various algorithms such as SVM. Different data structures are also adopted for the selection of relatively important features [18].

1.5 Problem formulation

A CBIR system usually involves the creation of image descriptor, which is comprised of two main steps (i) construction of feature vector transforming image features through an extraction algorithm; and (ii) comparison of two images using a similarity measure. The similarity measure is a distance function that provides the extent of similarity between the two images as characterized by their feature vectors.

An image $\hat{I}$ is defined as a pair $(\mathrm{P}_I, \vec{I})$ where: P_I is a set of pixels and $\vec{I}$: $\mathrm{P}_I \rightarrow \mathrm{R}^n$ is a function that allocates a vector $\vec{I}\,(p) \in \mathrm{R}^n$ to each pixel p in P_I (e.g. , $\vec{I}\,(p) \rightarrow \mathrm{R}^3$ when a pixel is assigned a color in RGB classification). The feature vector $\vec{V}_{\hat{I}}$ of an image $\hat{I}$ is defined as a point in R^n space and can be represented by $\vec{V}_{\hat{I}} = (v_1, v_2, v_3, \ldots\ldots, v_n)$ for an n dimension vector. Features from an image are extracted by $\varepsilon_D : \{\hat{I}\} \rightarrow \mathrm{R}^n$, which is a function that extracts a feature vector $\vec{V}_{\hat{I}}$ from an image $\hat{I}$ and $S_D : \mathrm{R}^n \times \mathrm{R}^n \rightarrow \mathrm{R}$ is a similarity function that calculates distance between two images. Query image Q given to the system is specified as $\mathrm{Q} = \{(G_q, C_q, F_q)\}$, where $G_q = ((V_q, E_q), L_q, N_q)$ is a directed graph having V_q vertices, E_q edges, L_q labels and N_q labeling function on the graph. C_q contain all the digital objects and F_q is a mapping function $F_q : V_q \rightarrow C_q$ [97].

A database or collection $IMG_{collection}$ of images is given as$(Collection, set_{id})$, where $Collection$ is a collection of digital objects or images and set_{id} is a set of image descriptors D.

The aim of this study is to construct an image descriptor D given as D (ε_D, S_D) by fusing features with different similarity measures.

$$D = (\varepsilon_{D1}, S_{D1}) + (\varepsilon_{D2}, S_{D2}) + (\varepsilon_{D3}, S_{D3}) + \cdots + (\varepsilon_{Dn}, S_{Dn})$$

Where, $\varepsilon_D = \varepsilon_{D1}, \varepsilon_{D2}, \varepsilon_{D3}, \dots, \varepsilon_{DN}$ refers to N different image features and $S_D = S_{D1}, S_{D2}, S_{D3}, \dots, S_{DM}$ represent M different distance measures. The searching method is a set of searching states $\{sc_1, sc_2, sc_3, \dots, sc_t\}$, where each state is a series of actions, and each action is related with function Fn_s defined as follows:

$$Fn_s : (\mathrm{Q} \times Collection) \times sim_s \rightarrow 2^{\mathrm{C}}$$

Where $sim_s = Fn_q(q, obj) | q \in Q \ , \ obj \in Collection$, and where $Fn_q : Q \times Collection \rightarrow \mathrm{R}$ is a matching function that associates a real number with $q \in \mathrm{Q}$ and a digital object $obj \in Collection$. Range of the function Fn_s is the C (contents) associated with the collection$IMG_{collection}$. The retrieved results are considered as a subset of C (contents). The computation of Fn_q relies on the use of appropriate image descriptors (e.g. their extraction and distance computation algorithms) defined in the image collection$IMG_{collection}$.

1.5.1 Objectives

The purpose of this study is to investigate the impact of image descriptors on the process of image retrieval. The detailed objectives include carrying out research on three main problems in the field of CBIR, which are 1) improve the quality of feature vector by incorporating maximum number of image features into it, 2) combining image descriptors with different similarity measures, 3) reduce the semantic gap by integration of user's feedback.

The first objective is to consider the number of image features from the three main categories like color, texture and shape. The greater the number of features, the richer will be the feature vector and the larger will be the amount of information extracted from the image.

The second objective is to improve the retrieval performance of the system by fusion of image features with different similarity measures. Same similarity measure used for different types of image descriptors might not give good results, as some descriptors when measured by one distance measure gives better results compared with the other distance measures.

Performance of the system will be further improved by incorporating user's feedback into the system. There is a gap between low level image features and high level user's prospects which can be reduced by getting user's feedback to produce the desired output. User's feedback can improve the accuracy of the CBIR system by labeling images as positive or negative based on user's requirement.

1.6 Context of proposed work

In this study, a CBIR system is proposed utilizing all key image features such as shape, texture, and color, to enhance retrieval accuracy and efficiency. In addition to drawing on the three main categories of image features, Edge Histogram Descriptors (EHDs) were also calculated to extract maximum information for accurate retrieval. Feature extraction was done at both the local as well as the global level to find the finest approach of CBIR. Feature vector calculation in the local approach was conducted after segmenting an image into different parts. In order to avoid loss of information, the image was segmented based on color content. The proposed system is designed for collection of natural images. After extraction of image features and calculation of the feature vector, similarity functions were applied to determine the distance between the query image and the database images. Using a similar function for all image features may not produce effective results as different features produce different outputs with different similarity functions [19]. In order to deal with the problem of using the same similarity measure for all image features, multifarious features were employed. Each image feature was amalgamated with a different similarity measure and the final distance was calculated by implementing genetic programming. Retrieval results were further improved by reducing the semantic gap through adding user's feedback to the system.

1.7 Hypothesis

The research conducted on CBIR in the past decade touched several aspects of the problem. Literature shows that image features is a major issue in designing such systems. Also reduction of the semantic gap by integrating user's involvement has taken interest of many researchers. In this study we propose to show better results for image retrieval by constructing a feature vector that incorporates all the important features, that describes image's content. Features fusion will be performed by constructing image descriptors having different distance measures for different features. We will be using genetic programming approach for constructing such image descriptors and finding the final distance between each pain of images. Relevance feedback will be added in the system to minimize the semantic gap and improve retrieval accuracy of the system.

Our hypothesis is that selection of significant image features and their fusion with appropriate distance metrics might produce better results for CBIR. Reduction of semantic gap will also result in improved accuracy by giving an average precision greater than 50%.

1.8 Organization of the thesis

The thesis is presented as follows. In Chapter 2, a general overview to the CBIR system is presented. Different image features are deliberated in detail along with brief explanations of some of the modern methods of CBIR in the literature.

In Chapter 3, two different approaches to CBIR - the local approach and the global approach - are described. Different features used in the proposed system are defined along with the image segmentation technique adopted for the local approach. This section further explains the calculation of the feature vector and recovery of images through a similarity function's value. The results of two proposed techniques will be matched with several recognized CBIR techniques.

Chapter 4 proposes a CBIR technique incorporating a vital texture feature, i.e., local binary patterns. In this technique, uniform local binary patterns are used, which are invariant to grays scale and rotation. They are jointly used with a contrast invariant operator producing improved retrieval results in comparison with other CBIR techniques.

Chapter 5 focuses on the formation of multifarious features where a combination of image features and a similarity function is defined. Three different types of such features are briefly described in this chapter. The genetic programming technique used for the calculation of the final distance between these features is also explained. The chapter further explains the relevance feedback method incorporated in the proposed CBIR technique. Comparison of the results is also given attention towards the end of the chapter.

Finally, Chapter 6 will discuss conclusions drawn from the investigations conducted in this thesis. Suggestions for further improvement in CBIR systems and further enhancement of the proposed system will also be described in this chapter.

Chapter 2

2 Content-based Image Retrieval Systems

CBIR systems are an application of computer vision processes to the problem of image retrieval, i.e., searching for digital images in huge databases. The search evaluates the image's content rather than the keywords, descriptions, or tags related to the image. CBIR is appropriate for image retrieval because image searches based on metadata are completely dependent on the quality, correctness, and completeness of the annotated text. Also, the process of annotating data manually by humans is time consuming and this might skip the keywords required for image description.

The word "content" in this perspective might belong to color, shape, texture, or other data that can be taken out from the image under consideration. This chapter introduces all these basic image features as well as the different techniques used to extract these features from images to create a feature vector. In CBIR systems, the distance between two images, i.e., the query and the database image, is calculated through a certain distance measure. These similarity measures play a vital role in retrieving the required image from an image repository. Some of the commonly used distance measures in image retrieval systems are given here with their formulas and brief description.

2.1 Overview of Content-based Image Retrieval

In content-based image retrieval (CBIR) systems, a feature vector is produced for every image describing its content and this is stored as an index in the feature database. Hence, images are automatically indexed through this process. For the retrieval of an image from a database, resemblance between the feature vector of the query image and the database image is calculated. A threshold is specified and the distance is measured between the enquiry image and the database image. This space must be less than the specified threshold [2]. Figure 2.1 shows a flow chart of CBIR. There are two main methods to define similarity between the query image and any target image in the image database. This similarity comparison is either performed globally or locally. The global feature descriptor looks at the whole image to find its different features, such as color, shape, texture, or Gabor filter features. However, global features have some downsides associated with them. Firstly, important information about spatial feature distribution is not available. Secondly, there is sensitivity to variations, intensity, and distortion. Spatial restrictions are removed by a color correlogram, color coherence vector [20, 21], spatial color histogram [22], and a spatial chromatic histogram [23].

Local approaches have also been suggested to augment the ability of CBIR. Region based image retrieval methods are extensively used in a number of local feature based methods. Each image is divided into smaller regions and features are taken out from each region. The correspondence of two images is computed on the basis of matching region- based features. The UCSB NeTra system is one region based retrieval system that uses an edge flow method to fragment the image, utilizing three user identified parameters including image features, scale for localization of image boundaries, and estimated number of areas. Segmentation based shape, texture, color, and spatial position are further used to explore and retrieve analogous regions from the database. Local approaches might suffer from information loss due to segmentation which will affect the feature vector calculation process as some of the feature might be lost during the segmentation process [24].

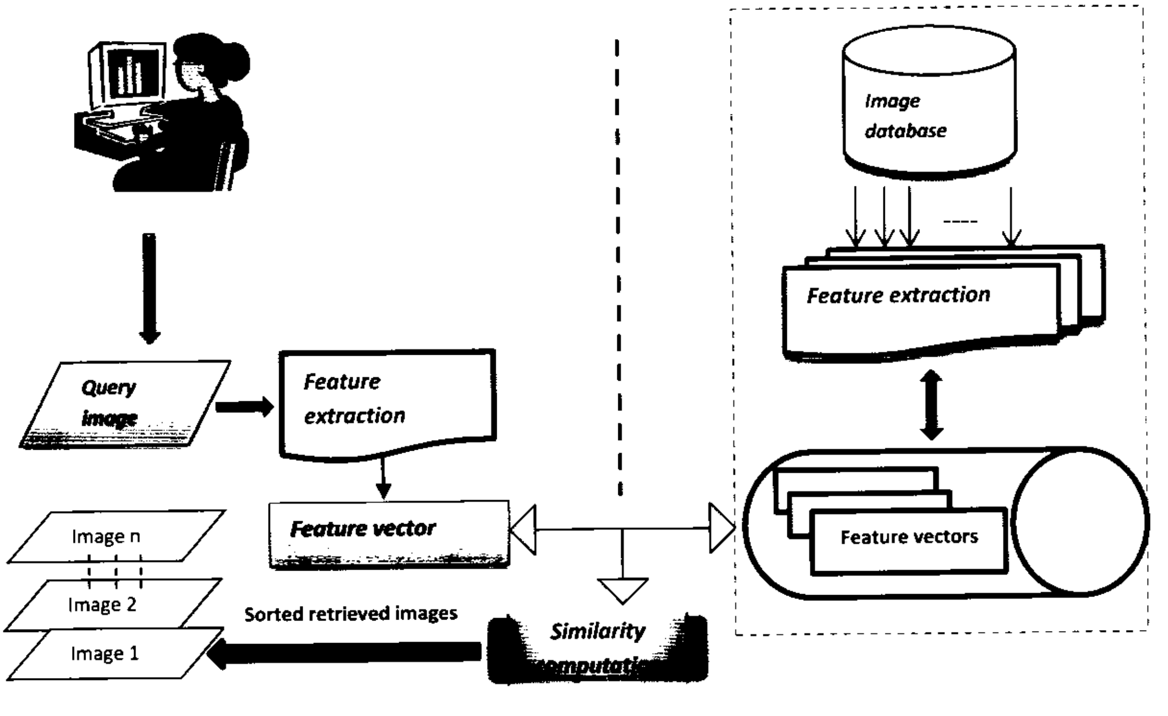


Figure 2-1: Flow chart of content-based image retrieval

The QBIC system from IBM [25] is one of the image retrieval systems that have existed from the start when CBIR systems were introduced. It uses a color histogram, texture descriptor, and moment based shape feature. In QBIC, query given to the system is composed by user graphically. Features extraction is then performed on the graphical image and similarity is calculated with feature vectors of images from the database. Human error can greatly affect the retrieval result [26].

The Photobook system from Massachusetts Institute of Technology (MIT) is also one such system. This system uses texture features, front features, and 2-D shape features. Face recognition technology was also added to this system for the purpose of searching images of particular persons. No text annotation is involved in this system but the number of features adopted is not enough to extract information [27]. Another image retrieval structure is Blobworld [28], established at UC Berkley. Here, images are characterized by regions that are set up in expectation-maximization (EM) similar to the segmentation process. This system follows the continuous approach using the nearest neighbor concept for image retrieval. SIMBA, CIRES, SIMPLICITY, Image Retrieval in Medical Applications (IRMA), and FIRE also use the same approach for image retrieval. In these systems segmentation is the main process before feature extraction and that has to be performed wisely to avoid loss of information. As features play an important role in CBIR systems, the performance of these systems can greatly improve by adopting some recent features discovered in the literature. The VIPER/GIFT system describes the local and global properties of the image by incorporating texture and color features [30]. The structure of the method allows for extremely high dimensional feature spaces (> 80, 000 dimensions) but dynamic features for each image are only 3,000 to 5, 000 [1].

2.2 Features of an image

Images are largely defined by features like color, shape, and texture [32]. Other than these three vital features, there are a few more features of images that also describe the content of images and are helpful in the extraction of information from images. An overview of image features is given in this section.

2.2.1 Color feature

Color is a vital and extensively used representative of an image characterized through some color model. The most commonly used color representations are RGB, YCbCr,and HSV. Various formulas are used with the RGB model to produce YCbCr and HSV models. A histogram is usually utilized for representation of color. It does not provide semantic statistics but is modified to revolve, translate, and scale an object. Thus, two images having similar histograms might differ from a contents point of view. Generally, a color indexing process is applied to retrieve images based on color content. Color moments indicate dissemination of color in an image [2]. Removal of color, competent cataloguing, and sound recovery are criteria for a good method. A number of essential objectives must be satisfied in the system, such as automatic abstraction of color, proficient indexing, and effective retrieval [30]. HSV color space is used to segment an image by the region growing approach. Image content is represented by the shape, color, and position features of all the regions in an image. A probabilistic structure is then employed to model the first order region properties and the second order spatial relationship of all the regions of an image. These relationships are calculated concurrently to evaluate the similarity between two images [32].

A method has been suggested for color indexing which offers for automatic abstraction of local color regions, effectual indexing, and exceptional query performance. Color indexing is carried out using two main techniques: the first is indexing by global color distribution and the second is indexing by local or regional color distribution. Global distribution indexing allows for the comparison of whole images while local distribution facilitates matching between regions confined within images. Color indexing by global distribution has good results when the user is not concerned with the location of color regions or objects in the image. Indexing by local distribution works efficiently when the user is concerned with the discovery of different objects in images. The later method is more complicated as it needs the effective abstraction and demonstration of local regions. Both methods require automated extraction and efficient representation of color. Thus, the feature extraction method comprises of two phases: the first is the extraction of regions and the second is the extraction of region characteristics. The technique used for region extraction partially uses the color histogram and back propagation for comparing images. The purpose of back propagation is to create the most probable position of a spatially confined color histogram within the image on to the image of proportion of the query histogram and histogram of the image [30].

Back propagation images stored in the database are pre-computed with predefined color sets. By treating these back propagations beforehand, best matches are returned at the time of query without calculation. Color features can be represented by mean color, prominent color, or a color histogram. A color histogram is computationally hard and is a high dimensional feature vector. It is more appropriate for global color regions than local color regions. A binary color set is another technique for demonstrating the color content of the image selecting colors that are sufficiently existing [30]. The joined possibilities of three color channels can be demonstrated through color histograms. The color histogram is given by the following equation:

$$hist[r, g, b] = P.Prob\{R = r, G = g, B = b\} \tag{2.1}$$

Where P is used for the number of pixels in the image and R, G, and B are the three color channels. The color histogram is calculated by counting the number of pixels in the image and discretizing the colors in the image. The three channel histogram is transformed to a single variable histogram. The transformation of RGB is given by m = r + Nrg + NrNgb, where Nr, Ng, and Nb are the number of bins for three color channels. This provides a single variable histogram.

$$H[m] = N.Prob\{M = m\} \tag{2.2}$$

The transformed color between RGB and generic color space xyz is denoted by

$$(x, y, z) = t\,(r, g, b) \tag{2.3}$$

There are four stages in the extraction of color regions and their features:

1. Transformation, quantization and filtering
2. Back propagation of binary color sets
3. Region labeling and threshold
4. Region feature extraction

The aim is to diminish a certain subset of colors and unimportant color information in the image. Thus prominent color regions are accentuated [30].

The first stage in color set back propagation is color processing. The color image is filtered and down-sampled to eliminate insignificant details, and then the color regions are extracted by methodically filtering over the color set for the image. A bi-level image B[m,n] is then filtered to eradicate noise and connect neighboring regions. B'[m,n] is produced by filtering B[m,n] with a 525 box median filter U[m,n]. The filtered image B'[m,n] is labeled with a unique non-zero label where each label is analyzed with regard to several thresholds. A zero label is consigned to a region not meeting any of the thresholds. After thresholding, colors are assigned the different regions from the color set such that there are not less than a specific number of pixels of that color in an image. The color set is recorded of these regions, and the area and location of each region is also measured. The location of each region is represented by the minimum bounding rectangular (MBR) that hems in the region. A feature table is then constructed to store all the features [30].

Color is considered to be the most effective image feature in almost all CBIR systems because of the advantages of effectiveness, robustness, implementation easiness and low space requirements. However this feature alone might not produce optimal results while retrieving images from large databases as the information retrieved in snot enough, therefore their combination with other features have been tried and proved to produce better results [31].

2.2.2 Texture feature

Texture is yet another vital characteristic used for the classification and recognition of objects. Texture representation methods are basically structural and statistical. Structural features provide information related to the structural arrangement of planes and objects in the image. Texture is an overall characteristic of an image that is not divergent for each image separately but is subject to its intensity distribution of pixels over the image. The texture features of an image include image properties like scalability and periodicity. Statistical features include the overall statistical parameter calculated from the intensity values of pixels. This type of feature also calculates the parameters constructed from the co-occurrence matrix [33]. Widely used techniques for the extraction of texture features are the gray level co-occurrence matrix (GLCM) and Gabor Transform, etc [2].

2.2.2.1 Grey Level Co-Occurrence Matrix

The first and most widely used method for the extraction of texture features is constructing a grey level co-occurrence matrix. This method was proposed by Haralick et al. [37] and 14 different descriptors were suggested by the authors. These descriptors included energy, contrast, correlation, homogeneity, entropy, and others. Each of these descriptors represents one property of texture.

This method finds out the occurrence probability C (i,j | Δx Δy) of a pixel pair with intensity i,j, spacing between pixels Δx and Δy in dimensions x and y. Some of the texture attributes are defined as follows:

Energy: Energy measures the homogeny that ranges from 0 to 1. It finds out the intensity between a pixel and the neighboring pixels over the image. Energy is given by the following equation:

$$Energy = \sum_{i,j} C(i,j)^2 \tag{2.4}$$

Contrast: This measures the intensity contrast between a pixel in the image and its neighbor. It is represented by the following equation:

$$Cont = \sum_{i,j} (i-j)^2 \; C(i,j) \tag{2.5}$$

Where i represent a pixel in the image and j is the neighbor.

Correlation: This finds out how each pixel is correlated to the neighboring pixel over the whole image. It ranges from 1 to -1. Correlation is represented as:

$$Corr = \sum_{i,j} \frac{(i-\mu_x)(j-\mu_y)C(i,j)}{\sqrt{\sigma_x \sigma_y}} \tag{2.6}$$

Where μ is the variance and σ is the standard deviation of pixels in image x, i.e. the query image and image y, i.e. the database image.

Entropy: Entropy converts any class except for logical to unit 8 for the calculation of the histogram count, so the values of pixels are discrete and are directly related to the value of a bin in a histogram. It is defined by the following equation:

$$Entropy = -\sum_{i,j} C(i,j) log C(i,j) \tag{2.7}$$

The co-occurrence matrices are calculated at four different orientations: 0°, 45°, 90°, and 135°, and all these values are extracted from these four matrices [38].

As grey scale is 256, the size of the co-occurrence matrix will be 256 x 256. Humans mostly measure the similarity between two images based on uneven texture features. The grey scale can be compressed to decrease calculations. Therefore, the computational complexity of co-occurrence matrix depends greatly on the number of gray scales used for quantization [2].

Example: A co-occurrence matrix calculates the relative positions of the points with various intensities. A matrix A_{LxL} is generated with dimensions L x L using the constraint generated by the position operator p. Therefore for this matrix A the intensity values ranges from 0 to L-1. The gray levels for this matrix are then given by z_0, z_1, … z_{L-1}.

A particular element suppose a(i,j) of matrix A_{LxL} indicates the number of times points with intensity value z_j occur at a position determined by P relative to points with intensity z_i. So, a point is taken with certain intensity value within the texture image suppose z_i and following the position operator p we come to some other points suppose z_j. The operator a(i,j) indicates that how many time such a pair z_i z_j is indicated by the location operator that appears within the given texture.

Example: Suppose image I with three distinct intensity levels is given by the following matrix

$$I = \begin{bmatrix} 0 & 0 & 0 & 1 & 2 \\ 1 & 1 & 0 & 1 & 1 \\ 2 & 2 & 1 & 0 & 0 \\ 1 & 1 & 0 & 2 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Let P = 1 pixel to right.

For the above matrix there are three intensity levels and are given by $z_i = 0$, $z_1 = 1$ and $z_2 = 2$. Now matrix A will have dimension 3 x 3 and is given as

$$A_{3\times 3} = \begin{bmatrix} 4 & 3 & 1 \\ 3 & 3 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

To get the co-occurrence matrix we need to find the total number of occurrences suppose n of all the pairs within the image and n= 19 for the above image example. Co-occurrence matrix is the division of all the elements of matrix A by n and is given as

$$C = \frac{1}{19}A$$

2.2.2.2 Gabor Transform

Gabor filters have been deployed in image retrieval, texture analysis, and different computer vision applications. The Gaussian function represents the general functionality of two dimensional (2D) Gabor filters. The 2D Gabor filter is represented by the following formula:

$$G(x, y, T, \theta, \partial x, \partial y) = \frac{1}{2\pi\, \partial x\, \partial y}\, \varepsilon \left[-\frac{1}{2}\left(\left(\frac{x}{\partial x}\right)^2 + \left(\frac{y}{\partial y}\right)^2 \right) + k\, T(x \cos\theta + y \sin\theta) \right] \tag{2.8}$$

Where $k=\sqrt{-1}$, ∂x and ∂y are the scaling parameters of the filter, T is the radial frequency of the sinusoid and $\theta \in [0, \pi]$ specifies the orientation of the Gabor filters.

The Gabor filtered output of the image is attained by the involvedness of the image with the Gabor function for each of the spatial frequencies or orientations. Although it has good multi resolution and multi orientation properties but it is lacking of the non-orthonormal bases. However, in feature extraction process non-othonormal properties are not considered [2].

2.2.3 Shape Feature

The shape feature is also one of the core image features used to find similarity between the query image and database images. It is an important and pivotal visual feature for image content description. The methods used for the representation of shape features can be divided into two different groups. One method characterizes the region in terms of its boundary or external characteristics and is known as the external method. Another method represents the region based on its internal characteristics, i.e., pixels forming the region, and is known as the internal method. Shape features comprise of two main classes: region descriptors and boundary descriptors. The former uses the whole area while the latter uses only the information existing in the contour of the object. Circularity, discontinuity, aspect ratio, angle irregularity, complexity, right-angleness, length irregularity, sharpness, and directedness are features of the image's contour.

2.2.3.1 Region Descriptors

Regions are explained as simple geometrical parameters, e.g., an area or density measure. The density measure is a ratio of the square of the perimeter to the area. The density measure is invariant to uniform scale variants and takes its smallest value in a circle shape region.

Grid based method: The grid based method describes the shape of an object. It can be described by two steps. The first step is the superimposition of a grid having cells of an explicit size, and the second step is the transformation of cells of a grid from top to bottom and from left to right.

Moments and their invariants: Moment invariants are presently the most common and broadly used region descriptors. These are based on the calculation of geometrical moments of the function of two variables [33]. Region based shape descriptors utilize a set of Zernike moments calculated within a disc centered at the center of the image. Zernike moments have many advantages such as rotation invariance, robustness, expressiveness, effectiveness, and multilevel representation [2].

Zernike moments calculates the angular frequency only where circular spectral features are not counted. The circular features captured by Zernike moments depend on the number of angular frequencies taken. Suppose the number of Zernike moments used is 36 then only 10 circular features are captures as shape descriptors [34].

Region based methods are good at indentation and other shape defection problems that cannot be handled well by contour based features but these methods are complex to implement [40].

2.2.3.2 Boundary Descriptors

The chain code: This defines the boundary of an object as a series of lines with certain orientation. A chain code is constructed by superimposing the image with the grid. Boundary points are then estimated by the adjacent grid nodes and neighboring nodes are connected by the lines.

Signatures: Signatures illustrate the boundary of a 2-D object through the function of a variable that can be easily described in comparison with the original 2D boundary.

Fourier descriptors: This is one of the most widely used methods for calculating contour parameters. The main idea of these descriptors lies in the presentation of discrete Fourier transform to a signature and using Fourier coefficients acquired as parameters defining the contour [33].

Fourier descriptors showed 100 % retrieval results of rotated shaped and almost 100% correct precise retrieval non- rigid and scaled shapes. The advantage of FD is that it can retrieve shapes with severe indentation that cannot be detected by contour based features.

Boundary of image can substantially change upon scaling. If scaling is performed at large level then the contour based features cannot detect the object and thus fails to retrieve the relevant image. However, FD is not affected by even large scaling. Thus, FD is more vigorous to shape variations [36].

2.2.3.3 BAS: Beam Angle Statistics

BAS are beams that are initiated from a boundary point of an object and are described as lines connecting that point with the remaining boundary points, contributing to the construction of shape descriptors. The angle between a pair of beams is calculated at each point to haul-out the topological arrangement of the boundary [11]. Third order statistics of all the beam angles in a set of neighborhood techniques are then exhausted to define a shape descriptor. BAS is invariant to scale, translation, and rotation, and is insensitive to distortion. BAS transforms 2-D shape information into 1-D, representing all convexities and concavities of shape with nominal distortion, which is based on the reference point with the other points on the boundary. At each point, the angle between a pair of beams is considered with the abstract topological structure of the boundary. The representation avoids leveling to preserve all the obtainable information in the shape [39].

Contour-based features have gained more popularity in literature in comparison with region-based methods. The main reason behind this is human perception, they discriminate shapes based on the contours. Another reason could be the shape contour being the only concern where interior content of shape is not essential. Contour base methods have also some limitations for example scaling and indentation of shapes might affect the retrieval performance [35].

2.2.4 Edge-Based features

Edge shape image features of objects are extracted with the comparative locations of edge pixels. Image features of the original features present in the database are taken out in advance and are kept in the matching table [12]. Original images are perceived from the edge and are skeletonized as a preprocessing step. Irregular sketches of the query are drawn by using input devices as binary lines. Features of the query sketch are extracted by the system and are compared to the image features in the matching table by computing the Euclidean distance in the feature space. Features of the binary line images are extracted. Pixels of interest are taken and on it the number of other edge pixels in each of the eight regions is counted.

$$S_{ix} = \begin{cases} \dfrac{C_{ix}}{C-1} & (C \neq 1) \\ 0 & (C = 1) \end{cases} \tag{2.9}$$

$$f_{ix} = \begin{cases} 0 & (S_{ix} \leq th) \\ 1 & (S_{ix} > th) \end{cases} \tag{2.10}$$

Where C_{ix} is the number of pixels in each region and C is the pixel count. This is normalized and given by $S_i(C \neq 1)$. The thresholding technique is applied to find the value of the binary pattern f_{ix}. The system is checked and proved to be scaled, shift, and rotation invariant. For retrieval, the system computes the Euclidean distance between features of the query image and features in the matching table in 256-dimensional feature space. Then the candidate image is given according to the increasing Euclidean distance. This system has produced better results compared to the conventional method reported by Anil K. Jane et al., which used a histogram of local edge detection. The results have been evaluated in terms of retrieval efficiency [14].

Graphical rough sketches can be used to describe important features of the image such as edge-based features of shape and texture. Effective extraction of edge-based features of shape is difficult. Work has been done in automatic extraction of scale, rotation, and shift invariant image features from an object's edge and application of these features to content-based image retrieval [12].

2.2.5 Local features

Local features are calculated in an image on certain interest points. For this purpose, different forms of interest point detectors like affine covariant features and difference of Gaussian are used. These are used in combination with global feature descriptors, for example, the affine covariant feature detector is used in combination with the SIFT feature descriptor. To compute the similarity between images, four approaches are compared with increasing complexity [3].

Simple: This is the first approach used. Here the comparison between images i and j is S_{ij} and is defined as:

$$S_{ij} = \sum_{k \in image_i} 1\{l_{gk} = j\} \tag{2.11}$$

k is the number of features in an image i. For each feature f_k, a nearest neighbor g_k is taken with label l_{gk} such that it is not in the same image. Finding S_{ij} simply counts the number of neighbors that belong to image j. 1{.} is an indicator function that returns when the expression in the parenthesis is true and false otherwise.

NN-ratio: Here images are further processed with at least five nearest neighbors; these five images have the potential of being similar that is started by an exhaustive nearest neighbor search between the features of images i and j. Here similarity is computed as:

$$S_{ij} = \sum_{k} 1\{d\ (f_{ik}, g_{ik1}) \times \delta_1 \leq d\ (f_{ik}, g_{ik2})\} \tag{2.12}$$

Where d measures the difference between two feature vectors, g_{ik} is the nearest neighbor to f_{ik} and g_{ik2} is the second nearest neighbor, $\delta1>1$ is a constant. A feature fk is counted when the distance to its nearest neighbor is considerably lesser than the distance to its second neighbor $\delta1=1.1$.

Image-aff: Spatial consistency of matched features is checked after getting the neighbors between images i and j. The RANSAC algorithm is used to fill an affine transformation Hij that records locations of features in image i to the corresponding feature in image j. This similarity is:

$$S_{ij} = \sum_{k} 1\{d\left(H_{ij}(x_{ik}), x_{jk}\right) < \delta_2\} \tag{2.13}$$

Where x_{jk} is the location of matching features in image j. This counts the number of features consistent with the calculated affine transformation H_{ij}. $\delta2 = 25$ pixels.

Region-aff: Some regions of the image can undertake different transformations and the similarity measure can be enhanced by considering different affine transforms for different regions in an image. The image is divided into 200×200 pixels overlapping regions with a stride of 100 pixels and a spectrum affine transform H_{ijl} is fitted for each such region. The overall number of features depending on these individual transformations is then counted as:

$$S_{ij} = \sum_{kl} 1 \{d\left(H_{ijl}(x_{ik}), x_{jk}\right) < \delta_3\} \text{ ,} where\ \delta_3 = 10\ pixels \tag{2.14}$$

Local features give more accurate results by giving an average precision greater than 50% when they are used with global features for example affine covariant feature detector are combined with SIFT [3].

2.2.6 Appearance-based feature extraction

Appearance-based feature extraction techniques like principal component analysis (PCA), linear discriminate analysis (LDA), and independent component analysis (ICA) transform images into a lower dimensional subspace. Some methods have shown good results using appearance-based techniques. The image is down sampled into 32×32 or any other lower dimension subspace and comparison is performed using any distance measure. The images are then classified with database images based on a distance measure. A contradiction exists in the literature regarding comparing the performance of appearance-based methods. Some researchers proved PCA as the best feature extraction method while others reported that LDA was the best. Others focused on ICA and showed its results to be better than the other feature extraction methods. Hence, no conclusion has yet been reached regarding the best performing of these dominant feature extraction techniques. Each method has achieved good results on different data sets [41].

"In PCA", a set of d-dimensional training images is arranged in a d×C column data matrix. PCA derives a mapping ω from the original d-dimensional space to a lower dimensional space, $y_i = W^T (x_i - \mu)$,

Where y_i (i=1, 2, 3…C) denotes a d'-dimensional PCA feature vector and μ stands for the mean of all the training images. It is the simplest of the true eigenvector based multi variant analysis. It reveals the internal structure of the data in a way which best explains the variance in the data. The lower dimensional picture that it provides is very informative. A few principle components are used to reduce the dimensionality of transformed data [41].

Another technique of appearance based feature extraction is LDA (Linear Discriminant Analysis). Its objective is to improve PCA by also considering class membership information. PCA is usually considered as more appropriate for the task of data compression and LDA for the task of classification. For better results, PCA and LDA are often combined. PCA is used to trim down the dimensionality of data and LDA is applied in reducing space.

The third most dominant feature extraction technique is independent component analysis (ICA). It is an extension of PCA which tries to minimize higher order dependencies in the training images. It represents non-orthogonal transformations. The most prominent of ICA's algorithms are algorithms Jade, InfoMax, and FastICA. ICA is presented in different architectures. The first is ICA Architecture I (ICA1) and the second is ICA Architecture II (ICA 2). The first one seeks statistically independent basis vectors, while the other one seeks statistically independent features. Both ICA Architectures can be employed either through the transformation matrix of the PCA technique or through PCA feature vectors corresponding to the training images of the FASTICA algorithm. Experimental results have shown that LDA outperforms other techniques in face recognition. All the techniques mentioned above reduce similarity in their results when they are subject to degraded images. Among all these techniques LDA give better results by producing the minimum error rate with most of the tested degradations [41].

2.3 Feature extraction techniques

There are several methods of feature extraction and selection. They have been proposed at different times and are based on different approaches.

2.3.1 Textual annotation

One of the major improvements achieved in image retrieval is the addition of textual information. Results can be improved specifically if query images are semantic in nature, as the retrieval process is difficult when based on just visual information. FIRE is one of the systems proposed that incorporates textual information in addition to visual information. Image retrieval tasks in FIRE can be performed by applying a decision rule that calculates a score for each image in the database.

For a set of positive sample images represented by Q+ and negative sample images represented by Q-, the score S (Q+, Q-, X) can be determined for each database image. X with respect to query image as:

$$S(Q^+,Q^-,X) = \sum_{q \in Q^+} S\,(q\,,W) \;+\; \sum_{q \in Q^-} (1 - S(q,X)) \tag{2.15}$$

In a query (Q+, Q-), images are placed in descending order of score and images with the highest score are returned. As mentioned earlier, textual information is incorporated in FIRE. An existing information retrieval engine is normally used. An alternate of smart-2 retrieval metric is implemented by this engine. The first step is the removal of insignificant function words. A list of 319 words is made called the stop word test. Porter's stemming algorithm is then used to decrease the leftover words. The stemmed words create the index terms that are used to index the text documents given with the image data [42].

One text document is attached to each image in order that textual information may be used in the image retrieval process. In order to find the distance between a query image with text and a database image with attached text, textual information retrieval is queried first using the query text. A list of all relevant text documents is produced based on the textual information retrieved from the database. These documents are categorized by retrieval status values. This value is high for documents like the query and low for dissimilar documents.

The second most important task is automatic annotation. The objective of automatic annotation is to minimize a large number of images into a small number of classes. There are two methods available for this task. The first method is similar to the approach used for medical retrieval tasks, except that in medical retrieval tasks, no textual information is available. The second method is a general object recognition method using a histogram of image patches and discriminative training of the log linear method [42].

There are several limitations of the text annotation method. First of all, it is difficult to append text with large volumes of databases. A lot of labor is required and the text added is based only on human perception. Some image queries cannot even be described properly. Also, these methods can be implemented using one language only [42].

2.3.2 Genetic programming approach

The genetic programming method can be applied in the process of image retrieval in CBIR. It has been used in CBIR because of its attainment in other machine learning applications, e.g., it can deliver better outcomes for pattern recognition compared to classical techniques like support vector machine (SVM). It accepts non-linear combinations of descriptors and images are retrieved based on their object shape. Genetic programming (GP) is applied to generate a composite descriptor and combines the predefined descriptor. The GP technique is then used to combine similar values obtained from each descriptor, integrating them into an effective similarity function [43].

Genetic programming is a simulated intelligence problem solving technique based on the principle of biological inheritance and evolution. Each potential solution is called an individual (chromosome) in a population. More individuals are created by applying genetic transformations such as cross over and mutation for better performing individuals in subsequent generations. Each individual is assigned a fitness value by some fitness function. GP has some key components, which are terminals, functions, fitness functions, reproduction, and mutation.

Combining different descriptors by GP is an iterative process. At first, GP works on a huge population of random combination functions. These combination functions are then calculated based on the relevant information and training from images. Genetic transformation systems such as reproduction, mutation, and crossover are used for the modification of population individuals. The best individuals are selected and copied to the next generation by the reproduction operator. Variation is brought by mutation and crossover operators. Mutation operates on one individual while crossover operates on two individuals.

A CBIR system using GP can be modeled by taking into account an image I as a pair of (D_I, I), where $D_I \subset Z^2$ is a finite set of pixels, and I is a function that assigns to each pixel p in D_I a vector I(p) of values in some arbitrary space D'. GP combines simple descriptors because of three reasons. The first reason is the large search space size for combination functions, the second reason is the previous success of using GP in information retrieval, and the third reason is that no work has been done on applying GP to image retrieval [43].

Genetic programming combines simple descriptors into composite descriptors D_{GP} where $\delta_{D\ GP}$ is a mathematical expression represented as an expression tree. Non-leaf nodes of this tree are numerical operators and leaf nodes are similarity values d_i, i=1,2,.....,k. The overall retrieval process can be divided into two different approaches based on the use of validation sets in the similarity function discovery process. The use of validation sets aims to avoid the effect of over training or over fitting. In a GP based retrieval framework without validation sets, the population starts with a random creation of individuals and this population evolves generation by generation through genetic operations. Each individual in a population is assigned a fitness value by using certain fitness functions. This value is used to select the best individuals. Then, genetic operators are applied to this population for more diverse and better performing individuals. Finally, the best individual - the one with the best performance in the training set - is determined and applied to the test set. GP frameworks with validation sets, however, do not consider this individual with best performance as the best individual as over fitting may occur in the learning phase. To remove this process, the best individuals over the generations are applied to a validation set. These individuals with best average performance in both sets, i.e., training and validation, are selected. The average, albeit, does not ensure that the selected individual has a similar performance in both sets as such a bias is corrected by considering the standard derivation. The GP framework is flexible and can find better similarity functions than the ones obtained from the individual descriptors [43].

The method described in the above paragraph has shown considerable improvements in results due to the incorporation of computational intelligence techniques in the system, but only shape features are used and the system is only applicable on a limited number of databases. Color and texture are important features that might improve its performance [43].

2.3.3 Implementation of Fuzzy Logic

A novel fusion structure for image retrieval has been proposed that is established on regional color and texture features, as well as global and semi global EHDs. The first step of this approach is to segment an image into different regions based on exclusive color features. Image retrieval is then taken based on global and semi global EHDs. Region based fuzzy texture and features are integrated for a region matching color-clustering based method used to segment an image since color features are considered more important than other features. The proposed segment method is good enough for retrieval because in image retrieval accurate segments are not required. The segmentation process starts with the dissection of the image into non-overlapping square image-blocks. The size of each block is selected to be 2×2. Regarding the color features for each image block, K-Means algorithms are used to group these color features into several sets; each group in the image features space relates to one spatial region in the image space. The K-Mean algorithm works iteratively and accommodates the fact that the number of regions in an image is unidentified before segmentation. Suppose there are M blocks M (x_i=1, 2… M) for an image. The aim of K-Means algorithms is to group each of the M blocks into one of the K clusters, whose cluster centers are $x_1, x_2, \ldots, x_k$ and to minimize D_k where segmentation starts at k=2 and stops at i^{th} iteration (i.e., total number of clusters k equals i) if D_i is less than a threshold or $|D_i - D_i - 1|$ is less than another threshold [4].

A set of regions will be obtained as the result of segmentation. Each region is represented by color and texture features. The color feature f^c for each region j is the l-Means cluster center of this region, i.e., the average of color features of all image-blocks in this region.

Here the computational cost of the driving regional representative texture feature is minimal compared to other methods where texture features of all image blocks are calculated and then averaged on the basis of segmentation results. Each regional color and texture feature is then fuzzified and fuzzy matching is applied on these fuzzified features. Any member function with a smooth transition between 0 and 1 can be selected for fuzzification.

It is hard to find an active and proficient fuzzy membership function using appropriate parameters with all types of images. Also, possible dilapidation of the retrieval performance may occur when segmentation becomes very accurate. Therefore, global and semi global EHDs are added to the system. The main advantages of using these EHDs are their important characterizing features. The EHDs are independent of segmentation and are invariant to rotation, translation, and scaling. Four categories of edges - namely horizontal, vertical, diagonal, and non-directional – are utilized to represent edge orientation in sixteen sub-images. The image space is divided into sixteen non-overlapping sub-spaces. The EHD is a total of 5×16 histogram bins. Global and semi global edge histograms are constructed on the basis of the EHDs. The global EHD represents the edge distribution for the entire image and for the semi global, five different clusters are generated [4].

Another method suggested by Hsieh and Grimson represents every image by a group of models and their spatial relations. Each model is described by a set of principal segmented regions. These templates reflect various appearances of an object in different conditions. The combined color, texture, and symmetrical feature dissimilarity between each region and model is calculated to find the visual similarity. Then, this visual similarity pools with relation similarity to measure the global similarity [44].

Fixed and static fuzzy membership functions are used in these systems. These functions have to be revised for different databases and different image categories. Also, the number of image features can be increased and different similarity measures should be used to find the distance between two feature vectors [44].

2.3.4 Implications of Principal Component Analysis (PCA) and Neural Networks

One approach for feature extraction in CBIR is to reduce the dimensionality of the data set. Dimensionality of the data set is reduced when the original variables are trained but processed into smaller sets to retain as much information as possible. These are considered as merits of feature extraction. Dimensionality of the input set is reduced mostly at the cost of a loss of accuracy because it can be achieved by either eliminating data to make a smaller set of features. PCA and auto associative neural networks are feature extraction techniques, whereas for selection we use genetic algorithms and sensitivity analysis. In PCA, the covariance matrix is calculated and its eigenvectors and eigenvalues are found, then the largest eigenvalues are retained. The associative neural network consists of a multilayer preceptor with d inputs, d outputs, and M hidden units, with M<d. input vectors given to the network as the targets used to train the networks. Thus the network maps each vector on to itself. Input vectors may not be perfectly reconstructed because of the lesser number of units in the middle layer. Using neural networks for feature extraction has not yielded good results as the dimensionality of the data set is reduced, which results in the removal of some features. In the radial basis function, however, the effect is less severe. Research indicates that PCA and sensitivity analysis techniques have produced optimum results [50].

These systems might improve the retrieval results but there are still some issues associated with these systems. Training of the data set is time consuming and reduction in the number of image features might affect the accuracy of the system.

2.3.5 Machine Learning Techniques

Machine learning can also be integrated into CBIR. Certain situations exist where the CBIR method is used for a special task. These images are investigated from a specific domain and there are a group of queries and related known images. In such conditions, parameter learning for a system is possible to optimize retrieval performance. Relevance feedback (RF) involves machine learning technology used in CBIR. RF is a procedure where the user relates with the method to report on relevant images for optimal retrieval. [46] presents a survey of relevance feedback technologies for image retrieval until 2002.

Most of the approaches have marked images as individual queries. The most recent approach, however, is a query instance based approach [47] or support vector machine to use a two-class classifier [48]. This approach is similar to the one presented in [44] as both follow a nearest neighbor approach, but instead of using only the best matching query/database image combination, all query images are jointly considered.

Relevance feedback is employed to narrow down the gap between low level feature representation and the user's high level semantic concepts. Images are classified into relevant and irrelevant images by relevance feedback. Binary class SVMs with relevance feedback may not give accurate classification of small training samples. This limitation can be overcome by using one class SVMs to form relevance feedback for content-based image retrieval. A framework has been proposed in which one class SVMs are ensemble. The final decision of the ensemble framework depends on the number of votes of one class SVM classifiers. Each of the one class SVM classifiers is trained by using different sub feature vectors. The image is first segmented by either a block-based or region-based segmentation method. Then, features of each region are extracted. One class SVM classification is used to classify one class of the target sample from all other class samples. Each of the one class SVM classifiers selects the top 100 matched images. Then, the top 20 images are provided for users to label. Relevance feedback improves the accuracy of the retrieval process by giving higher precision values [49].

Research focus has shifted from finding low level feature extraction algorithms to reducing the semantic gap between low level features extracted from an image and high level features understood by humans. The semantic gap can be reduced by using certain current techniques that can be divided into five major categories. Object-ontology provides a mapping of descriptors to high level semantics. Color and texture quantization is key in such systems. These systems are easy to design and are suitable for systems with simple semantics. For more complex semantics, machine learning techniques are required. A decision tree is an effective image retrieval tool in mapping from low level features to high level concepts. Relevance feedback (RF) has promising results in reducing the semantic gap in the image retrieval process. A semantic template is also a useful and practical way to reduce the semantic gap. Web image retrieval is a new concept in the image retrieval process to reduce the semantic gap. It is an active research area and a practical product of it may come in the near future. There is so far no generic approach in high level semantic based image retrieval [50].

These systems have been implemented and tried on static databases; however, the challenge now is to dynamically update the hierarchal data based on user information that is fed to the system.

2.3.6 Miscellaneous Techniques

Image semantics has been taken into consideration in a technique proposed in [51]. Here, feature extraction is followed by a region based distance measure. Comparison of two images is made through semantic categorization. The retrieval speed of this system is high, and is measured through incorporated region matching with feature clustering. Another image retrieval method based on image segments has been proposed in [52]. Here, learned region weights and region codebooks are utilized in the process of image retrieval.

In another study, instead of using image regions, similar patterns of color and texture were used for comparison of images [53]. These patterns are called blobs. To illustrate, if one or more blobs are identified by a user as a 'flower', then the search can comprise of looking for a flower in other images which probably have different backgrounds. This can lead to a semantically more specific representation of the operator's query objects.

A hierarchal perceptual combination of basic image features and a relationship between different features for structure categorization has been proposed in [54]. Here, vector quantization is used on image blocks to codebooks for retrieval and representation inspired from text–based techniques and data compression. Another hybrid technique is proposed in [55], which uses rectangular blocks for segmentation of the background or foreground on the user's query region of interest. A database search is then performed by using only the foreground regions.

A statistical method containing the Wald-Wolfowitz test for the comparison of non-parametric multivariate allocations has been considered for color image retrieval [62], signifying images as groups of vectors in *RGB*-space. A number of techniques proposed in the recent years [63, 64, and 65] assume that feature space is a multiple fixed in Euclidean space. The design of the interface has been improved by clustering. A renowned technique proposed in [66] is based on feature extraction at the global level. In this study, advanced color and texture (ACT) features are used in the calculation of the feature vector.

The impact of features on image retrieval has been highlighted in many studies where so e features are selected for the calculation of feature vectors and others are skipped. The technique proposed in [56] used color and texture features. The features are used separately as well as in fused form, however, it has been concluded that addition of shape features will improve retrieval performance of the system. Another CBIR technique proposed in [57] is based on edge based structural features or shape features. A new algorithm is proposed that detects edges in images without calculating shape features. The algorithm was tested on images of cities, buildings etc where the retrieval performance is much better than another texture feature based extraction method. But for images with rich content like horses, cars etc the results of the proposed method based on edge based features were not good. The main reason given for poor results are that the information retrieved from these images was not enough. Even though with the edge based features results are not good enough, they are improved by adding color and texture features. Further improvement is brought by addition of relevance feedback into the system. Precision with the edge based feature was only 1.42 which was improved to 2.25 by adding color and texture features.

The impact of image features is of considerable importance in CBIR systems. Most of these systems used the finest features for the construction of the feature vector but hardly any of the systems tried to incorporate all the vital features. Our main objective for the improvement of retrieval results of CBIR systems is to add all the vital features to the feature vector. Selection of the optimal feature from each category such as shape, color, and texture may improve the retrieval results. Additionally, combining these features with different distance measures instead of just one might affect the system's performance in a positive way.

2.4 Similarity Measures

The distance measure plays a vital role in the process of feature comparison in CBIR systems. Different similarity measures have been used in these systems to get the best possible results.

Euclidean distance: One of the most widely used distance measures is Euclidean distance. Euclidean distance is the measure of the difference between a group of pixels in each bin of a histogram of one image and a group of pixels in each bin of a histogram of another image. Using Euclidean distance has an advantage in that adding a new image to the database does not have an impact on the distance between the query image and the existing database images [48]. It is represented by the following equation,

$$dist\,(\hat{\mathrm{I}}, \bar{\mathrm{I}}_D) = \sqrt{(Vq - Vd)^2}$$

Where Vq is a feature vector of the query image and Vd is the feature vector of database images.

Euclidean distance is compared with 6 other distance measures city block distance, sum of squared of absolute differences (SSAD), Canberra distance, maximum value distance, minkowski distance, sum of absolute difference SAD in [58] and results show that the retrieval performance using Euclidean distance is better than other distance measure used for comparison of feature vectors.

Manhattan distance: Another form of distance measure used in CBIR is Manhattan distance. It finds the similarity from one pixel to another in the form of a grid. The distance between two pixels of two images is calculated as a summation of the dissimilarity of the corresponding pixels. Its equation is given as follows:

$$Manhattan\ dist = \sum |p_{\mathrm{xi}} - q_{yi}| \tag{2.17}$$

Where i has values from 1 to n; p and q denote the database and query images respectively [67]. Distance between two points in a grid is based on horizontal or vertical path i.e. end to end with the grid lines opposite to diagonal. The Manhattan distance is a sum of the vertical and horizontal components, where diagonals are not counted and can be calculated by using Pythagorean Theorem [59].

Histogram Intersection: Histogram intersection is the similarity measure used in CBIR systems for the comparison of histograms. It unearths the common fragment of two histograms and does not rely on features present only in one histogram. The histogram intersection of the two histograms is given by the following equation:

$$d(h, h') = \sum_{i=1}^{I} \min(h_i, h'_i) \tag{2.18}$$

Histogram Intersection is the best distance measure for finding the distance between two histograms and works well for comparing the color content. Histogram method is a process of global image illustration. It removes the information about shape, texture and location. Images can have the same color content but different semantics where histogram intersection might not be produce accurate results [60].

Mahalanobis distance: Mahalanobis distance is another similarity measure used in image retrieval. It is based on the association between variables and is used for analyzing different patterns. The distance between a known data and an unknown sample data can be easily determined by using the Mahalanobis distance measure. Here, the unknown sample data refers to the query image and the known data set is the database images. The equation used to find the Mahalanobis distance between query Q and target image T is given by the equation:

$$Mahalanobis\ dist(Q, T) = (\sigma_Q - \sigma_T)^t \sum_{1}^{-1} (\sigma_Q - \sigma_T) \tag{2.19}$$

Where σ_Q and σ_T are the mean vectors of query image Q and target image T [9].

Regardless of its strength in finding the accurate distance in clustering data, Mahalanobis is has a major flaw of time complexity which is considerably high. Also a huge size of samples is required for training [61].

Minkowski distance: Minkowski form distance is an appropriate distance measure for finding the difference between two images where the dimensions or feature vector of each image are independent of one another and are equally important. It is the most widely used similarity measure in image retrieval and is given by the following formula:

$$Minkowski\ dist(x,y) = \left(\sum \mid V_i(x) - V_i(y) \mid^p\right) \tag{2.20}$$

Where x is the query image and y is the image in the database; $V_i(x)$ is the number of pixels of image x in bin i [68]. The main shortcoming of Minkowski distance is that matching is done between similar bins of histogram.

Quadratic form distance: During comparison of two images, certain pairs of bins in a feature histogram correspond to features that are more related than other pairs. For such comparisons, Minkowski distance is not considered as a good distance measure as it treats all the bins entirely independent of each other. Quadratic form distance is introduced for such problems. It is given by the following equation:

$$QD(x,y) = \sqrt{(V_i - V_j)^p\, S\, (V_i - V_j)} \tag{2.21}$$

Where, S = [s_{ij}] is the similarity matrix, s_{ij} = similarity between bin i and j, V_i and V_j are feature vectors

Quadratic form distance has been widely used in CBIR for color histogram comparison. It has shown better results compared to the Histogram Intersection method and Euclidean distance [68].

The performance of different similarity measures has been evaluated through various studies conducted over recent years. It has been found that Manhattan distance yields the best result in image retrieval. Its performance is reported to be marginally better than Euclidean distance. Quadratic distance's performance, on the other hand, has been shown to be considerably lower than the Euclidean and Manhattan distance measures. The main reason for this low performance is the weights related to the quadratic distance measure [69, 70].

2.5 Summary

Regardless of the significant progress of research in CBIR, the effect of this technology is not considerable. This is due to a number of reasons. Firstly, regarding image semantics, an image might not belong to only one class based on its semantics. Different contexts can lead to different interpretations of the same image. Secondly, improvement is required in the process of feature extraction. Most of the CBIR systems are based on features that are pre-calculated and stored in the system. Retrieval efficiency is very slow for such systems because of the continuous insertion or removal of images from the database.

Furthermore, there is only slight improvement in the accuracy of the retrieved results in CBIR systems. The reason behind this is the lack of ability of current image features to extract the unique characteristics of the image. Another factor causing poor retrieval is the semantic gap. Although recent research has incorporated relevance feedback to improve results, these systems are mostly designed for static image databases and are less effective for dynamic image databases. This study will focus on different image features and their combination with different similarity measures. Retrieval at both the global and local level will be utilized to find a better system with improved results. Relevance feedback will also be incorporated to propose a suitable framework for CBIR.

2.6 Problem Statements

- Impact of segmentation on image retrieval.
- Impact of image features (Fourier descriptors, edge histogram descriptors, and local binary patterns) on image retrieval.
- Impact of feature fusion and computational intelligence techniques on image retrieval.
- Impact of automatic fuzzy membership function on the process of image retrieval.
- Impact of feature selection process on image retrieval.

Chapter 3

3 Proposed CBIR Approach using Local and Global Features

The two main approaches for retrieving a required image from a database are known as the local approach and the global approach. Each method is named based on the type of feature extraction used and the calculation of the feature vector. In the local method, the image undergoes the segmentation process and features are taken out from each segment independently. The global technique, on the other hand, does not rely on the segmentation process. Rather, it uses the image as a whole for abstraction of features. This chapter describes two methods proposed for CBIR based on the local and global approaches of feature extraction. The three major image features - color, texture, and shape - are adopted along with two additional features - FD and EDH - for extortion of information from the boundaries and edges. Retrieval results of the two proposed methods are compared to find the optimal approach to image retrieval. The proposed systems are also evaluated by comparing their experimental results with some state of the art systems in the literature.

3.1 Local Approach to CBIR

There are two different approaches to CBIR that are differentiated based on their process of feature extraction. The local or region based approach divides the image into meaningful parts first and then extracts all image features from each of the regions separately. Such an approach is known as the local approach, as the image is segmented into multiple parts each being local to the image under consideration. The segmentation process is the initial stage in CBIR systems developed following the local approach. An appropriate segmentation technique is adopted in order to divide the image into different regions without any loss of information. The segmentation technique implemented for such CBIR systems are of great importance as data loss can occur as a result of deficient segmentation. Segmentation based on color, texture, shape, and spatial location is further utilized to search and retrieve similar regions from the database [24]. The segmentation algorithm plays an important role in feature extraction at the local level and it can be achieved by a number of means. Each method segments an image based on certain properties [1]. Image features are then extracted from each segmented region and a feature vector is constructed for the analogous region. Figure 3.1 shows the general flow of processes for local level feature extraction. After the feature vector calculation of the query image, comparison is drawn with all the feature vectors in the database and the image with least distance is retrieved as an output.

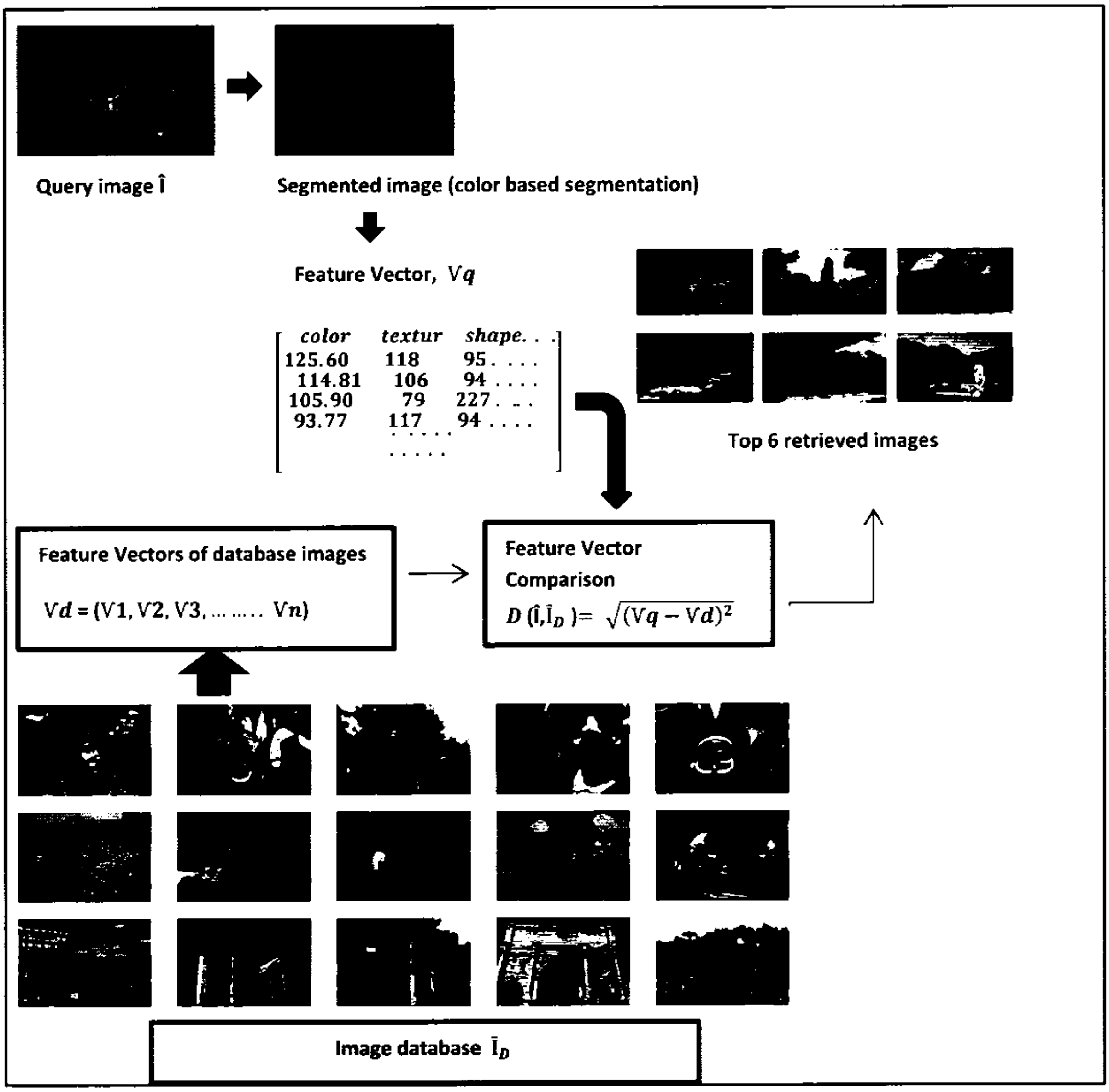


Figure 3-1: Image retrieval at local level

3.1.1 Proposed Framework

In the proposed framework, the image will first be subjected to the segmentation process to divide it into meaningful regions. After segmentation, all the main image features will be taken out. The proposed system abstracts the three significant features (color, shape, and texture). In addition to the three main image features, EHD will also be calculated to extort information at the edges. A flow chart of the proposed technique is presented in Figure 3.2.

Algorithm 3.1:

Step 1: Divide the query image Î into 3 clusters by K-means clustering technique.

Step 2: A color histogram Ḣ is constructed to extract color feature Î :f for each segment.

Step 3: Extract texture feature Î: f by calculating the Co-occurrence matrix *C(i, j)*.

Step 4: Let $x|k|$ and $y|k|$ be the coordinates of the k^{th} pixel on the boundary of a given 2D shape containing n pixels, a complex number can be formed as

$z|k| = x|k|+py|k|$, and the *Fourier Descriptor (FD)* of this shape is$a(\mu)$, which is given by the following equation

$$a(\mu) = \frac{1}{N}\sum_{k=0}^{N-1} z|k|e^{-p\frac{2\pi}{N}\mu k} \quad (N = 0, \; \dots, \; n-1)$$

N refers to the number of images.

Step 5: The edge histogram descriptor EHD is calculated from four directional edges, namely, vertical, horizontal, 45 degree, and 135 degree, and one non-directional edge.

Step 6: Euclidean distance is used to calculate the distance between the feature vector of the database images *Vd* and the feature vector of the query image *Vq*.

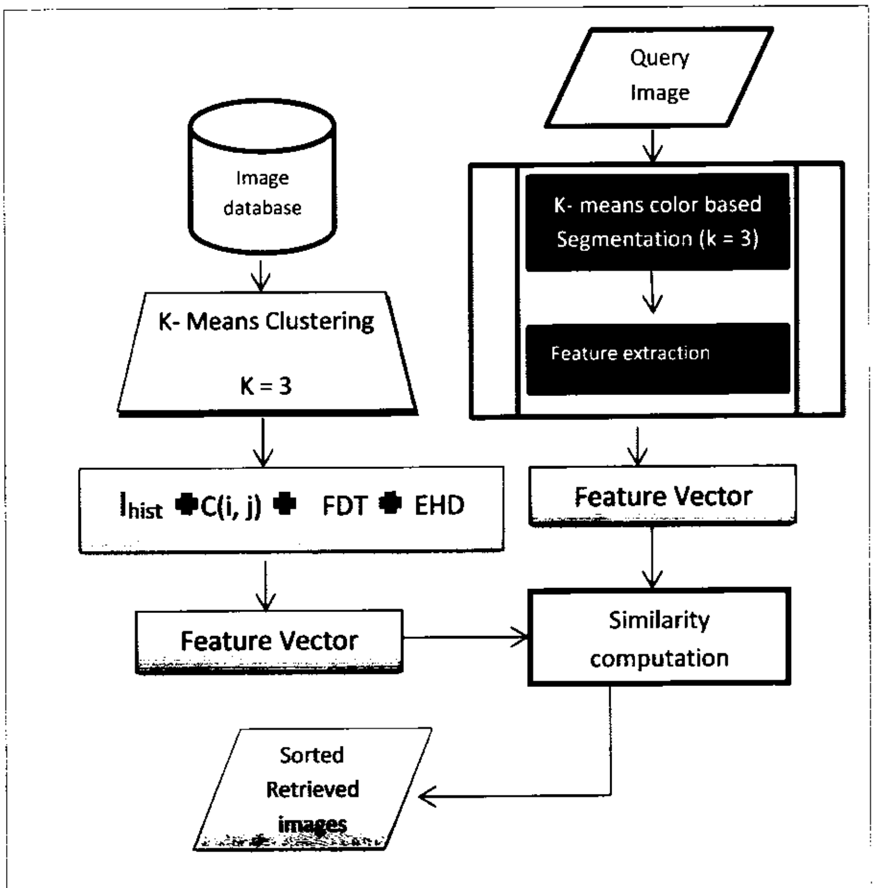


Figure 3-2: Flow Chart of the Proposed Technique.

3.1.1.1 Image segmentation

In the proposed system, image segmentation is the primary step for image retrieval like other CBIR techniques based on the local approach. Segmentation divides the image into its constituent parts or regions. The subdivision process ends when all the objects of interest are taken out from an image. Thus the outcome of segmentation demonstrates the objects that constitute the image. This normally results in an image having two parts, the foreground and the background. The foreground denotes objects of interest or important parts of an image, while the background characterizes the left over portion [71].

Image segmentation can be attained by a number of means. Each process segments an image grounded on certain characteristics. One of these methods is edge detection, which has been used for many years. In this method, once the edges in an image are identified, they can be taken into consideration for detecting different objects. All edges after detection can produce convenient information about different contents of the image [72]. The Canny edge detector is one of the tools used to extract boundaries from an image. The Canny edge detection algorithm creates thin fragments of image contours and is proscribed by a single smoothing parameter. The image is leveled with a Gaussian filter of certain suitable spread and then gradient magnitude and gradient direction are calculated for each pixel of the leveled image [73]. During the course of region developing, the image is not essentially segmented, rather partitioned into regions; the region grower begins at one end of the image and moves to the other adding the same pixels. However, when a pixel with a changed value comes across its way, it stops and limits the region up to the different valued pixel. Thresholding is one of the most vital techniques for image segmentation. A value is designated as a threshold to partition the image. The method of thresholding is also used in text categorization, that is, the process of assigning text into one or more categories. The objective is to allocate a label to a new unseen document [74].

Image segmentation is of great importance in the proposed local approach. The segmentation method selected here is the K-means clustering technique which is a flexible method of dividing the image into segments as the number of clusters is not fixed and can be changed based on the results to avoid under or over segmentation. The proposed system tried different numbers of clusters and then three was found as the best choice for the proposed CBIR system.

3.2 Global approach to CBIR

The global technique extracts image features from the whole image. It does not involve any segmentation or division of the image prior to the feature extraction process. This study proposes a method based on global level feature extraction. Since no segmentation is involved in such a method, chances of losing any data are less. Our proposed technique based on the global approach to CBIR accomplishes the following characteristics:

- Three vital image features (color, shape, and texture) are extorted for feature vector calculation.
- FDs and EHDs are also added to the feature vector to extract information from the edges and boundaries.
- Similarity between the query and database images is calculated through Euclidean distance.

3.2.1 Proposed framework

The second technique proposed here is based on the global approach where the image is considered as a whole when sent for the feature extraction process. The image features to be extracted for the calculation of the feature vector are color, texture, shape, FD, and EHD. After extraction, these feature vectors will be kept in a database in the form of Excel sheets. The query image given to the system also goes through the feature extraction process in the same way and comparison with all database images is made by calculating the difference between the characteristics of two images. Retrieved images would be sorted on the basis of the least Euclidean distance between two images. A diagrammatic representation of the proposed technique is given in Figure 3.3.

Algorithm 3.2:

Step 1: Color histogram Ḣ is constructed to extract color feature Î: ƒ for each segment of image Î, where i=3.

Step 2: Construct a texture feature Î: ƒ by calculating Co-occurrence matrix *C(i, j)*.

Step 3: Let $x|k|$ and $y|k|$ be the coordinates of the k^{th} pixel on the boundary of a given 2D shape containing n pixels, a complex number can be formed as

$z|k| = x|k| + py|k|$,

and the *Fourier Descriptor (FD)* of this shape is defined as the DFT of Z|*k*|

$$a(\mu) = \frac{1}{N}\sum_{k=0}^{N-1} z|k|e^{-p\frac{2\pi}{N}\mu k} \quad (N = 0,\ \ldots,\ n-1)$$

Step 4: Edge histogram descriptor EHD is calculated from four directional edges, namely vertical, horizontal, 45 degree, and 135 degree, and one non-directional edge.

Step 5: Euclidean distance is used to calculate the distance between the feature vector of the database images Vd and the feature vector of the query image Vq.

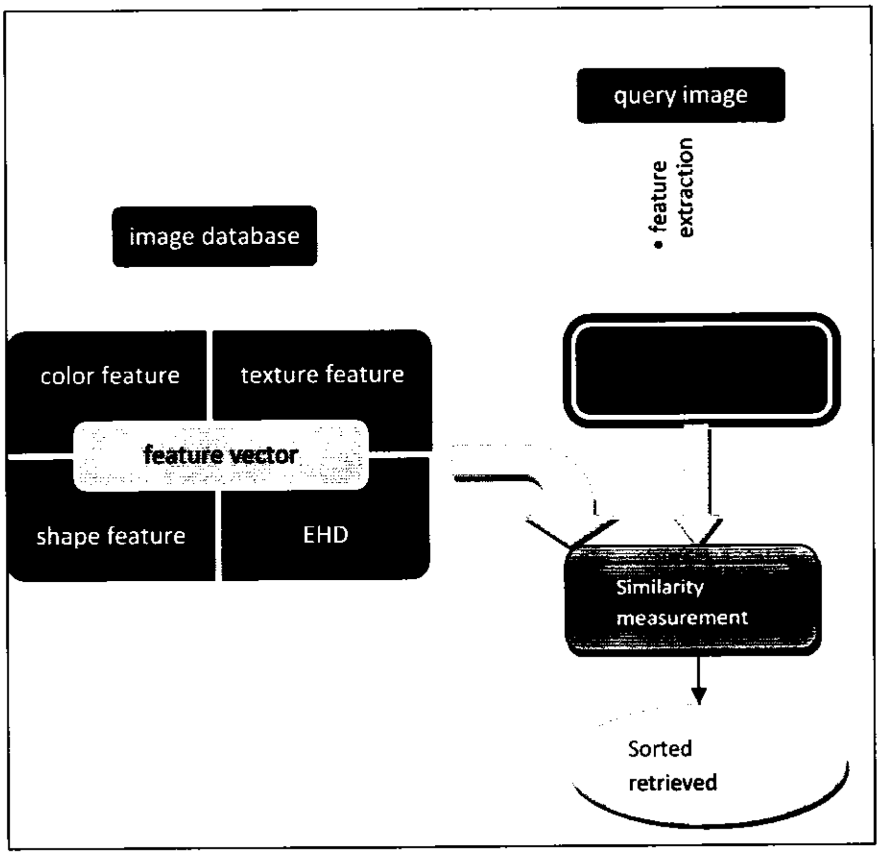


Figure 3-3: Block Diagram of proposed technique

3.3 Feature Extraction

All the key features of an image are extracted in the proposed technique. These features include color, texture, and shape features. EHD will also be calculated in addition to the core features to extract information at the edges.

3.3.1 Color Feature

The color features used in the proposed technique are calculated from a color histogram. A color histogram defines the dissemination of colors in an image and is invariant to translation, rotation, and scaling of an object. However, color histograms do not contain any semantic information, hence two images can hold different contents having the same histogram. The color histogram H for an image is given as:

$$H = \{h[1], h[2], \ldots h[i], \ldots, h[N]\}$$

Where i denotes a certain color in the histogram, $h[i]$ is the pixel count of colors i, and N defines the number of bins in the color histogram.

The proposed method extracts color features by calculating the mean, mode, median, and standard deviation for every color channel. Thus, twelve color features will be extracted for the query image and every database image. Gray levels and pixel counts will be extracted from the histogram of gray images. Each time, one color channel is taken for calculation of the above-mentioned features. Table 3.1 shows a list of color features used by the proposed technique along with their equations [2].

Color is considered to be the most effective image feature in almost all CBIR systems because of the advantages of effectiveness, robustness because of its implementation easiness and low space requirements [31].

Table 3-1: Color features used in the proposed technique

Image feature	Equation with symbol definition	
Mean	$\text{Mean} = m = \frac{\sum_{i=1}^{n} X_i}{n}$ where X= gray levels, i = index	(3.1)
Median	$\text{Median} = \frac{n+1}{2}$ where n = pixel count	(3.2)
Mode	$\text{Mode} = \max(X_i)$	(3.3)
Standard deviation	$\text{Standard deviation} = \sigma = \sqrt{\frac{\sum (X-m)^2}{n-1}}$	(3.4)

3.3.2 Texture Feature

Texture feature computation was done by constructing a co-occurrence variance matrix. A gray level image was used as input for computation of the co-occurrence variance matrix.

The co-occurrence matrix C(i, j) sums the number of pixels in a certain order having gray values I and j at a specific distance d which is given in polar coordinates (d, θ). These polar coordinates have distinct length and direction. The direction θ takes values from eight different angles: 0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°.

The first and most widely used method for the extraction of texture features is constructing a grey level co-occurrence matrix. Ten different features including mean (M), standard deviation (S), maximum probability (MAXP), uniformity (UNI), entropy (ENT), skewness (SK), difference moment, inverse element difference moment of order 2 (IKMOM), homogeneity (HOMO), contrast (CONT) and correlation (CORR) were extracted by using different formulae [2]. All these texture features are presented with their formula in Table 3.2.

Table 3-2: Texture features used in the proposed technique

Image feature	Equation with symbol definition	
Skewness	$$SK = \frac{\sum\left(\frac{X-m}{\sigma}\right)^3}{n}$$ where m = mean, σ = standard deviation	(3.5)
Correlation	$$CORR = r = \frac{\sum_i \sum_j (ij)C(i,j) - \mu_i\mu_j}{\sigma_i\sigma_j}$$ Where, $\mu_i = \sum_i i \sum_j C(i,j)$ $$\mu_j = \sum_j j \sum_i C(i,j)$$ And $\sigma_i = \sum_i (i - \mu_i)^2 \sum_j C(i,j)$ $\sigma_j = \sum_i (j - \mu_j)^2 \sum_i C(i,j)$ C(i, j) = gray level co-occurrence matrix	(3.6)

Maximum probability	$$\text{MAXP} = \max\left(\frac{n(A)}{n}\right)$$ where A = probability	(3.7)
Uniformity	$$\text{UNI} = \left(1 - \frac{\sigma}{m}\right) \times 100$$ where σ = standard deviation, m = mean	(3.8)
Entropy	$$\text{ENT} = -\sum_{i}\sum_{j} C(i,j)\log C(i,j)$$ Where C(i,j) = gray level co-occurrence matrix	(3.9)
Contrast	$$\text{CONT} = \sqrt{\frac{1}{MN}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}(I_{ij} - I)^2}$$ where M by N = size of 2 dimensional image, I = average intensity of all pixels in the image, $I_{i,j}$= i-th and j-th element of an image.	(3.10)
Homogeneity	$$\text{HOMO} = \sum_{i,j=0}^{N-1}\frac{C(i,j)}{1 + (i-j)^2}$$ where C(i,j)= gray level co-occurrence matrix	(3.11)
Difference Moment	$$\text{IKMOM} = \sum_{i}\sum_{j}\frac{1}{1 + (i-j)^2}\, C(i,j)$$ where C(i,j) = gray level co-occurrence matrix	(3.12)

3.3.3 Shape Features

In contour based shape feature abstraction procedures, a contour system was utilized. Eight different features were extracted from the image's contour, which are presented in Table 3.3 with equations [2].

Table 3-3: Shape features used in the proposed technique

Image feature	Equation with symbol definition	
Circularity	$$Cir = \frac{4_P A}{P^2}$$ Where A = area of polygon enclosed by segment boundary, P = perimeter of polygon enclosed by segment boundary	(3.13)
Aspect Ratio	$$\text{Aspect Ratio} = \frac{P1 + P2}{C}$$ P1,P2 = greatest perpendicular distances from longest chord to boundary, in each half-space either side of line through longest chord, C = length of longest boundary chord	(3.14)
Complexity	$$\text{Complexity} = \text{LogN}\,\frac{l!}{n(n)!}$$ A measure of the number of segments in a boundary group weighted such that small changes in the number of segments have more effect in low complexity shapes than in high complexity shapes.	(3.15)
Discontinuity-angle irregularity	$$\text{Dar} = \sqrt{\frac{\sum \lvert \theta_i - \theta_{i+1} \rvert}{2 \times \pi \times (n-2)}}$$ where θ_i= discontinuity angle between (i−1)-th and i-th boundary segment	(3.16)
Solidity	$$\text{Solidity} = D = \frac{A_s}{H}$$ where As = area of the shape region, H = convex hull area of the shape	(3.17)

3.3.3.1 Fourier Descriptor Method

Methods used for boundary-based recovery of shape characteristics include Fourier descriptors FD [75], Wavelet descriptors, Curvature scale space descriptors, Shape signatures, etc. Among all these methods, the FD method is the most fundamental one. The FD method computes shape signature functions by using shape boundary coordinates. These shape signatures utilize Fourier transform for computation of FD [76]. Shape is analyzed in the spectral domain by FD to surmount the consequences of sound and boundary differences on shape feature extraction. Also, the FD method for shape feature extraction is computationally inexpensive, easy to normalize, and has been proved to outperform many other boundary based techniques in terms of accurate and fast recovery [77]. FD is applied on each object separately after its identification through boundaries or edges.

Employment of the shape signature function, calculation of boundary pixels, and FD are the main constituents of the General FD method. Boundary pixel computation for boundary coordinates is performed through edge detector and boundary tracing techniques [78].

A pixel set is formed after the computation of boundary pixels and is represented by the following formula:

$$P = \{(x(t), y(t)) \mid t \in [1, N]\} \tag{3.18}$$

Where x and y are the coordinates of a pixel t in an image and the number of pixels can range from 1 to N.

In FD method, pixel coordinates of the shape boundary in a figure are used to measure shape signatures. Fourier transforms are employed for these shape signatures to evaluate Fourier transformed coefficients which are used as FDs [75].

3.3.3.2 Edge Histogram Descriptor (EHD)

Edge is also an important image feature that can be used to extract information present at the contrast. EHD signifies the spatial circulation of five types of edges. Among these, four are called directional edges, namely vertical, horizontal, 45 degree, and 135 degree, and one is a non-directional edge. Edge strength is detected by the application of filter coefficients as shown in Figure 3.4. Edge blocks bigger than a given verge were selected.

1	-1
1	-1

1	1
-1	-1

$\sqrt{2}$	0
0	$\sqrt{2}$

2	-2
-2	2

Figure 3-4: Filter Coefficients

MPEG-7 standards reveal that retrieval performance of images can be improved by calculation of EHD. This descriptor shows invariance to scale and rotation.

The edge histogram descriptor represents the spatial distribution of five types of edges, four directional edges and one non-directional edge and might improve the retrieval results. Its effectiveness in image retrieval could be found by testing it on a dataset of images where shape is of main concern. MPEG-7 is a database which consists of 70 classes of shapes each having 20 members. According to the MPEG-7 standard which is a benchmark for shape based images, if the edge histogram descriptor is joined with a color histogram descriptor, the recovery performance of an image is radically improved. The descriptor is scale invariant and supportive to rotation invariant and rotation sensitive matching operations because edges are calculated in five different directions [27].

3.4 Experimental Results

3.4.1 User Interface

Experiments were carried out on natural images to test the retrieval efficiency of the proposed method. An easy to use graphical user interface (GUI) was designed where the query image was browsed from the system and a threshold was chosen for the segmentation process. The system was then able to retrieve the top 10 similar images from the database. Figure 3.5 shows the main GUI of the proposed system.

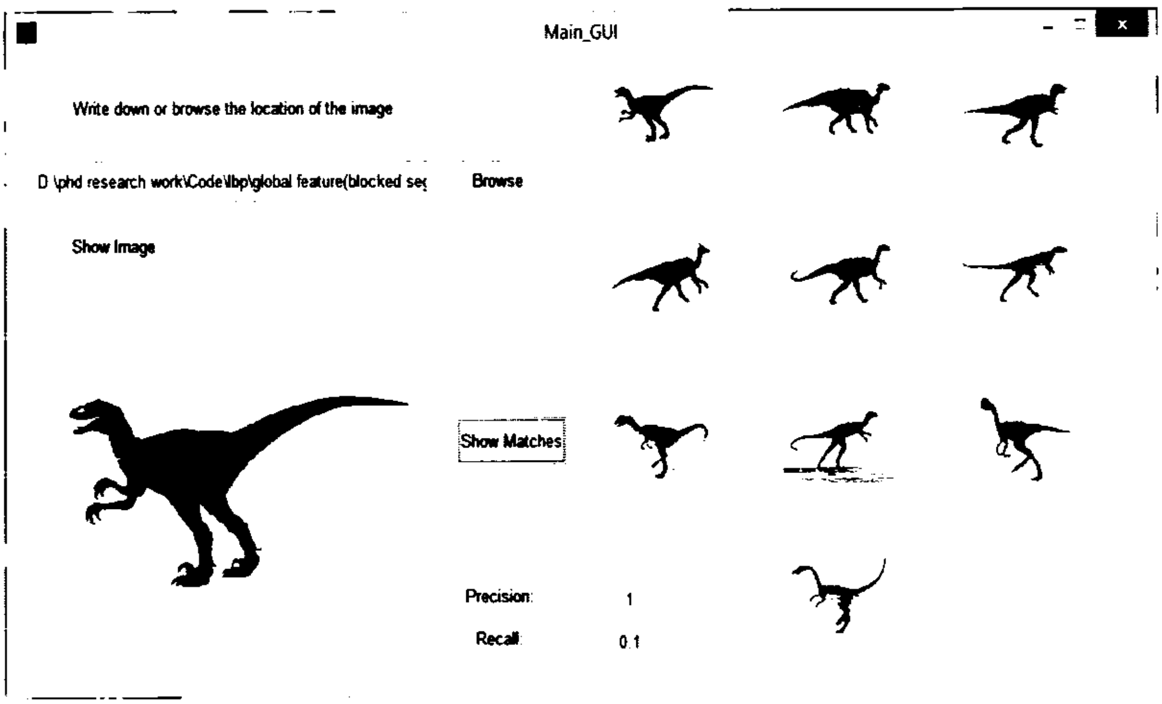


Figure 3-5: Main GUI of the proposed technique

3.4.2 Data Set

The proposed technique was tested on a database of 1,000 natural images from the Corel image database available free on the Internet. Corel images are stored in JPEG format with sizes 384 × 256 or 256 × 384. The entire database has 10 different image categories, where each category contains 100 images. All these images' categories contain diverse semantics including 'beach', 'vehicle', 'dinosaur', etc. Experiments were performed on 10 randomly chosen image queries, one from each category. The sample image from each category selected for experimentation is given in Figure 3.6. All these classes had different semantics and could therefore better evaluate the proposed method.

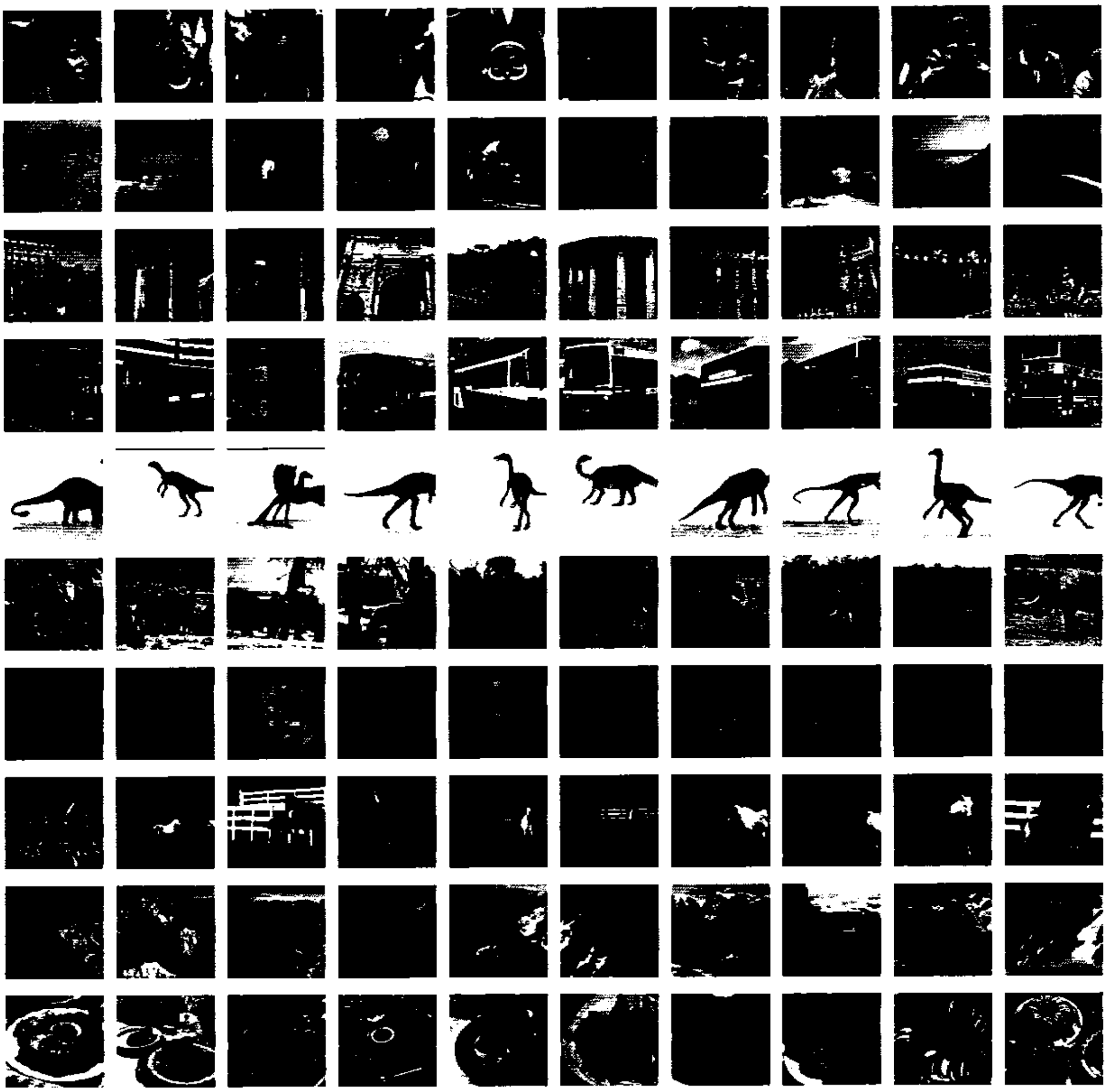


Figure 3-6: Sample images from the 10 different image categories of the Corel image database used in the experiments

3.4.3 Performance Measures

The performance measures used to find the retrieval efficiency of the proposed system were constructed on the measures used in image retrieval. These metrics comprised of precision and recall. Precision is defined as the ratio of the number of retrieved relevant images to the total number of retrieved images in the database. Precision indicates accuracy and is represented by the following equation:

Precision = R / N

Where

R = Number of retrieved relevant images

N = Total number of retrieved images

The best precision is the one where each image retrieved is relevant. However, it does not provide any information declaring as to whether all the relevant images are retrieved.

Recall is the ratio of the number of relevant retrieved images to the total number of relevant images in the database. Recall signifies completeness and is represented by the following equation:

Recall = R / M

Where

R = Number of retrieved relevant images

M = Total number of relevant images

The best recall is the one where all images retrieved are relevant. However, it does not give any information about the irrelevant images in the database that could have also been retrieved.

3.4.4 Impact of FD and EHD on Image Retrieval

In addition to the three basic features (BF) color, texture, and shape, FDs and EHD were also combined to extract the maximum statistics of the image. FD is invariant to rotation, translation, and scaling. FDs relate to the low frequency components of the boundary to represent a 2D shape. The reinstated shape based on these descriptors estimate the shape without details corresponding to high frequency components vulnerable to noise. Figure 3.7 shows the comparison of retrieval results for two feature vectors. The retrieval results of the feature vector integrated with FD and EHD were better than the retrieval result of the feature vector with BF for five distinct image categories.

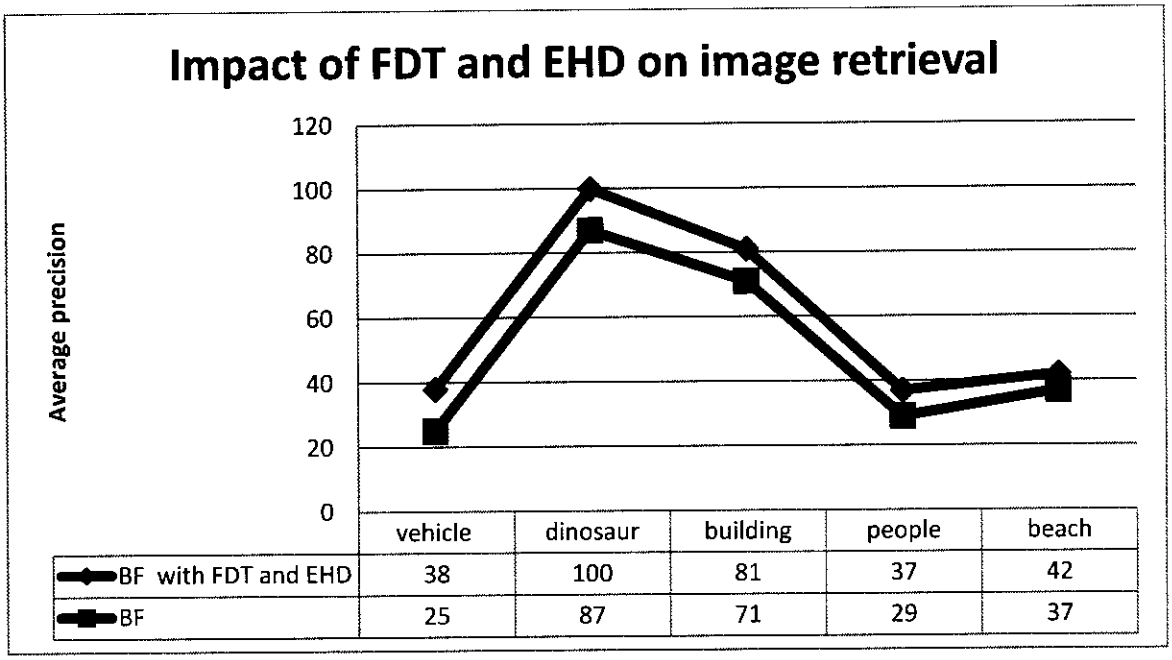


Figure 3-7: Comparison of retrieval results of feature vectors using FD and EHD with basic features

Greater improvement was observed in the results for 'vehicle' and 'dinosaur'. These image categories had a lesser number of objects; therefore most of the edges and boundaries were detected by FD and EHD, which added to the information retrieved from these image categories. While the other three categories, having more objects, resulted in a less number of edges detected - hence showing little improvement in results as compared to 'vehicle' and 'dinosaur'. The graph shows a consistent improvement in the retrieval precision of 5 different image categories. Improvement is observed in all types of images which show that FD and EHD are important shape features that improve the average precision. Maximum difference in precision is found for dinosaur having images with a plane background and one object only. For these images calculation of five types of edges through EHD was enough to extract shape features while people and beach has many objects with different background and the calculation of five different types of edges did not help much to extract shape feature based information thus giving little improvement in precision results.

3.4.5 Proposed Local Technique versus Universal Model (UM) Technique

A local level CBIR technique was suggested where images are first subdivided based on the color content and then image features are extracted from different segments of image. The proposed technique was matched with.one of the local level feature extraction techniques in the literature, i.e., UM [79]. Results showed that the proposed technique had greater precision compared to the UM technique. The overall retrieval result of 10 images returned by using six different image categories is compared in Figure 3.8.

UM is a CBIR technique that extracts image features at a local level. Color and texture features from each segment of an image are extracted.to calculate the feature vector. EHD is also considered to grasp information at the edges. Similarity is based on a greedy algorithm where a threshold is set first to compare the query image's segment with the image segments in the database. Here, fine-tuning of the threshold is testing. The threshold value is different for each image category and is inconsistent even for each image within the particular image category [79, 80].

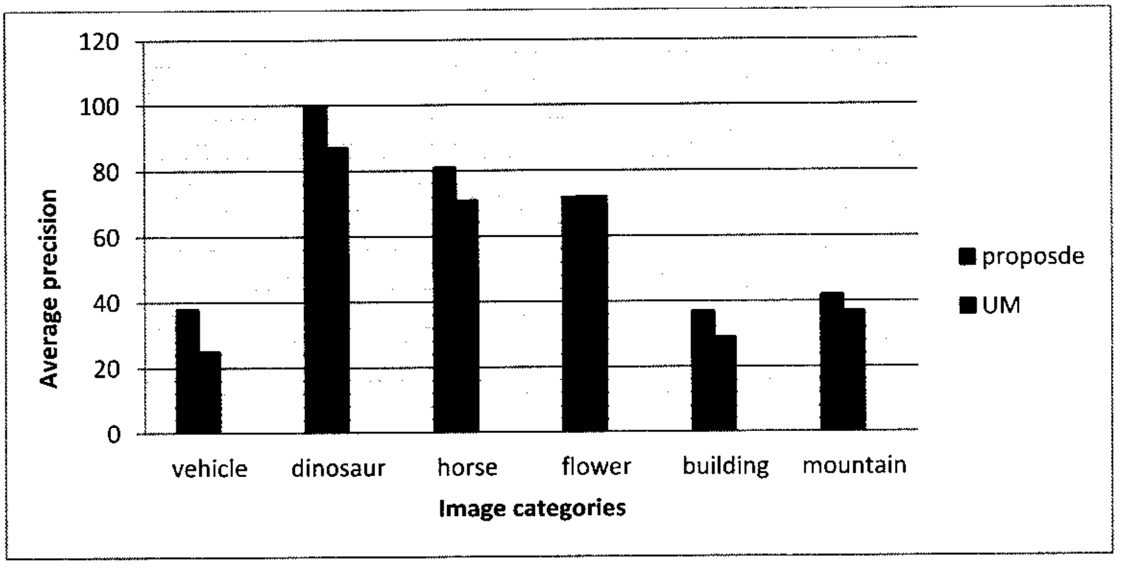


Figure 3-8: Comparison of the overall retrieval precision of 10 images returned by using six different image categories.

The above figure shows that the proposed method has better results for all types of images. The features used.by UM were color, texture, and EHD. The proposed system, however, added shape and spatial features in the feature vector, which gave more discrimination power to it. This larger number of features and the segmentation technique used were crucial in improving retrieval results for the proposed technique. K-means clustering segmentation is employed based on the color content of the image. Different values of k were tried and after retrieving results through different clusters k = 3 was found as a best choice which gave better precision compared to other values of k. Increasing the number of clusters cause data loss due to which the feature vector constructed in not beneficial and value of k less than 3 did not segment the image properly. The 'flower' image category showed less improvement because of the structure of objects in these images and the reduced number of edges compared to other image categories, where edges were identified well.by the proposed system. The addition of shape features did not have a great impact on the feature vector of this image category.

3.4.5.1 Variants of UM

We have proposed two variants of UM, which is a local level feature extraction method proposed in 2008.

1. UM with shape feature.
2. UM without threshold.

After matching the results of the first alternative, i.e., UM with shape, with the original UM, we found no enhancement in the results. This means that the incorporation of the shape feature does not work well with the UM model. However, the second variant, i.e., UM without threshold, showed improved results in comparison with the original UM method. Table 3.4 demonstrates the comparison of the UM model with its two variants. The threshold (for greedy algorithm) decreases the retrieval precision and increases the time complexity. Also, fine-tuning of the threshold is tedious in the UM model. It gives diverse precision results every time the same threshold is given for the same image. Therefore, there is no uniformity in results. In this study, variants of the UM model were applied and tested on six different image categories of the Corel image database. Table

3.4 reviews the results for the two variants of UM in comparison with the original UM technique for different image categories.

Table 3-4: Comparison of percentage average retrieval precision of UM with its two different variants.

S.No	Image Category	UM (%)	UM without threshold (%)	UM with Shape Feature (%)
1	Vehicle	25.00	43.33	24.42
2	Dinosaur	87.22	100.00	87.12
3	Horse	71.25	83.00	72.41
4	Flower	72.20	77.45	75.34
5	Building	29.14	29.30	28.20
6	Mountain	37.24	41.33	38.00
Average		51.05	62.40	54.24

The above table shows that shape features did not improve the retrieval results much. For three image categories i.e. horse, flower and mountain only a slight improvement is observed. Other than these three categories there is no improvement in the precision because in images in 'vehicle' and 'dinosaur', the number of objects is less and color and texture feature were enough to extract information from these images. Flower and mountain on the other side has many objects and shape addition of shape feature helped to improve feature vector and extract information from the image. UM uses a threshold value during the segmentation stage of image retrieval system. These results also show that threshold was not found good to be incorporated in CBIR system. The value of threshold was not fixed and each time for a different image category it took a different threshold value. We removed the threshold and retrieved images without giving a threshold value. Results were better compared to the one using threshold for all the 6 different image categories. The reason for this could be a non-determined or a variable value of threshold.

3.4.6 Comparison with recent local level CBIR techniques

The local technique proposed is compared with FRCE and PRIR. These are the CBIR techniques based on the local approach where segmentation is performed before feature extraction. Average precision for different categories of images is calculated. The top number of images considered was 10, 20, 30, 40 and 50.retrieved. It is shown the proposed method outperforms the other two methods in the literature.

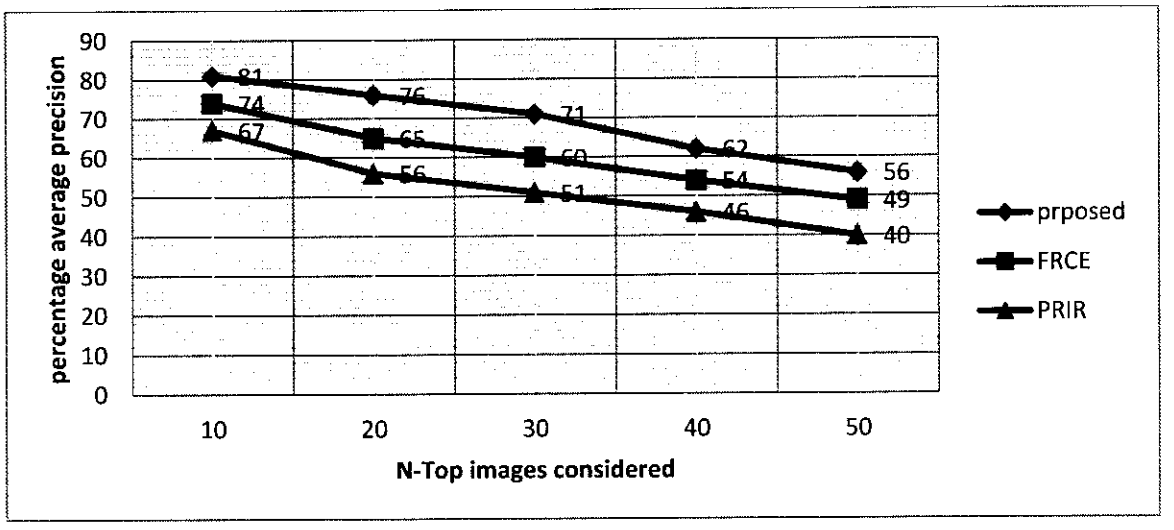


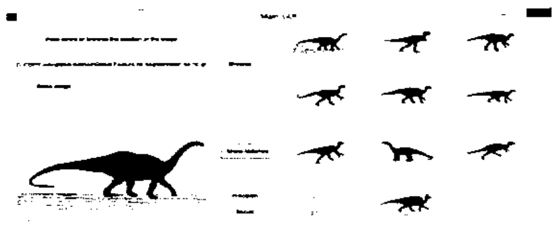
Figure 3-9: Comparison of percentage average precision of the proposed methods with FRCE and PRIR at N images retrieved

3.4.7 Qualitative Evaluation of Proposed Global Technique

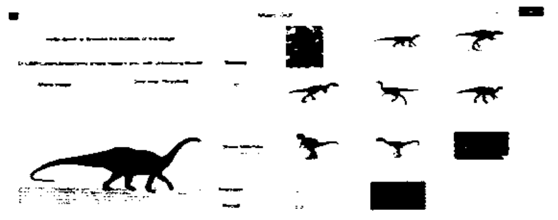
Five different pictures with varied structures, namely: 'beach', 'vehicle', 'horse', 'dinosaur', and 'people' were randomly selected for a qualitative evaluation. All these selected image categories were totally different from.one another in their content. Here, the image retrieved was taken as a correct match if it fell in a category parallel to the query image. The results of the proposed global technique for CBIR compared to one of the local CBIR techniques, UM [79], which is one of the top region based retrieval methods we aware of, are shown side by side in Figure 3.9. For each member query image, the top 10 retrieved images are given for both methods.

Retrieval results by using our proposed technique

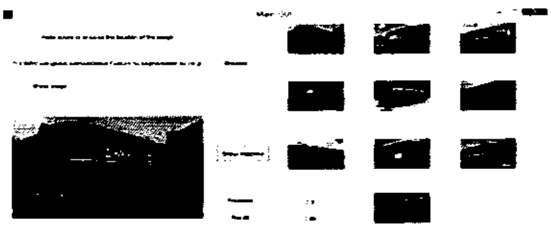
Retrieval results by using the UM method



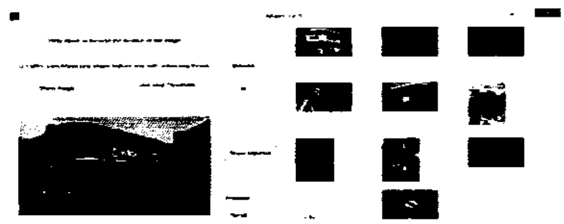
(a) 10 matches out of 10



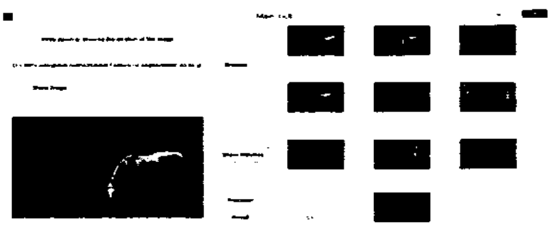
7 matches out of 10



(b) 8 matches out of 10

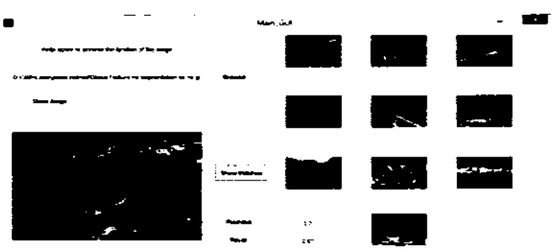


2 matches out of 10

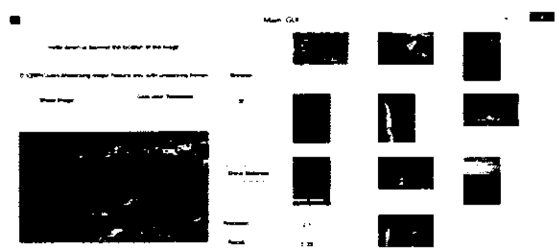


(c) 10 matches out of 10

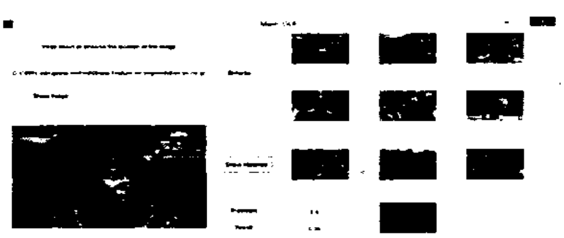
8 matches out of 10



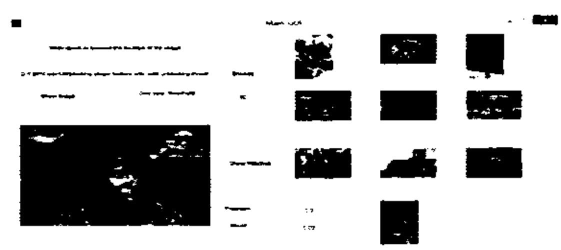
(d) 7 matches out of 10



3 matches out of 10



(e) 6 matches out of 10



3 matches out of 10

Figure 3-9: Retrieval results of five query images by using the proposed technique and the UM method

Our method produced expected outcomes for 'dinosaur' and 'horse' queries while the results of the UM method showed some extraneous images. This is because of the elimination of segmentation from our proposed system, resulting in less information loss and better construction of the feature vector. Our proposed method also generated much better outcomes of 'vehicle', 'beach' and 'people' than the UM method.

3.4.8 Quantitative Evaluation of Proposed Global Technique

For quantitative evaluation, the proposed technique was tested on 10 different images selected randomly from five distinct image categories. The average retrieval accuracy of each category was calculated by finding the top 10, 20, 30, 40, and 50 retrieved images and is given in Table 3.5. It was found that the results for 'dinosaur' were more consistent compared to other image categories. All the image categories produced the best results when the top 10 images were returned as an output. However, precision decreased by increasing the number of returned images.

Table 3-5: Average precision at different number of images retrieved

Image Category	Beach	Vehicle	Horse	Dinosaur	People
Retrieval Precision @ 10	44.05	57.00	94.45	100.00	57.21
Retrieval Precision @ 20	37.15	51.08	85.03	98.35	48.24
Retrieval Precision @ 30	39.16	43.00	80.30	99.09	39.72
Retrieval Precision @ 40	36.37	39.71	78.03	98.28	35.05
Retrieval Precision @ 50	35.46	32.30	71.04	98.28	31.80

3.4.9 Comparison of Proposed Global Technique with ACT

Our proposed global technique was also compared with a prominent global technique from the literature called ACT. The average retrieval precision of the proposed technique was compared with the retrieval values of ACT for six different image categories. Results showed that the retrieval precision of the proposed technique was better than ACT for all image categories. Comparison of the two methods is shown in Figure 3.10.

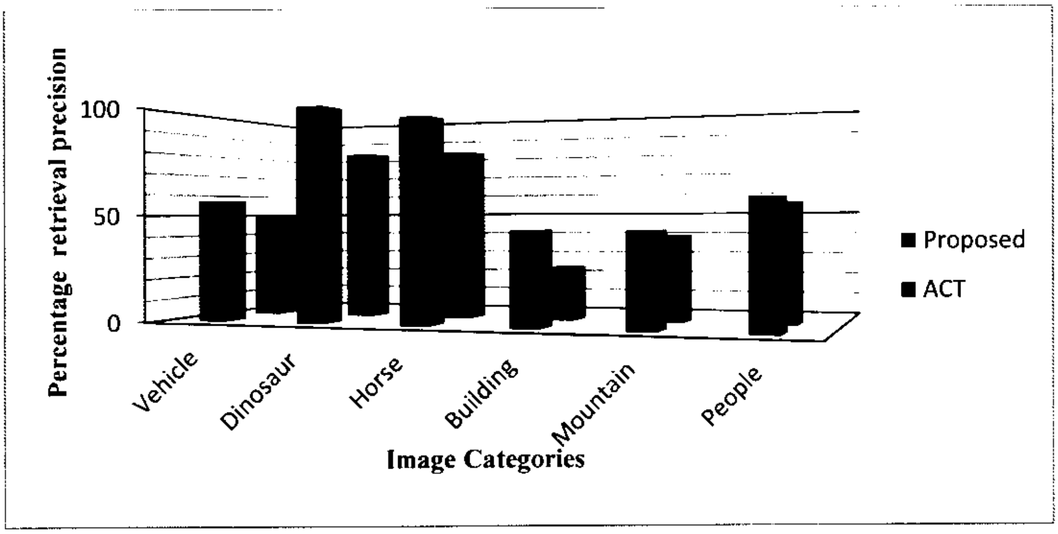


Figure 3-10: Comparison of the average retrieval precision of the proposed global technique with ACT

The proposed technique makes use of all the basic features, thereby extracting greater information in constructing a feature vector. Considerable improvement was shown for 'building' image types because of the shape features and EHD in the proposed method that detected and extracted most of the edges in these images.

3.4.10 Comparison of Proposed Global Method (GM) with UM

Comparison of the proposed global technique was also made with a prominent local technique called UM. Results showed that the retrieval precision of the proposed global technique was high compared to UM. UM is a local technique where segmentation is done before feature vector calculation. Segmentation might result in data loss, consequently affecting retrieval of required images from the database. Figure 3.11 reports a comparison of percentage retrieval precision of the two methods for six distinct image categories.

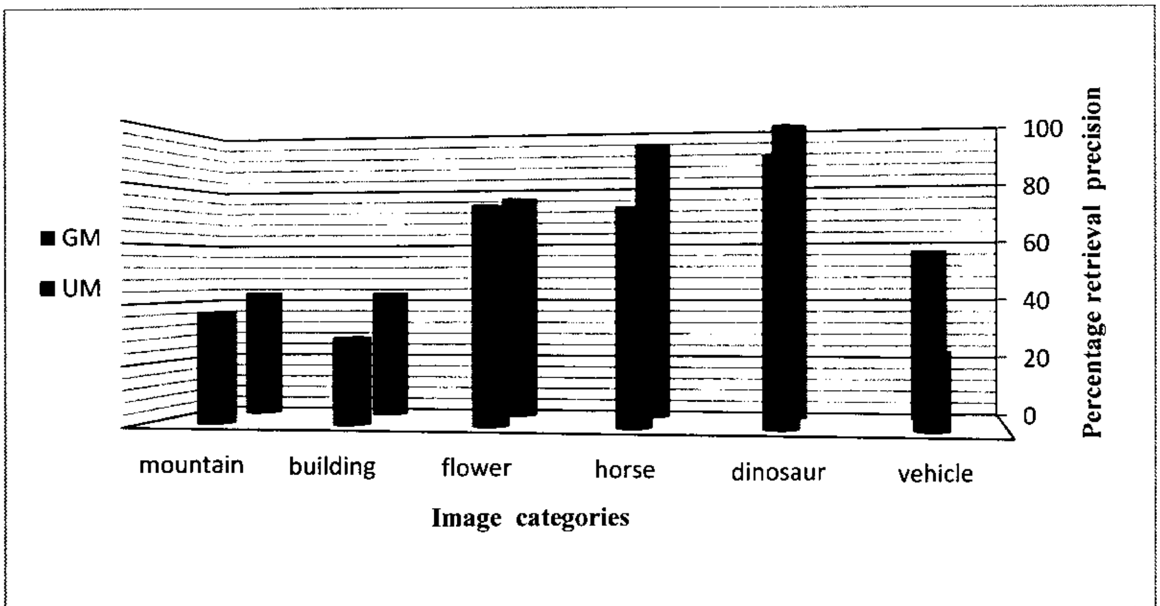


Figure 3-11: Comparison of percentage retrieval precision of proposed global technique with UM for five distinct image categories

The above figure illustrates remarkable improvement in results for images from the 'vehicle' group because of the incorporation of FD and shape features in the system. This particular image category has more edges and boundaries, which were extracted through FD. Also, for 'building' type images, FD and shape features finely perceived edges, resulting in greater retrieval precision.

3.4.11 Comparison of Overall Retrieval Performance

Fair comparisons were ensured by using the same image database as a tester and similar 50 images as queries for comparison with eight different techniques. Average exact retrieval of the selected five image groups from the top 10, 20, 30, 40, and 50 was calculated. It is obvious from the results that our proposed method outperforms all eight techniques in approximately all image categories.

Specifically, our proposed method showed better results compared to the UM method for all image categories except for 'beach' and 'dinosaur' images. NFA's results for 'vehicle' and 'horse' were better compared to our proposed technique, but the overall retrieval results of our proposed technique were greater than NFA [4]. For all the other six methods – UFM method [81], IRM method [82], HSV color histogram technique with two different types of bins (i.e., 32 and 64 bins) [83], EHD method [83] and the color indexing method [84] - our proposed method achieved improved results and better precision values for all types of pictures. Figure 3.12 shows a comparison of whole retrieval precision at 10 returned images for five different image categories using nine different methods.

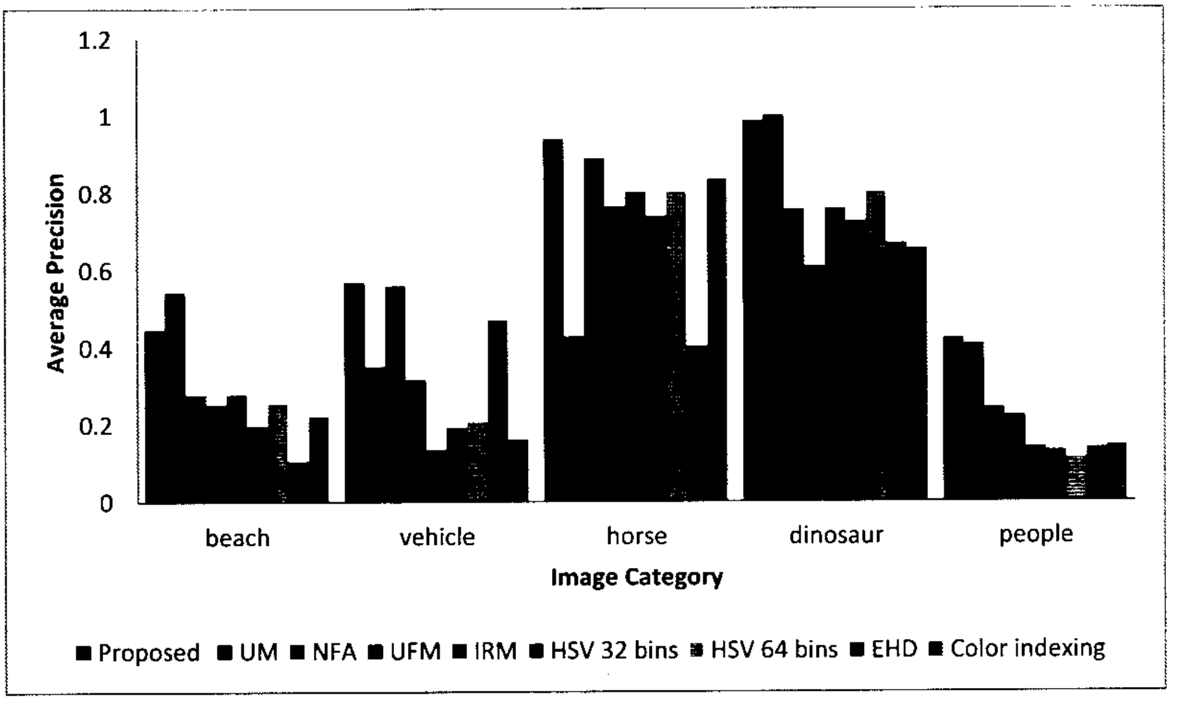


Figure 3-12: Comparison of overall retrieval precision of different image categories by using nine different methods.

Our proposed method did achieve highest precision for two image categories ('beach' and 'dinosaur'), but for all other image categories, the proposed technique had the highest precision. Comparison with UM, NFA and UFM shows that there is not much improvement in results but for other methods like IRM, HSV, EHD and color indexing the difference in the retrieval precision is considerable. The reason for improved results is because of the greater number of features forming a feature vector. In IRM, EHD etc. the number of features used for the calculation of feature vector is less while NFA and UM made use of all the three types of features.

3.4.12 Proposed Global Technique versus Proposed Local Technique

As noted above, the proposed global level method for CBIR achieved better results compared to eight different techniques. We also compared it with our proposed local technique to determine the optimum approach to CBIR. All the 10 different image categories of the Corel image database were used for the comparison. Images were selected randomly from these 10 diverse image groups, each with a different structure and semantics. Table 3.6 shows the comparison of results between the two different proposed techniques for 10 retrieved images from the database.

Table 3-6: Percentage average precision after 10 images are returned for 10 different image categories, considering the proposed local and global approach to CBIR

S.No	Image Category	Global Approach to CBIR (%)	Local Approach to CBIR (%)
1	Vehicle	57.09	38.12
2	Dinosaur	100.00	100.00
3	Horse	94.43	81.25
4	Flower	76.12	72.43
5	Beach	44.23	52.24
6	Building	43.21	37.23
7	Mountain	43.00	42.03
8	People	57.04	42.40
9	Elephant	65.20	48.11
10	Food	51.10	36.05

Results indicated that our proposed global approach to CBIR outperformed our proposed local approach in retrieving the best similar image from the database. The number of features adopted by the two systems was the same but the elimination of segmentation produced improvement in results. Retrieval results were the same for 'dinosaur' as there was only one object in these images and segmentation did not lose much of the information in this case. Using the local method, however, for other image classes like 'food', 'vehicle' and 'elephant' that had a greater number of objects might have resulted in increased loss of information, hence generating reduced retrieval precision.

3.5 Summary

Image retrieval can be performed both at the local and the global level. These two approaches differ in their feature extraction process. The global technique takes into account the whole image in extracting its different features, while the local technique extracts features from the image's segments. Local methods therefore perform image segmentation prior to the feature extraction process. This study proposes two different techniques for local and global CBIR. The proposed techniques were developed for natural images and the Corel image database was used as a data set for the evaluation of these systems.

The proposed local technique starts with a segmentation process based on the color content of the image and the K-means clustering technique was used for the division of the image into meaningful parts. Clusters were restricted to three to avoid over segmentation and loss of data. All the three vital image features (color, texture, and shape features) were then extracted from image segments. Along with these image features, the proposed technique utilized information at the edges by calculating FD and EHD. The feature vector was calculated from all the extracted features and the distance between two feature vectors was calculated through Euclidean distance. The proposed technique showed better results in comparison to UM, which is one of the promising local level techniques for CBIR. Additionally, two variants of UM were also proposed to explore the impact and importance of different image features. One of the modifications made was to add shape features to UM as it currently considers only color and texture features. The addition of the shape feature, however, did not generate any improvement in the results. Retrieval results were better for the second version of UM, where threshold was removed from the system.

The global method on the other hand extracts image features by taking the whole image into account. In the proposed global technique, extraction of image features was done at a global level and the feature vector was calculated for all the main image features such as color, texture, shape, FD, and EHD. The distance between two feature vectors or two different images was measured through Euclidean distance. Precision and recall were used as performance measures to evaluate the retrieval efficiency of the proposed system.

The proposed system was compared with eight different techniques from the literature. Some of these techniques were based on the local approach, some on global, and some, however, were hybrids utilizing both approaches to CBIR. Five different image categories were selected and all the techniques were tested on the same query image. Results showed that the proposed system was better in retrieving the desired image from the database compared to all other eight methods. In addition, comparison of the proposed global method with the proposed local method described was also carried out to determine the optimum system for CBIR. All 10.different image groups of the Corel image database were utilized and the same image was given as a query to both the systems to ensure fair comparison. Results showed the proposed global method for CBIR outperformed the proposed local method for all the image categories except for one.

Further improvements to our system could be added by incorporating other important features, such as local binary patterns that are significant texture features and that can extract useful information about the structural arrangement of pixels. The implication of computational intelligence techniques such as genetic programming and genetic algorithms can also improve the retrieval efficiency of the system.

Chapter 4

4 Content-based Image Retrieval using Uniform Local Binary Patterns

Image features play an important role in the retrieval of required images from huge databases. Local binary patterns are an essential texture feature that has been utilized and adopted by many CBIR systems with different variations. It has greatly improved the retrieval efficiency of these systems. Accordingly, we further improved the CBIR system proposed in Chapter 3 by incorporating this vital texture feature into it. This chapter introduces and explains the calculation of uniform local binary patterns. Its impact on our CBIR system is investigated through experimentation and comparison with other known CBIR systems.

4.1 Introduction

Local binary patterns are the finest method for the extraction of structural properties of texture. Its effectiveness originates in being able to find various micro patterns such as points, edges, and constant areas. Local binary patterns have shown remarkable performance in several applications where texture is an important feature [85, 86, 87]. At present, there are some CBIR systems where different versions of LBP features are included. LBP was first introduced in [88] as a corresponding measure for the local image. It worked with eight neighboring pixels with the center pixel as a threshold. Threshold values are multiplied by weights set by powers of 2 to produce the LBP code. The result is then added to get a final value as shown in Figure 4.1.

A CBIR technique proposed in [89] has utilized local binary patterns in the calculation of the feature vector. Two different approaches were proposed here based on the division of the image and calculation of the LBP histogram. In the first approach, the image is divided into blocks, and then the LBP histogram is calculated from each block. In the second approach, a single histogram of the query image is calculated. A different technique proposed in [90] introduced local derivative patterns (LDP) for the indexing and image retrieval of CBIR systems. Local tetra patterns LTrPs were also introduced in [91] for feature vector calculation, which developed a scheme to calculate n^{th} order LTrP with $(n\text{-}1)^{th}$ order vertical and horizontal derivatives. Compared to LBP, which can encode images with two different values, LTrPs can encode images with four different values, extracting comprehensive information.

4.2 Proposed technique

Image retrieval in our proposed technique commences with a feature extraction process of the whole image. The color feature is extracted through the calculation of a color histogram. Values for the mean, median, mode, and standard deviation were then calculated from the histogram. Different shape features, FDs, and EHDs were also added to the feature vector that helped in object recognition, boundary detection, etc. The imperative texture feature, i.e., uniform local binary patterns, was also calculated. The operator LBP assigns a unique label to each pixel and is invariant to translation and rotation. It functions well with another operator *Var* to add contrast invariance to the system. These operators are discussed in detail in Section 4.2.1. The retrieved results were greatly improved by the addition of the LBP operator. Feature vectors with 288 different features were constructed and stored in an Excel sheet for all the database images. The distance between the feature vectors of the query and database images was calculated through Euclidean distance. A flow chart of the proposed system is shown in Figure 4.1.

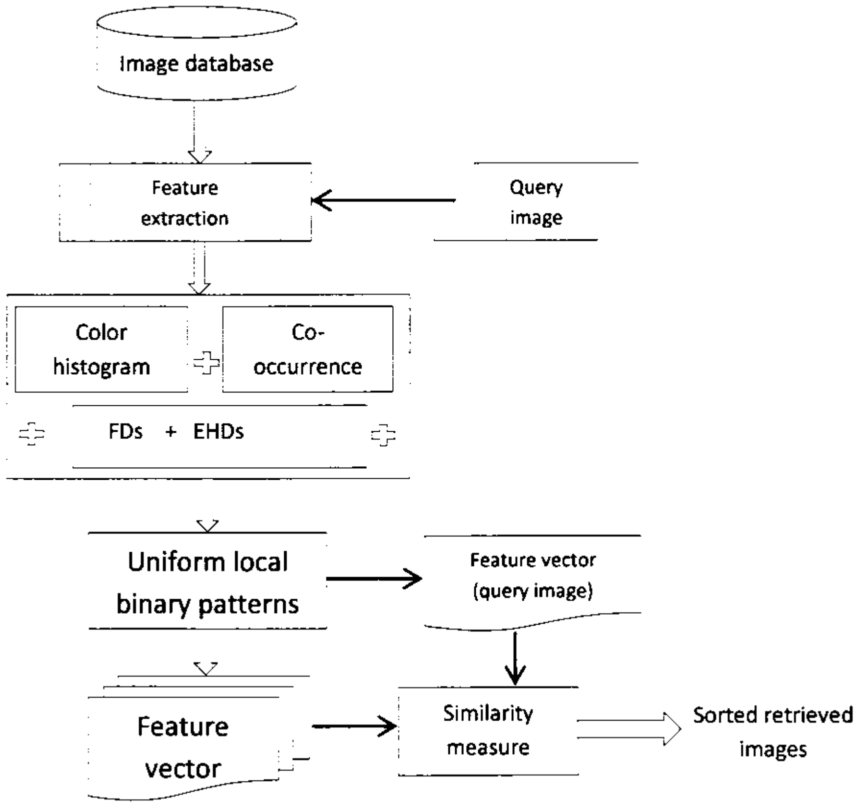


Figure 4-1: Flow chart of the proposed system

Algorithm 4.1:

Step 1: Color histogram Ḣ is created to extract color feature Î: f for each segment of image Î, where i=3.

Step 2: Construct a statistical texture feature Î: f by calculating Co-occurrence matrix $C(i, j)$.

Step 3: Structural information related to image texture is added through uniform local binary patterns. The operator LBP $/Var$ is used for the calculation of local binary patterns, which is represented by the formulas explained in Section 4.2.1.

Step 4: Let $x|k|$ and $y|k|$ be the coordinates of the k^{th} pixel on the boundary of a given 2D shape containing n pixels, a complex number can be formed as:

$z|k| = x|k| + py|k|$,

and the *FD* of this shape is defined as the DFT of Z|*k*|:

$$a(\mu) = \frac{1}{N}\sum_{k=0}^{N-1} z|k|e^{-p\frac{2\pi}{N}\mu k} \qquad (N = 0,\ \ldots,\ n-1)$$

Step 5: The edge histogram descriptor EHD is computed from the four directional edges, that is, horizontal, vertical, 135 degree, 45 degree, and one non-directional edge.

Step 6: Distance between the feature vector of database images *Vd* and the feature vector of the query image *Vq* is measured by Euclidean distance.

4.2.1 Uniform local binary patterns

Uniform local binary patterns are detected at the circular zones of every quantization of angular space. The LBP operator used is valid on a broad range of I members that are circularly symmetric neighbors on a circle having radius R. The performance of each operator is evaluated with specific values of (I,R) and responses of various operators are then combined to analyze multi-resolution. A histogram of these uniform patterns is then constructed over the whole image or a region of an image. Hence, the structural and statistical approaches of texture feature are combined effectively where structural information is extracted through local binary patterns and statistical information is calculated through an occurrence histogram. Spatial structure and contrast are two important properties of image structure. Spatial structure is measured well by the uniform LBP operator but this operator is deficient in calculating contrast. Therefore, it is combined with another operator, i.e., *Var*. Combining these two operators can work well for rotation invariant texture classification. LBP/*Var* then becomes a grayscale, rotation, and contrast invariant operator for texture feature extraction.

Texture $\underline{T}$ is defined as a mutual distribution of gray levels of I image pixels, (I >1), and is given by the equation [92]:

$$\underline{T} = t(s_c, s_0, \ldots s_{I-1}) \tag{4.1}$$

Where s_c denotes the gray value of the center pixel and s_i(i =0,.....I-1) denotes gray values of pixels on the circle of radius R. If coordinates of s_c are (0,0), then coordinates of

s_i are-Rsin(2πi/I) and Rcos(2πi/I). The circular arrangement of neighbor pixels for different values of (I, R) is shown in Figure 4.5.

Subtract the value of center pixel S_c from the value of all the pixels forming a symmetric circular neighborhood [92]:

$$T = t(s_c, s_0 - s_c, s_1 - s_c .... s_{I-1} - s_c) \quad (4.2)$$

It is assumed that $s_i - s_c$ are independent of s_c and $t(s_c)$ does not give any useful information about the texture feature of the image [92].

$$T = t(s_c, d(s_0 - s_c), d(s_1 - s_c), ... d(s_{I-1} - s_c)) \quad (4.3)$$

Where

$$d(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (4.4)$$

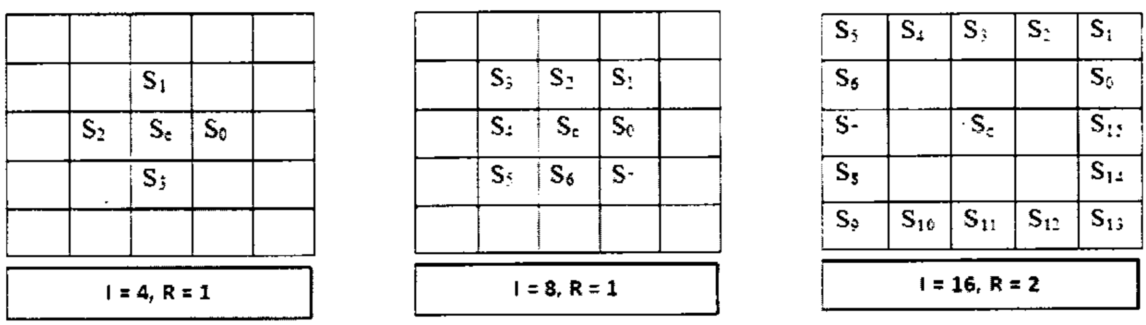


Figure 4-2: Circular arrangement of neighbor pixels for different values of (I, R)

By allocating a binomial factor 2^I for every sign $d(s_i - s_c)$, Equation 4.3 is transformed into a unique $LBP_{I,R}$ number that illustrates the spatial arrangement of image texture [92].

$$LBP_{I,R} = \sum_{i=0}^{I-1} d(s_i - s_c)2^I \quad (4.5)$$

It is observed that some local binary patterns are essential properties of an image's texture that contribute to the huge majority of 3x3 patterns in the image. These essential patterns are called "uniform" and they have a small number of spatial transitions illustrated in Figure 4.3. Patterns that have a maximum two 0 to 1 or 1 to 0 transitions are uniform. Hence the gray scale and rotation invariant operator is given as [92]:

$$LBP_{I,R}^{un2} = \begin{cases} \sum_{i=0}^{I-1} \mathrm{d}(\mathrm{s_i} - s_c)^{\mathrm{I}} & \text{if}\, \mathrm{U}(LBP_{I,R}) \le 2 \\ I+1 & otherwise \end{cases} \tag{4.6}$$

Where

$$LBP_{I,R} = |\, d(s_{I-1} - s_c) - d(s_0 - s_c)| + \sum_{i=1}^{I-1} |d(s_i - s_c) - d(s_{i-1} - s_c)\,| \tag{4.7}$$

The $LBP_{\ I,R}$ operator is invariant to any gray scale transformation and rotation; its output remains constant for the same gray values. Contrast invariance is brought by Var_{IR} and is given by the following equation [92]:

$$Var_{I,R} = \frac{1}{I}\sum_{i=0}^{I-1}(s_i - \mu)^2 \tag{4.8}$$

Where

$$\mu = \frac{1}{I}\sum_{i=0}^{I-1} s_i \tag{4.9}$$

By definition Var_{IR} is invariant in contrast to alterations in gray scale. So the two operators, being complementary to each other, are a powerful measure of image texture when combined.

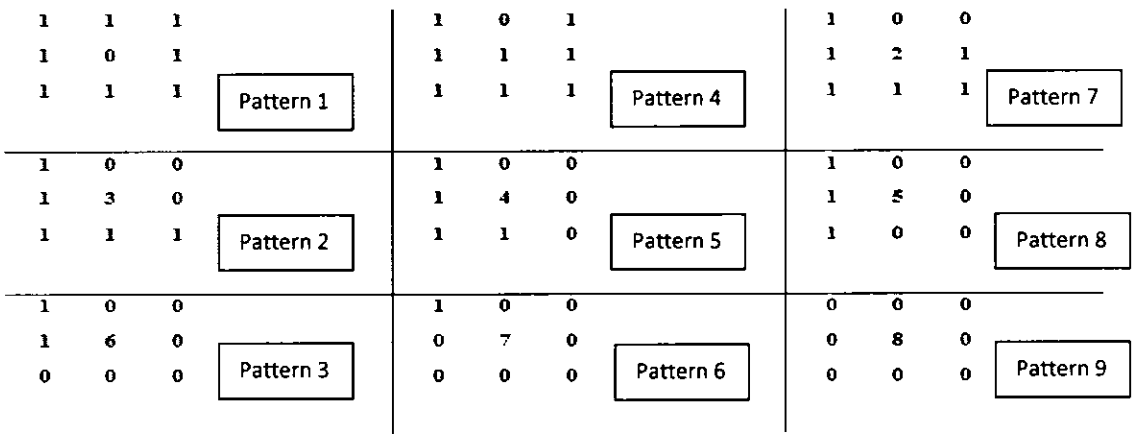


Figure 4-3: Uniform rotation invariant patterns for circularly symmetric neighboring pixels of $LBP_{8,R}^{un2}$ for 8-bit output.

The $LBP_{I,R}$ operator produces *2I* dissimilar output values, matching with 2I different binary patterns that can be formed by I pixels in the neighboring set of pixels. The gray value of s_c will move correspondingly along the perimeter of circle around s_c with the rotation of the image. Figure 4.3 shows the uniform rotation invariant patterns for circularly symmetric neighboring pixels of $LBP_{8,R}^{un2}$ for an 8-bit output. For I = 8, $LBP_{8,R}^{un2}$ can have 36 distinct values. For instance, the first pattern in Figure 4.3 detects a bright spot, the fourth pattern detects edges, and the last pattern detects dark spots and flat areas [92].

4.3 Experimental Results

The proposed system was evaluated by comparing its retrieval meticulousness with other CBIR systems using local binary patterns. Local binary patterns were further worked on in [93] to improve results and local ternary patterns (ICLTP) were introduced. Results found the proposed method as producing optimal results compared to ICLTP. Experiments were performed on 3x3 and 5x5 windows of an 8 and 16 pixels neighbor set respectively. Figure 4.4 additionally illustrates that our proposed method outperformed another CBIR system that introduced local derivative patterns (LDP) [90].

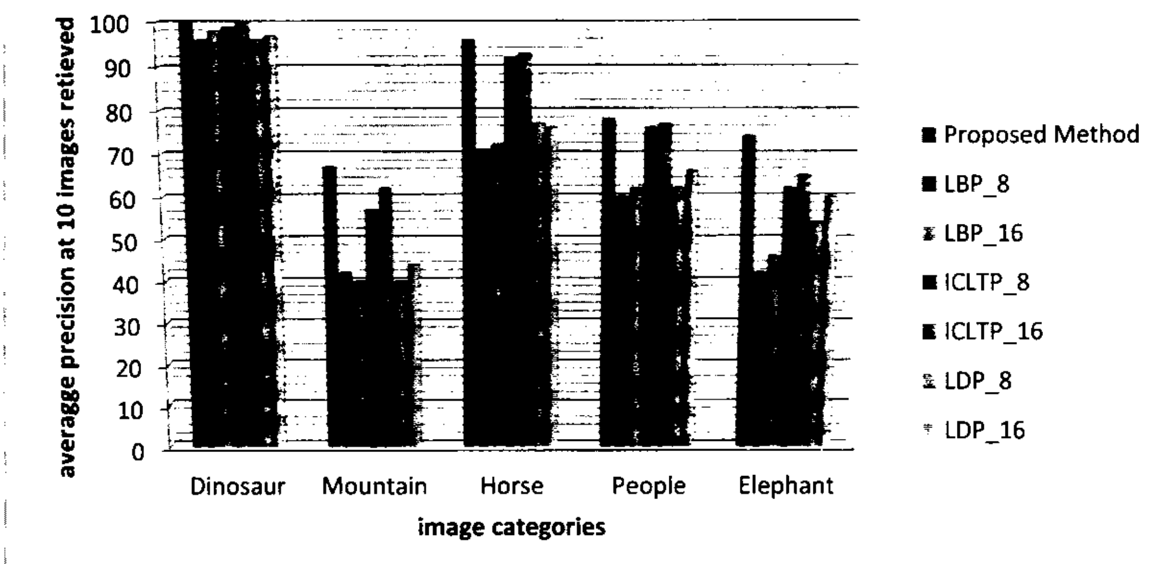
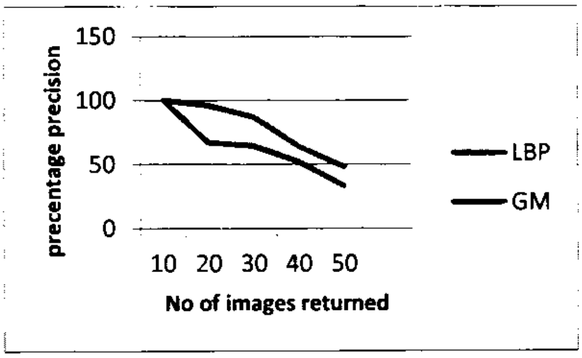


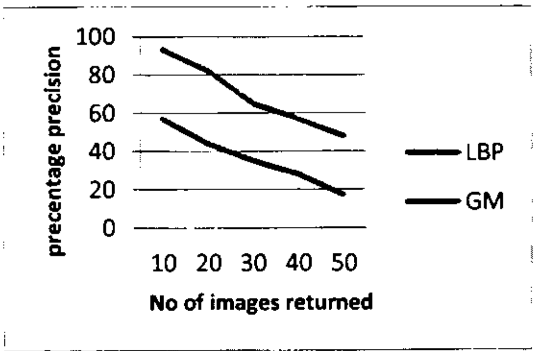
Figure 4-4: Comparison of retrieval precision of proposed technique with six different methods for five distinct image categories

The above figure shows that the proposed method had better retrieval precision for images classes like 'mountain', 'horse' and 'elephant', due to the enhanced feature vector of the proposed method, which had color and shape features, unlike the other CBIR techniques that utilized the texture feature only. The results for 'dinosaur' from all the methods were retrieved with not much difference due to the lesser number of objects in this image group, leading to less information.

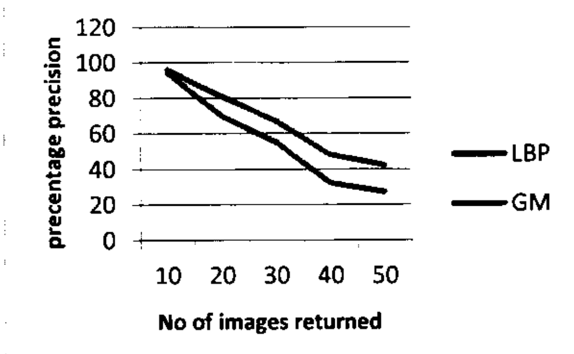
The technique proposed in this study was appraised by finding the retrieval precision for 10 image categories selected randomly. The proposed method was also compared with the proposed global method presented in Chapter 3. Comparison of the proposed method with the proposed global method in terms of average retrieval precision at different numbers of images retrieved is shown through graphs in Figure 4.5.



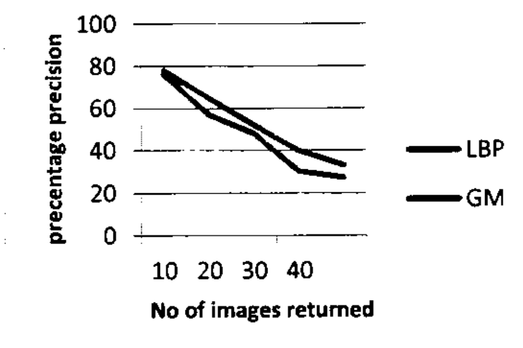
(a) Dinosaur



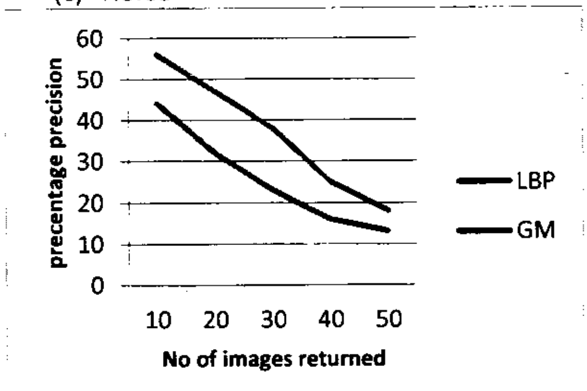
(b) Vehicle



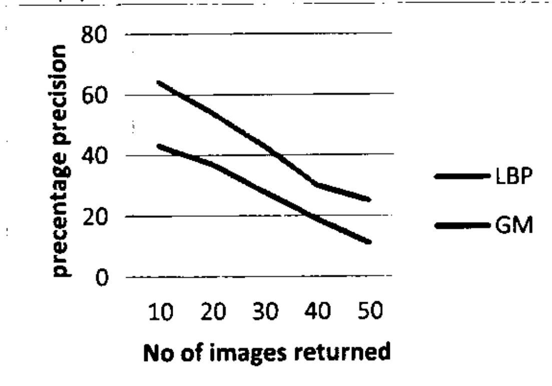
(c) Horse



(d) Flower

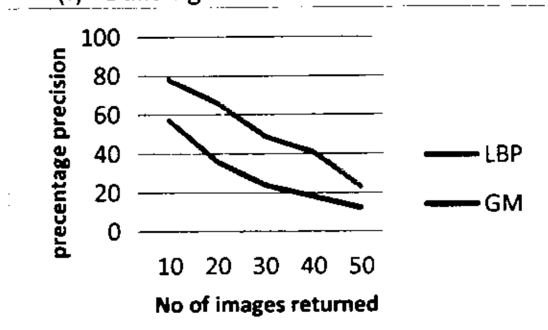


(e) Beach

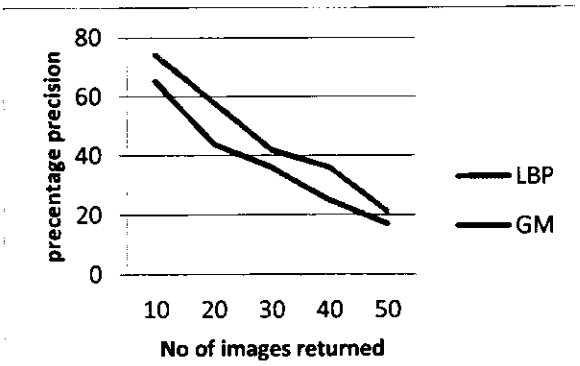


(f) Building

(g) Mountain



(h) People



(i) Elephant

(j) Food

Figure 4-5: Comparison of proposed method with proposed global method in terms of average retrieval precision at different number of images retrieved.

The impact of LBP on retrieval precision is palpable from the graphs shown above. Precision for 'building', 'vehicle' and 'mountain' greatly improved through the addition of uniform LBP. The texture in these image groups was finely minced through LBP operators, hence providing more information to the feature vector.

4.4 Summary

A global level CBIR technique was proposed utilizing all the three basic image features: color, texture, and shape, along with FDs and EHDs. An essential texture feature was added to extract and add information related to the structural arrangement of pixels. The texture property of an image is of two main types, structural and statistical. Structural information is provided by local binary patterns and statistical information is provided by the co-occurrence matrix. A generalized rotation and gray scale invariant operator $LBP_{I,R}^{un2}$ has been developed that detects uniform local binary patterns in the circular neighborhood of quantization of angular space. The feature vector created now is better as all the important features have been added into it. The $LBP_{I,R}^{un2}$ operator is used to calculate uniform local binary patterns that are invariant to gray scale transformation and rotation. It can also be combined with rotation invariant $Var_{I,R}$ that illustrates the contrast of image texture. These two operators were used in combination in the experiments, producing optimal results for image retrieval.

Experiments were performed on a Corel image database with 10 distinct image categories each containing 100 images. Retrieval results of the proposed technique were contrasted with other CBIR techniques. The average precision of the proposed method was higher than the other methods utilizing local binary patterns for image retrieval.

Chapter 5

5 A Hybrid Approach to CBIR

Feature fusion and effective relevance feedback methods can contribute wide-ranging benefits to content-based image retrieval. Feature fusion combines different image features in such a way to get a single feature vector for all of them; however, combining different image features is not always beneficial. The similarity measure plays a vital role in the comparison of image features. Various CBIR systems exploit a single similarity measure for all the features used in the relevance procedure. Nevertheless, the similarity function changes image measure and repetition in a significant way by comparing different image features. Hence, multifarious features (MF) features are proposed in this study with a diverse similarity measure for each group of descriptors. MF features are constructed by the combination different image features with different similarity measures. Genetic programming is also incorporated for the selection of the final distance from MF features. This chapter describes the construction of MF features and the use of the computational intelligence technique. The method proposed here also adds relevance feedback to the system, resulting in improved precision.

5.1. Introduction

Our findings showed that MF features used in the proposed technique improved retrieval results significantly. Adeptness of CBIR can also be improved by inferring active relevance feedback procedures, where response from the user is provided to the system for retrieval of the enquiry image. An extensive system was proposed based on one of the computational intelligence techniques for the effective retrieval of required images from systems. The proposed technique consists of two main modules, namely: feature vector processing and result enhancement, as shown in Figure 5.1. The major tasks performed in these methods are as follows.

- Abstraction of image structure through extractor solicitation and permutation of feature space by formation of multifarious features.

- Combination of dissimilarity measures acquired as a result of multifarious features comparison by application of genetic operators to find the final distance between probe and stored images.
- Subsequently estimated retrieved images are presented to the user, first k- images will be selected by the user as K-NN query images and are ranked according to their relevance values provided by the user.
- Optimization of retrieved images by iteratively providing user's feedback to the system.

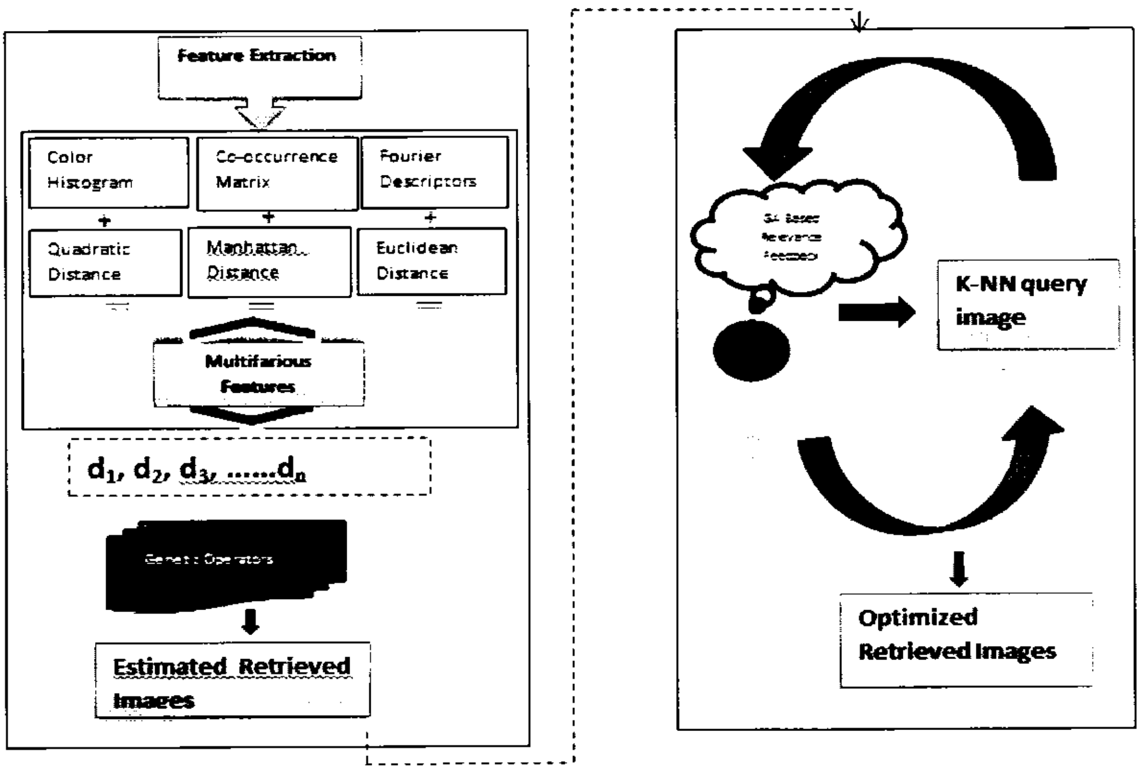


Figure 0-1: Diagrammatic representation of proposed hybrid method

Algorithm 5.1:

Step 1: An image $\bar{I}$ is considered as a pair $(f_I, \hat{I})$,

Where: f_I is a predetermined set of pixels,

$\hat{I}: f_I \rightarrow f'$ is the function that assigns a vector $\hat{I}(P)$ to each pixel in f_I.

Step 2: A simple MF feature is given by a pair (V_f, δ_f) where:

Where V_f is a function to extract feature vector from image and

δ_f is the similarity measure.

- $\dot{H}$ with Quadratic distance *QD(x, y)*
- *C(i, j)* with *Manhattan dist (x, y)*
- DFT with Euclidean dist (x,y)

Step 3: Genetic programming GP is implemented on the distances from the comparison of MF features.

Step 4: Let d= {di} be a series of similarities in any of the three feature spaces. Gaussian normalization gives the mapping $d_i \rightarrow (d_i - \mu)/\sigma$ where μ and σ are mean and standard deviation.

Step 5: Images retrieved from the system are then presented to the user and user evaluates the first K images by ranking according to their relevance with the query image.

Three different values are used to acquire the user's response.

Relevant: $(q_k) = 1$,

Irrelevant : $\gamma(q_k)$ = 0 and not required: $\gamma(q_k)$ = -1

5.2. Feature Fusion

Advancement of features used in image retrieval has resulted in studies on combining different features in CBIR. Features can be fused together by linearly combining different similarities. This method has good results with less training computation. Several features are used in image retrieval, such as color histograms, co-occurrence matrix measures, Tamura features, and Gabor features. Experimental comparison has shown that appearance based image features, a co-occurrence matrix, and SIFT have given better results. These three methods are combined for better results. Gaussian normalization is used for each similarity of three similarities using the above three features [94].

Gaussian normalization is used for each similarity. Let d= {di} be a sequence of similarities in any of the three feature spaces. Gaussian normalization results in the mapping $d_i \rightarrow (d_i - \mu)/\sigma$ where μ and σ are mean and standard deviation [94].

The retrieval process can be made efficient by fusing different image features. Color and texture features are fused to design a CBIR system by constructing weights of the feature vector. The process can be performed in three steps. First is the creation of color histograms of an image. As extraction of colors is not enough for image retrieval, the texture feature is also calculated. The image is then converted into grey scale and the co-occurrence matrix is calculated for it. These two features are then fused together by determining their weights for better image retrieval [95]. Image retrieval from large databases can be made easy by clustering similar images and comparison of the query image with a cluster rather than searching the whole database. A technique has been proposed in which clustering is performed through artificial neural networks. Similar images are clustered based on image features. Color and texture features are used for clustering [96].

5.3. Multifarious (MF) Features

Most CBIR systems are designed to retrieve images by combining image features regardless of the similarity functions, but these methods are meaningless. Similarity functions have an important part in making descriptors as invariant as possible to change in image scale and rotation. For example, Bean angle statistics (BAS) and multi scale (MS) fractal dimensions are shape descriptors that cannot be combined irrespective of their similarity function. Different feature extraction algorithms are defined that encode segment saliencies (SS) and BAS with optimum correspondent subsequence (OCS) and Euclidean metric as similarity functions [19]. Both BAS and SS give better results when used with OCS as a similarity function and MS fractal dimension is better when combined with Euclidean distance.

The influence of each image feature is enhanced by combining the image feature with its respective similarity function. This grouping of image feature with its corresponding similarity measure creates multifarious (MF) features. Our proposed technique makes use of MF features for higher performance in comparison with the existing CBIR techniques. For the calculation of MF feature vector, color, shape, and texture features are compared and evaluated separately with an appropriate distance measure. Their results are fused in the end for retrieval of the required image from the database. Below is a description of the MF features utilized in the proposed system.

5.3.1. Color Histogram with Quadratic Form Distance

Color is the most vital and extensively used characteristic of an image. We have employed this feature in our proposed system as it extracts some basic information about the color distribution of an image and different objects present in that image at different localities. Visually, color is the most basic feature through which images can be distinguished from each other. Color features can be shown through mean color, prominent color, and color histograms. The color histogram is a multi-aspect feature vector and is computationally demanding, as it is well appropriate for global color regions.

Quadratic form distance produces more desirable results compared to other distance measuring techniques [41]. Histogram comparison for retrieval of images based on color content has been tried with a number of distance measures; here quadratic form distance generated better results compared to other distance measures such as Euclidean distance, histogram intersection, and Minkowski distance. Many CBIR systems use quadratic form distance for retrieval of images based on color histograms. Cross similarity between colors is considered while finding dissimilarity between the color histograms of different images; therefore two images with the same histogram may differ in contents [97]:

$$d^2(\mathrm{Q},\mathrm{I}) = (H_Q - H_I)^t \, A \, (H_Q - H_I) \qquad (5.1)$$

where d represents the distance between two color images, i.e., the query image Q and a database image I. A is the similarity matrix and t is the transpose. H_Q and H_I are the histograms for the query image and database image respectively.

5.3.2. Co-occurrence Matrix with Manhattan Distance

One of the major contents of images used for categorization in CBIR is texture. This is usually represented by the geometric distribution of image intensity. This calculates the gray level distribution of pixels in an image by the application of different statistical methods. Our proposed technique considers a set of important texture features because they represent structural and statistical important characteristics that do not exist in other types of features, namely color and shape [98].

Texture features were compared with different distance measures such as Euclidean, Manhattan and Chebyshev in [99]. Among all the distance measures tried with texture features, Manhattan distance (MD) appeared to have the best similarity measure [99] and is represented by the following formula:

$$Manhattan\ dist = \sum_{i=1}^{n} |X_i - Y_i| \tag{5.2}$$

Where X and Y represent the query and database image.

5.3.3. Fourier Descriptors with Euclidean Distance

Images recovered on the basis of shape are the yardstick to measure likeness between shapes of the query image and feature based database images. Shape is not only a vital visual feature but is also among the primal characteristics for image content description. Depiction of shape is difficult to calculate because to figure out likeness amid shapes is complex. Two main techniques used for shape feature extraction are contour-based and region-based [100]. The former is used to extract information from the boundary of an image and the latter is used to extract information from the interior. The former uses merely the given in the outline of an object, but the latter uses the whole area.

FDs are finest measured through Euclidean distance (ED) [19]. Euclidean distance is given by the formula:

$$\text{Euclidean distance} = d = \sqrt{\textstyle\sum_{i=1}^{n}(x_i - y_i)} \tag{5.3}$$

Where d calculates distance, x is the query image, and y is the database image for all the shape features ($i = 1,2,3..n$).

5.4. Image Retrieval Based On Genetic Programming

The GP technique is used to combine the similarity values obtained from each MF feature and fuse them into an effective distance measure. Obtaining a distance measure from different MF features by GP is an iterative process. At first, GP works on a huge population of indiscriminate combination functions that are then estimated on the basis of relevance information and training from images. Genetic transformation systems such as reproduction, mutation, and crossover are used for the modification of population individuals. The best individuals are selected and copied to the next generation by the reproduction operator. Variation is brought by mutation and crossover operators. Mutation operates on one individual while crossover operates on two individuals [19].

The process of creation of MF features and calculating the final distance through GP is shown in Figure 2. In Figure 5.2, image features are represented by If_i, where i = 1,2,3…n for n different image features, similarity measures are given by δm_i, where i = 1,2,3,…m form different distance measures. They are paired together to form MF features. Distance from MF features is represented by $\hat{c}_1, \hat{c}_2, \hat{c}_3, \ldots \hat{c}_d$. GP combines these distances to calculate $F\hat{c}$, where $F\hat{c}$ is a mathematical expression characterized as an expression tree. Non-leaf nodes of this tree are numerical operators and leaf nodes are distance values, $\hat{c}_i$, i=1,2,…..,d. In a GP based retrieval framework the population starts with random creation of individuals and genetic operations are applied for the evolution of this population. A fitness value is assigned to each individual in a population by using certain fitness functions. Fitness functions used in experiments in the proposed method are FFP1, FFP2 and FFP3 [85].

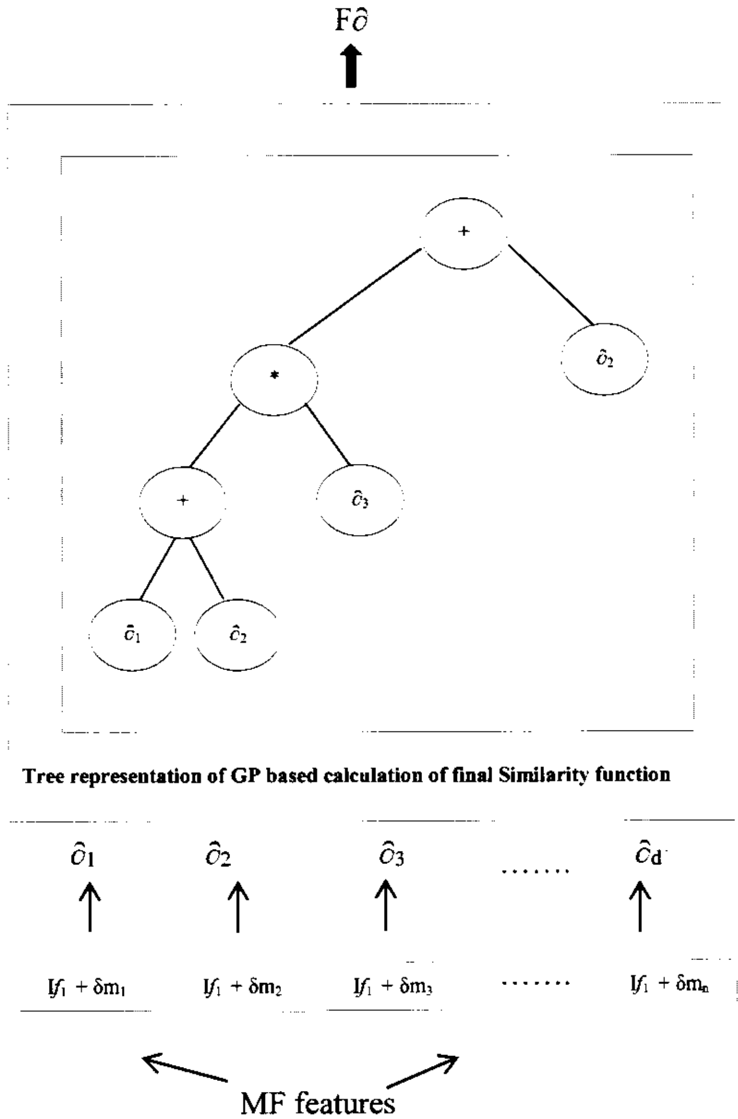


Figure 0-2: Calculation of final distance measure of MF features

5.4.1. Functions and Operators of GP Method

GP uses more complex data structures compared to genetic algorithm techniques. The GP tree structure is used by this study's proposed method for the evolution of population by application of different genetic transformations. Some basic terms and symbols used in the system are given below:

An image $\bar{I}$ is considered as a pair $(f_I, \hat{I})$, where:

- f_I is a predetermined set of pixels,
- $\hat{I}$: $f_I \rightarrow f'$ is the function that assigns a vector $\hat{I}(P)$ to each pixel in f_I.

A simple MF feature is given by a pair of feature vectors of an image and its similarity function that calculates similarity between two images. It is given as (V_f, δ_f) where:

- Where V_f is a function to extract a feature vector from an image and
- δ_f is the similarity measure.

Multifarious features will result in the creation of a composite feature and is given by $(Đ, \delta_Đ)$ where:

- Đ is a set of simple descriptors and
- $\delta_Đ$ is the similarity function .

Initial population: Generation of an initial population is carried out by a random process which produces a set of trees randomly generated with a constraint to be four levels deep at maximum. Terminals of the GP trees are the similarity functions corresponding to their respective descriptors. These terminals are joined by a set of functions extensively used in GP experiments. Functions used by the proposed method to join terminals and sub trees are +, ×, /, sqrt. The function set is based on an existing GP technique that has shown good results for image features and similarity measures as leaf nodes of GP tree. Effectiveness of the combination function characterized by an individual tree is measured through the fitness function.

Fitness Function: A good fitness function is required to avoid local optimum and to efficiently explore the search space. Below are the formulas of fitness functions used with a brief description.

$$funcFFP1 = \sum_{i=1}^{|N|} x\,(Ii) \times sf_1 \times \ln^{-1}(i + sf_2) \tag{5.5}$$

Where i is the position of the image after it has been retrieved by the system. I is the image at position i. "$x(I\text{i})$" is the degree of relevance of an image. The total number of images retrieved is given by |N|. sf1 and sf2 are the scaling factors and are given the values as follows for our experiments, sf1 = 6, sf2 = 1.2 [101].

$$funcFFP2 = \sum_{i=1}^{|N|} x\,(I\text{i}) \times sf_3 \times log_{10}(1000/i) \tag{5.6}$$

where (Ii) and |N| are the same as in Equation (2). $sf_3$is a scaling factor and is set to sf_3 = 2 in our experiment [21].

$$funcFFP3 = \sum_{i=1}^{|N|} x\,(I\text{i}) \times sf_4^{-1} \times e^{-sf_5 \times ln(i) + sf_6} - sf_7 \tag{5.7}$$

where (Ii) and |N| are the same as in Equation (2). sf_4, $sf_5$and $sf_6$are scaling factors, their values are set as 3.65, 0.1, 4, and 27.32, respectively [102].

Genetic transformations, for instance reproduction, mutation, and crossover are applied on the individuals of the population for modification. The best individuals are selected and copied to the next generation by reproduction operation. Variation, on the other hand, is brought by mutation and crossover. In crossover, trees are randomly selected from the current population and a pair of trees with the highest fitness is brought together for the exchange of sub-trees. Mutation in individuals is carried out by selecting a point where the sub-tree is removed and replaced by another randomly generated sub-tree. GP operations are carried out till 25 generations.

5.5. Relevance Feedback

The semantic gap is reduced by the implementation of relevance feedback (RF) provided by the user. RF methods are very useful for CBIR systems as they let the system learn the best features that should be considered for effective retrieval. During the RF process, the user evaluates the retrieved images by assigning them values based on their relevance with the query image. After getting feedback from the user, the whole process of image retrieval is repeated taking into account the feedback given by the user. One drawback of RF is that it is considered a search problem because of the parameters, weights, and data combination models, such as functions combining multiple features. A technique based on relevance feedback was proposed in [103] where weight correction of features was done by a set of rules using mean and standard deviation of the feature vector of relevant and irrelevant images. Then, for each feature, weight was adjusted according to the ranking provided by the user for each image.

To cope with search problems, the genetic algorithm (GA) technique has been employed in previous studies. In the proposed method, a GA based RF method is used to enhance the image retrieval task. Weighting functions are used to adjust the dissimilarity function. These weighting functions adjust the effect of each feature in a known feature space.

Images retrieved from the system are then presented to the user and user evaluates the first K images by ranking them according to their relevance with the query image. Three different values are used to acquire the user's response:

Relevant: $(q_k) = 1$,

Irrelevant: $\gamma(q_k) = 0$ and

Not required: $\gamma(q_k) = -1$

This is an iterative process which gives a set of relevant images ρ_q. The distance function is adjusted by a series of f weighting functions to reflect the feedback provided by the user. These values are utilized in the fitness function of GA [15].

5.5.1. GA Framework for Optimized Results

An integer value is assigned to code a chromosome having *n* different points where each point matches an individual of a function in the feature space. These chromosomes provide a new transformed feature space.

A fitness function employed from the measure of ranking quality Ƥ that has been calculated in after KNN query images *kNNi* are retrieved is given as follows:

$$func4[\varphi(Ƥ_i, \rho_q)] = \sum_{k=1}^{n} \frac{\gamma(q_k)}{P}\left(\frac{(P-1)}{P}\right)^{k-1} \quad (5.8)$$

Where ρ_q is the set of objects that are relevant to the set of feature vectors taken out from the query image. If q_k is a subset of ρ_q, then

$\gamma(q_k) = 1$

Otherwise

$\gamma(q_k) = 0$

A larger value of P will not be considered strongly as it gives the relative status of the position of the images in rankings. Smaller values for P show the importance of relevant images.

Variation is brought by crossover and mutation. A mask is generated randomly to indicate the use of chromosomes that will provide the genes for the first generation. A compliment of the mask is then used to produce the second generation. Offspring chromosomes are subject to mutation where probable genes are selected and their values are replaced by a randomly generated valid value [104].

5.5.2. Fitness Function

GP has been tried for extraction of MF features. Table 5.1 shows the average retrieval precision of each feature with a distinct fitness function. Results revealed that *func*3 gives better results with GP for all the three features in comparison with other fitness functions.

Table 0-1: Comparison of average retrieval precision of each MF feature using three different fitness functions

Multifarious features	GP with *funcFFP*1	GP with *funcFFP*2	GP with *funcFFP*3
Color with QD	80.4	81.0	85.3
Co-occurrence matrix MD	70.5	78.4	82.6
Fourier descriptor with ED	81.2	82.5	83.0

5.6. Experimental Results

Experimental setup:

The experimental setup use for GP technique is given below.

Population size	**20**
Generations	50
Crossover	0.8
Mutation	0.01
Function set	+, -, x, /
Terminal	Image features and similarity measures
Depth of tree	10
Stopping criteria	Precision = 72%

Two different methods were used to perform experiments: the direct weight generator (WG) and the weighting method (WM). The fitness function defined for GA was employed by these functions. Parameter P was tested with 10 and 20.

Regarding KNN, 30 nearest neighbor images to the query were selected and 10 cycles were given to the relevance feedback process. Results were analyzed by finding the precision and recall for the two different methods selected for experimentation. Figure 5.3 shows the number of relevant images retrieved using WG and the proposed WM. It shows that WM outperforms WG for both values of P of the fitness function. A maximum number of relevant images was retrieved for P = 20 in our experiments. Figure 5.4 shows the graph for a number of relevant images retrieved for five different categories of images. These different categories were selected from the database because of their diverse nature. The content for each category of images differed from one another. These various image types were selected for the better analysis of results given by the proposed system.

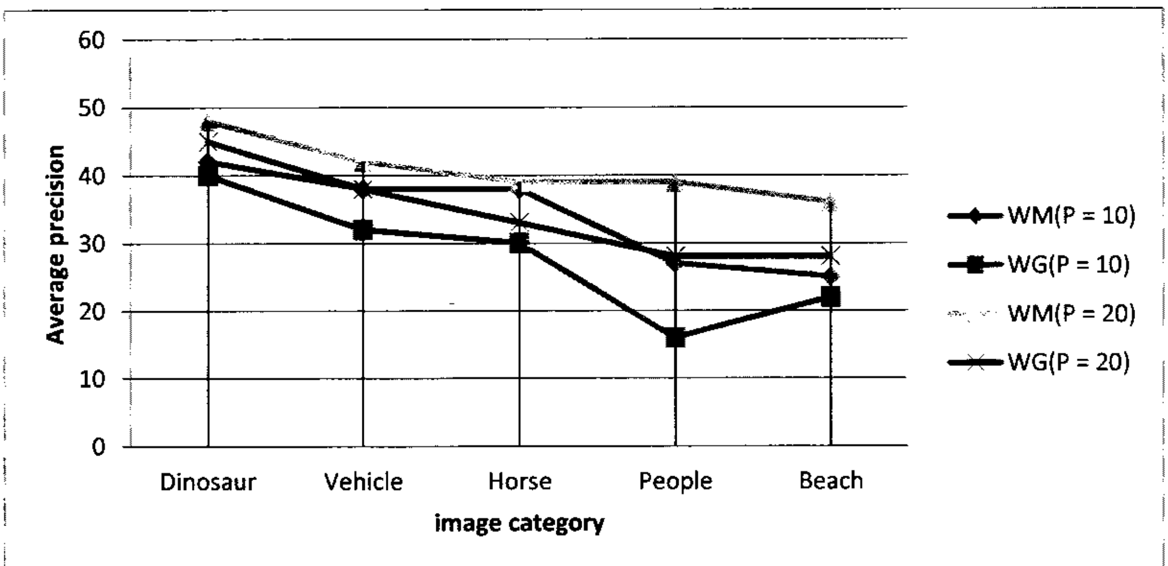
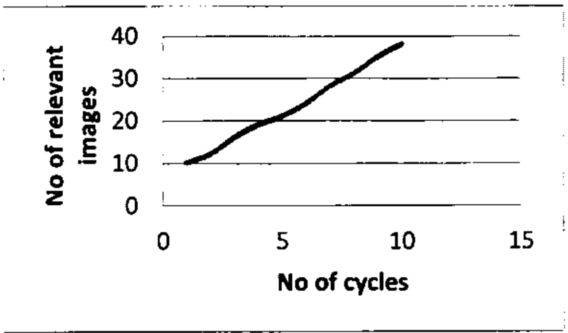
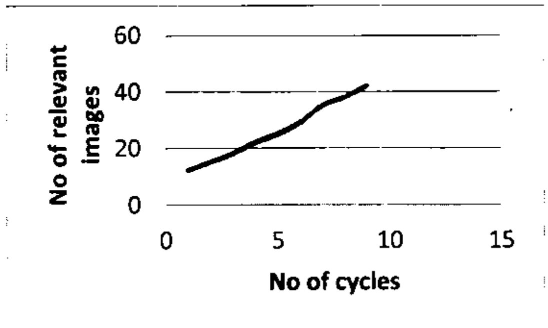


Figure 0-3: Number of relevant images retrieved for each image category using WM and WG

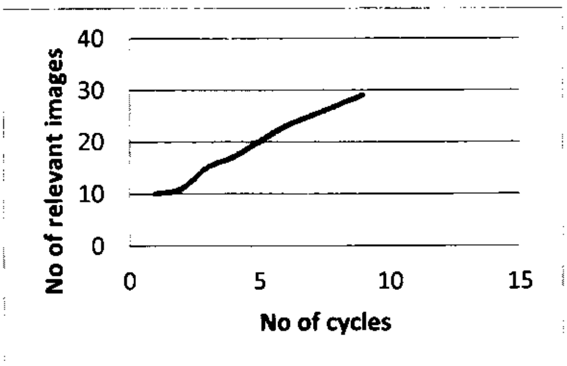
The retrieval results showed that the proposed method with p = 20 for both WM and WG was superior compared to p = 10 as the fitness function had great impact on the selection of fused features for the creation of the feature vector. The precision for 'dinosaur' and 'vehicle' was higher because of the lesser number of objects in these image groups. However, precision for 'beach' and 'people' was a bit low as images were congested with more objects demanding for more information retrieval.



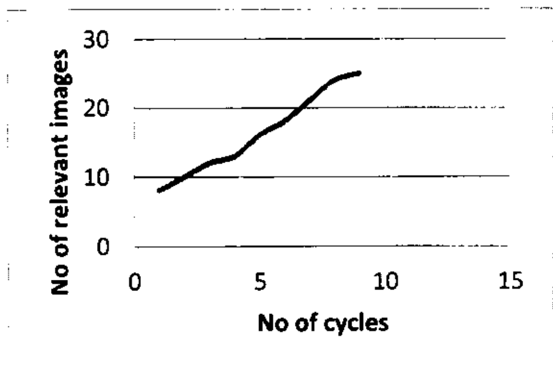
(a) Vehicle



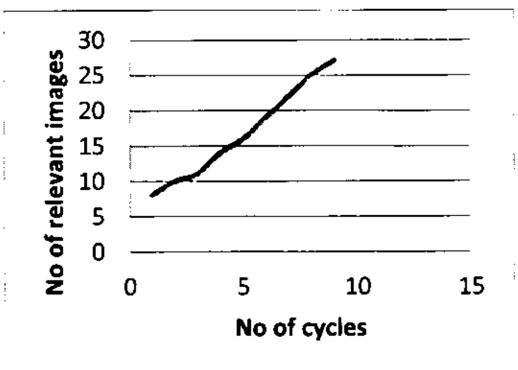
(b) Dinosaur



(c) Horse



(d) People



(e) Beach

Figure 0-4: Graphics for the number of relevant images retrieved in 10 cycles for five different categories of images

Adding relevance feedback to the system raised the retrieval precision by incorporating user's feedback into the system. The above graph shows the increase in the retrieval of number of relevant images with the increase in the number of cycles.

5.7. Comparison of Overall Retrieval Performance

Fair comparisons were ensured by using the same image database as a tester and 50 alike images as queries. Average retrieval of the selected three image groups from the top 10, 20, 30, 40, and 50 was calculated. Table 5.2 compares different image categories and techniques in terms of average retrieval precision. It is obvious from the results that the proposed method outperformed all eight techniques in approximately all image categories.

NFA's results for 'vehicle' were better compared to our proposed technique but the overall retrieval accuracy of our proposed technique was greater than NFA [4]. When compared to all other four methods – the UFM method [81], IRM method [82], EHD method [83], and the color indexing method [84] - our proposed method had improved results and better image retrieval accuracy for all types of pictures.

Table 0-2: Comparison of average retrieval precision of the proposed method with five different methods for four distinct image categories

Image Category	Proposed	NFA	UFM	IRM	EHD	Color Indexing
Beach	0.3670	0.2800	0.2533	0.2800	0.1067	0.2233
Vehicle	0.4267	0.5600	0.3167	0.1367	0.4700	0.1633
People	0.3940	0.2433	0.2233	0.1433	0.1400	0.1467
Average	0.39	0.36	0.26	0.1833	0.2366	0.1733

5.8. Summary

The global level technique discussed in Chapter 3 was further improved by the integration of computational intelligence techniques and implementation of relevance feedback. Firstly, image feature extraction and feature vector calculation were performed utilizing all the major image features. The same database of natural images, that is, the Corel image database, was used for experimentation purposes.

In this study, we applied the idea of feature fusion by introducing multifarious features. The main idea of these features lies in the combination of image features and the similarity distance. Three different combinations of image features and similarity measures are used by the proposed system. Here, a different similarity measure is used for each image feature. The same distance measure used for all image features might not produce good results; therefore each distance measure was combined with a feature that works well with that distance measure more than any other measure. Distances from the multifarious features were combined through different operators by the implementation of the genetic programming technique and the final distance was then retrieved after genetic operations. The final distance compared the two feature vectors and found the most similar image from the database.

The semantic gap was reduced by the assimilation of relevance feedback. Retrieved images were subject to relevance feedback and images were ranked based on the user's input to the system. Feature space thus obtained by the k-nearest neighbor query images was transformed through a genetic algorithm to get optimized retrieved images from the system. Different image features in combination with different similarity measures can be analyzed and the transformation functions used can be further investigated in future research studies.

Chapter 6

6. Conclusions and Future Work

Content-based image retrieval is an important research area that is yet to be worked upon. It has been applied in only a few systems and there is a great need still to realize the potential benefits of this technology. The most commonly used search engines such as Google, Yahoo, and MSN have already incorporated image retrieval into their systems. However, the technique used for image retrieval in such systems is text-based where a user is required to enter keywords to search for images. These systems do not take the visual content of the image into account for searching and comparison. Content-based image retrieval is a highly challenging field in research with strong research efforts presently being concentrated on several characteristics of image retrieval technology.

There are many issues related to the CBIR system regarding the efficient and accurate retrieval of images. The literature study described in the beginning of this thesis illustrated that image features is one of the major issues. Features play an important role in describing the content of the image and therefore selection and extraction of these features greatly affect retrieval results. Most CBIR systems are based on features that are already used in other systems. Research should be conducted to find and extract some more informative features that can improve the efficiency of these systems. Another important issue is the semantic gap. Image features are low level and users' expectations are high. The quest to bridge this gap between the conceptual interpretation of images by users and low level visual features of images is the most active research area in CBIR systems. A user may be interested in a certain part of the image. This part may have some semantic properties such as objects or scenes. However, such properties are not well represented by the visual features of the image. Translation of these low level features to the high expectations of users requires more information, which can be provided by the users' interaction with the system.

Three different frameworks of CBIR are proposed in this thesis. These techniques are based on the use of multiple image features in combination with different similarity measures while integrating the implication of relevance feedback.

6.1. Conclusions

The first method proposed in this study was based on the local approach to CBIR. In this method, image features were extracted at the local level. The image was first divided into different parts by using the K-means clustering technique based on the color content of the image. We made use of all important image features, such as color, texture, and shape. In addition, EHD was also calculated to extract maximum information from the image. The similarity measure used was Euclidean distance which was a single measure used for all the features. The proposed system was compared with UM, which is one of the local level CBIR systems proposed in 2008. Results showed the proposed system to be better in retrieving the desired image from the database compared to UM. Experiments were performed on natural images from a Corel image database of 1000 images. Images from 10 different categories were selected for the evaluation of the system. The performance measures used were precision and recall.

The second method proposed in this work was based on the global approach to CBIR. In this method, image features were extracted from the image as a whole. All the image features described for the local approach in Chapter 3 were utilized for the global approach as well. Euclidean distance was used to measure the distance between two feature vectors. Experiments were performed on the same database of images and retrieval effectiveness was found by taking precision at 10, 20, 30, 40, and 50. The proposed method was compared with eight different CBIR techniques. Results found the proposed global level technique to outperform all other techniques from the literature, for all 10 different categories of images from the Corel image database. This technique was then compared with the proposed local technique and results were much better for the former than the latter. This observed superiority of the global approach to the local approach might be due to the loss of information during the segmentation process in the local approach. Additional improvements could be brought by incorporating relevance feedback.

In the third method, the proposed global technique for CBIR was further improved by introducing multifarious features. Here, different image features were combined with different similarity measures and the distances were combined by implementing the genetic programming technique to find a single distance measure. In addition, relevance feedback was also integrated into the system to reduce the semantic gap. Retrieved images were subject to relevance feedback and images were ranked based on the user's input to the system. Feature space thus obtained by the k-nearest neighbor query images was transformed through a genetic algorithm to obtain optimized retrieved images from the system.

6.2. Future Work

Several frameworks have been proposed in this thesis as having the ability to enhance the retrieval results of CBIR systems. There are many other issues that need to be resolved for future enhancements in CBIR systems. Image features are of great importance in CBIR systems. Research has to be carried out to find and explore other image features than those currently used by many systems. Also, the number of multifarious features proposed in this thesis can be increased by further investigation of different image features in combination with different similarity measures. The local approach proposed here can further be improved by the application of a proper segmentation technique to avoid any loss of information and to extract all the image features appropriately from each segmented part of the image.

The genetic programming incorporated in the proposed system can further be improved by investigating other fitness functions and trying different genetic operations for better results. Selection of important image features through a good selection method such as PCA and neural networks can be incorporated into these systems. This will increase the efficiency of the systems by counting image features that are of greater importance compared to other features.

Most CBIR systems are based on static image databases where the numbers of images are fixed and where feature vectors created are stored in the systems. Scalability can be brought by developing a system for dynamic databases where images are added and removed constantly.

References

1. T. Deselaers, "Image retrieval, object recognition, and discriminative models," Ph.D. dissertation, RWTH Aachen University, Aachen, Germany, 2008.
2. R. S. Choras, "Image Feature Extraction Techniques and Their Applications for CBIR and Biometrics Systems," *International Journal of Biology and Biomedical Engineering*, vol. 1, no. 1, 2007.
3. M. Aly et al., "Automatic discovery of image families: global vs. local features," in *Proc. 16th IEEE Int. Conf. Image Processing*, 2009, pp. 777–780.
4. X. Qi and Y. Han, "A novel fusion approach to content-based image retrieval," *Pattern Recognition*, vol. 38 pp. 2449–2465, 2005.
5. W. Muller et al., "Strategies for positive and negative relevance feedback in image retrieval," in *Proc. 15th Int. Conf. Pattern Recognition IEEE*, 03 Sep 2000, pp. 1043–1046.
6. Y. Sun et al., "Myphotos - A system for home photo management and processing," in *Multimedia Conf.*, France, 2002, pp. 81-82.
7. H. Armitage and P. G. Enser, "Analysis of user need in image archives," *Journal of Information Science*, vol 23, no. 4, pp. 287–299, Apr. 1997.
8. I.K. Sethi and I.L. Coman, "Mining association rules between low-level image features and high-level concepts," in *Proc. SPIE Data Mining and Knowledge Discovery*, 2001, pp. 279–290.
9. I. F. Rajam and S. Valli, "A survey on Content based image retrieval," *Life Science Journal*,vol 10, no 2, pp. 2475-2487, 2013.
10. P. de Vries and T. Westerveld, "A comparison of continuous versus discrete image models for probabilistic image and video retrieval," in *Int. Conf. Image Processing*, Singapore, 2004, pp. 2387-2390.
11. Antani et al., "A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video," *Pattern Recognition*, vol 35, no. 4, pp. 945-965, 2002.
12. Vailaya et al., "Image classification for content-based indexing," *IEEE Transaction on Image Processing*, vol. 10, no. 1, pp. 117-130, Jan. 2001.

13. Z. Lijia et al., "Learning based combining different features for medical image retrieval," in *Int. Conf. Image and Graphics*, 2009, pp. 969 - 972.
14. G. Ohashi and Y. Shimodaira, "Edge-based feature extraction method and its application to image retrieval," *International Journal of* Systemics, *Cybernatics and Informatics*, vol. 1, no. 5, 2003.
15. L.P.S. Avalhais et al., "Image retrieval employing genetic dissimilarity weighting and feature space transformation functions," in *Proc. ACM Symposium Applied Computing*, 12 March 2012, pp. 1012-1017.
16. S. Bhattacharjee and S. Borah, "A survey on the application of fuzzy logic controller on dc motor", *International Journal of Application or Innovation in Engineering and Management (IJAIEM)*, vol. 2, no. 6, pp. 84 – 90, June 2013.
17. A. Nagathan and I. Manimozhi, "Content based image retrieval system using feed-forward backpropagation neural network," *International Journal of Computer Science and Engineering (IJCSE)*, vol. 2, no. 4, pp. 143 – 151, July 2013.
18. S. Zhou and T. S. Huang, "Relevance feedback in image retrieval: A comprehensive review," *Multimedia Systems*, vol 8, pp.536-544, 2003.
19. R. D. S. Torres et al., "Programming Framework for Content-based Image Retrieval," *Pattern Recognition*, vol. 42, no. 2, pp. 283–292, Feb. 2009.
20. G. Pass and R. Zabith, "Histogram refinement for content-based image retrieval," in *IEEE Workshop on Applications of Computer Vision*, 1996, pp. 96–102.
21. J. Huang et al., "Image indexing using color correlogram," in *IEEE Int. Conf. Computer Vision and Pattern Recognition*, June 1997, pp. 762–768.
22. A. Rao et al., "Spatial color histograms for content-based image retrieval," in *IEEE Int. Conf. Tools with Artificial Intelligence*, 1999, pp. 183–186.
23. L. Cinque et al., "Color-based image retrieval using spatial-chromatic histogram," *Image and Vision Computing*, pp. 979–986, 2001.
24. W.Y. Ma and B.S. Manjunath, "NeTra: a toolbox for navigating large image databases," *Multimedia Systems*, vol. 7, pp. 184–198, 1999.
25. Faloutsos et al., "Efficient and effective querying by image content," *Journal of Intelligent Information Systems*, vol. 3, pp. 231- 262, July 1994.

26. M. Flickner et al., "Query by image and video conten: the QBIC system", *IEEE 11th annual computer security applications conference*, 1995, pp.23 – 32
27. Pentland et al., "Photobook: Content-based manipulation of image databases," *International Journal of Computer Vision*, vol. 18, no. 3, pp. 233-254, 1996.
28. Carson et al., "Blobworld: Image segmentation using expectation maximization and its application to image querying," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1026-1038, Aug. 2002.
29. M. Squire et al., "Content-based query of image databases, inspirations from text retrieval: Inverted les, frequency-based weights and relevance feedback," in *Scandinavian Conf. Image Analysis*, Greenland, 1999, pp. 143-149.
30. J. R. Smith and S. Fu Chang, "Tools and techniques for color image retrieval, " *Storage and Retrieval for Image and Video Databases*, vol. 2670, 1996.
31. H.A, Jalab, "Image retrieval system based on color layout descriptor and Gabor filters", *IEEE Conference on Open Systems,* 25-28 Sept. 2011, pp. 32 – 36.
32. T. Wang et al., "Constraint based region matching for image retrieval," *International Journal of Computer Vision*, vol. 25, pp. 37–45, 2004.
33. Y. Mistry and D.T. Ingole, "Survey on content based image retrieval systems," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 1, no. 8, pp. 1827 – 1836, Oct. 2013.
34. D.S. Zhang and G. Lu, Enhancedgeneric," Fourier descriptorfor object-basedimage retrieval", *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 3668–3671, May 2002.
35. M. Yang, "A survey of shape feature extraction techniques", *Pattern recognition*, pp. 43 – 90, 2008.
36. D. Zhang and G. Lu, "Review of shape representation and description techniques", *Pattern Recognition*, vol. 37, pp. 1-19, 2004.
37. Haralick et al., "Textural features for image classification," *IEEE Transaction on Systems, ManCybernetics*, vol. 3, no. 6, pp. 610–621, 1973.
38. A. Hariram, "Image retrieval system based on feature computation – an integrated approach," in *Int. Conf. Information Systems And Computing,* India, vol. 3, no. 1, pp.464-467, 2013.

39. N. Arica and F. T. Yarman, "BAS: a perceptual shape descriptor based on the beam angle statistics," *Pattern Recognition*, vol. 24, pp. 1627–1639, 2003.

40. R. S. Torresa et al., "A graph-based approach for multiscale shape analysis," *Pattern Recognition,* vol. 37, pp. 1163–1174, 2004.

41. V. Struc and N. Pavesic, "A case study on appearance based feature extraction techniques and their susceptibility to image degradations for the task of face recognition," *World Academy of Science, Engineering and Technology, vol. 3, pp.811-819,* 2009.

42. T. Deselaers and T. Weyand, "FIRE in image CLEF2005: combining content-based image retrieval with textual information," *Accessing Multilingual Information Repositories*, vol. 4022, Springer Berlin Heidelberg, pp. 652-661, 2006.

43. R. D. S. Torres et al., "A genetic programming framework for content-based image retrieval," *Pattern Recognition*, vol. 42, no. 2, pp. 283–292, Feb. 2009.

44. J. W. Hsieh and W. E. L. Grimson, "Spatial template extraction for image retrieval by region matching," *IEEE Transaction on Image Processing*, vol. 12, no. 1, pp. 1404–1415, 2003.

45. J. F. D. Addison et al., "A comparison of feature extraction and selection techniques," 2002.

46. S. Zhou and T. S. Huang, "Relevance feedback in image retrieval: A comprehensive review," *Multimedia Systems*, vol. 8, no. 6, pp. 536-544, 2003.

47. Giacinto and F. Rolli, "Instance-based relevance feedback for image retrieval," *Neural Information Processing Systems Conf.*, Canada, Dec. 2004.

48. Setia et al., "SVM-based relevance feedback in image retrieval using invariant feature histograms," *IAPR Workshop on Machine Vision Applications,* Tsukuba Science City, Japan, May 2005.

49. R. S. Wua and W. H. Chung, "Ensemble one-class support vector machines for content-based image retrieval," *Expert Systems with Applications*, vol. 36, pp. 4451–4459, 2009.

50. Y. Liua et al., "A survey of content-based image retrieval with high-level semantics," *Pattern Recognition*, vol. 40, pp. 262–282, 2007.

51. J. Z. Wang et al., "SIMPLIcity: semantics-sensitive integrated matching for picture libraries," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 947–963, 2001.

52. F. Jing et al., "An efficient and effective region-based image retrieval framework", *IEEE Transaction on Image Processing*, vol. 13, no. 5, pp. 699–709, 2004.

53. C. Carson et al., "Blobworld: image segmentation using expectation-maximization and its application to image querying," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1026-1038, 2002.

54. Q. Iqbal and J. K. Aggarwal, "Retrieval by classification of images containing large manmade objects using perceptual grouping," *Pattern Recognition Journal*, vol. 35, no. 7, pp.1463–1479, 2002.

55. C. Dagli and T. S. Huang, "A framework for grid-based image retrieval," in *IEEE Proc. Int. Conf. Pattern recognition*, 2004.

56. Jun Yue et al," Content-based image retrieval using color and texture fused features", *Mathematical and Computer Modelling* ,Vol. 54, Issues 3–4, pp. 1121–1127, August 2011.

57. X.S. Zhou and T.S. Huang , "Edge-Based Structural Features for Content-Based Image Retrieval", *Pattern Recognition Letters - Special issue on image/video indexing and retrieval*, Volume 22 no. 5, pp. 457 – 468, April. 2001.

58. F.Malik and B.Baharudin, "Analysis of distance metrics in content-based image retrieval using statistical quantized histogram texture features in the DCT domain", *Journal of King Saud University Computer and Information Sciences* , vol. 25, pp. 207–218, 2013.

59. Dr. M. Sharma and Batra, "Analysis of Distance Measures in Content Based Image Retrieval", *Global Journal of Computer Science and Technology*, Vol. 14 no. 2 , 2014

60. A. K. Sinha and K.K. Shukla, "A Study of Distance Metrics in Histogram Based Image Retrieval", *International Journal of Computers & Technology*, Vol. 4 no. 3, March, 2013.

61. Siti Salwa Salleh et al. , "Combining Mahalanobis and Jaccard Distance to Overcome Similarity Measurement Constriction on Geometrical Shapes", *International Journal of Computer Science Issues*, Vol. 9, no. 4, pp 124 – 132, July 2012.

62. C. Theoharatos et al., "A generic scheme for color image retrieval based on the multivariate wald-wolfowitz test," *IEEE Transaction on Knowledge and Data Engineering*, pp. 808–819, 2005.

63. X. He, "Incremental semi-supervised subspace learning for image retrieval," in *ACM Proc. Multimedia*, 2004, pp. 10- 16.
64. N. Vasconcelos and A. Lippman, "A multiresolution manifold distance for invariant image similarity," *IEEE Transaction on Multimedia*, vol. 7, no. 1, pp.127–142, 2005.
65. X. He et al., "Learning an Image Manifold for Retrieval," in *ACM Proc. of Multimedia*, 2004.
66. S. Soman et al., "Content Based Image Retrieval Using Advanced Color and Texture Features", *International Journal of Computer Applications*, 2012.
67. Malini, R. and C. Vasanthanayaki, "Color perception histogram for image retrieval using multiple similarity measures," *Journal of Computer Science,* vol. 10, no. 6, pp. 985-994, 2014.
68. P. S. Suhasini et al., "Graph based segmentation in content based image retrieval," *Journal of Computer Science,* vol. 4, no. 8, pp. 699-705, 2008.
69. M. Kokera et al., "Comparison of similarity metrics for texture image retrieval," in *Conf. Convergent Technologies for the Asia Pacific*, Bangalore, India, 2003, pp. 571- 575.
70. D. Zhang and G. Lu, "Evaluation of similarity measurement for image retrieval," in *Int. Conf. on Neural Networks and Signal Processing*, 2003, pp. 928 – 931.
71. S. Farid and F. Ahmed, "Application of Niblack's method on images," in *Int. Conf. on Emerging Technologies*, 2009, pp. 280 – 286.
72. R. C. Gonzalez et al., *Digital Image Processing Using MATLAB.* Gatesmark Publishing, 2009.
73. L. Shapiro and G. Stockman, *Computer Vision.* Prentice Hall, 2001.
74. N. M. Wanas et al., "A study of local and global thresholding techniques in text categorization," in *Proc. of 5th Australasian Data Mining Conf.,* 2006, pp. 91 - 101.
75. G. Zhang et al., "Shape feature extraction using fourier descriptors with brightness in content-based medical image retrieval," in *IEEE Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing*, 2008, pp. 71 – 74.
76. H. Kauppinen et al., "An experimental comparison of autoregressive and Fourier based descriptors in 2D shape classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 201–207, 1995.

77. D. S. Zhang and G. Lu, "Evaluation of MPEG-7 shape descriptors against other shape descriptors," *Multimedia Systems*, vol. 9, no. 1, pp. 15–30, 2003.

78. J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 679–698, 1986.

79. S. Nandagopalan et al., "A universal model for content based image retrieval," *International Journal of Computer Science*, vol. 4, no. 4, pp. 242, 2009.

80. S. M. Khan et al., "Comparative study on content-based image retrieval (CBIR)," in *Int. Conf. on Advanced Computer Science Applications and Technologies (ACSAT)*, Nov 2012, pp. 61–66.

81. Y. Chen and J. Wang, "A region-based fuzzy feature matching approach to content-based image retrieval," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1252–1267, 2002.

82. J. Li et al., "SIMPLIcity: semantics sensitive integrated matching for picture libraries," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 947 – 963, 2001.

83. B. S. Manjunath et al., *Introduction to MPEG-7 multimedia content description interface*. New York: Wiley, 2002.

84. M. Swain and D. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

85. M. Pietikainen, "View-based recognition of real-world textures," *Pattern Recognition*, pp. 313–323, 2004.

86. T. Ojala, "A comparative study of texture measures with classification based on feature distribution," *Pattern Recognition*, vol.29, pp.51–59, 1996.

87. T. Ahonen, "Face recognition with local binary patterns," in *European Conf. on Computer Vision*, 2004, pp. 469–481.

88. T. Maenpaa, "Real-Time surface inspection by texture," *Real Time Imaging*, pp. 289–296, 2003.

89. V. Takala, "Block based methods for image retrieval using local binary patterns," *Scandinavian Conf. on Image Analysis*, 2005, pp. 882–89.

90. P.V. N. Reddy and K.S. Prasad, "Content based image retrieval using local derivative patterns," *Journal of Theoretical and Applied Information Technology*, vol. 28, no. 2, pp. 95 – 103, 2011.

91. T. Prathiba and G. SoniahDarathi, "An efficient content based image retrieval using local tetra pattern," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 2, no. 10, pp. 5040 – 5046, 2013.

92. T. Ojala, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis And Machine Intelligence*, vol. 24, no. 7, pp. 971 – 987, 2002.

93. P.V.N. Reddy et al., "Color image retrieval using mixed binary patterns," *International Journal of Engineering Sciences Research*, vol. 4, no. 1, pp. 1189 – 1194, 2013.

94. P. H. Bugatti et al., "Improving content-based retrieval of medical images through dynamic distance on relevance feedback," in *Int. Symp. on Computer-Based Medical Systems*, 2011, pp. 1–6.

95. J. Yuel et al., "Content-based image retrieval using color and texture fused features," *Mathematical and Computer Modelling,* vol. 54, no. 3: Pp. 1121 – 1127, 2011.

96. S. S. Park et al., "Expert system based on artificial neural networks for content-based image retrieval," *Expert Systems with Applications*, vol. 29, pp. 589–597, 2005.

97. R.S. Torres and A. X. Falcao, "Content-based image retrieval : theory and applications abstract," *Revista Iberoamericana de Tecnologias del Aprendizaje,* vol. 8, pp. 165– 189, 2006.

98. D. A. Chandy et al., "Texture feature extraction using gray level statistical matrix for content-based mammogram retrieval," *Multimedia Tools Application*, vol. 72, pp. 2011 - 2024, 2014.

99. T. Deselaers et al., "FIRE in ImageCLEF 2005 : combining content-based image retrieval with textual information retrieval," *Accessing Multilingual Information Repositories*, Springer Berlin Heidelberg, pp. 652-661, 2006.

100. M. Lam et al., "Content-based image retrieval for pulmonary computed tomography nodule images," in *Proc. of SPIE on Medical Imaging 2007: PACS and Imaging Informatics*, vol. 6516, 2007.

101. J. Eakins and M. Graham, "Content-based image retrieval," University of Northumbria at Newcastle, Technical Report, 1999.

102. W. Fan et al., "The effects of fitness functions on genetic programming-based ranking discovery for web search," *Journal of the Association for Information Science and Technology*, vol. 55, no. 7, pp. 628-636, 2004.

103. A. Shamsi et al., "A short-term learning approach based on similarity refinement in content-based image retrieval," *Multimedia Tools Application*, vol. 72, pp. 2025-2039, 2014.

104. Sumaira Muhammad Hayat Khan and Ayyaz Hussain, "A hybrid approach to content based image retrieval using computational intelligence techniques", *Indian Journal of Science Technology*, 10 Oct 2015