ENHANCED QUALITATIVE APPROACHES FOR CONTENT BASED IMAGE RETRIEVAL

Ph.D. THESIS

by

MEGHA AGARWAL

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247 667 (INDIA)
JULY, 2013

ENHANCED QUALITATIVE APPROACHES FOR CONTENT BASED IMAGE RETRIEVAL

A THESIS

Submitted in partial fulfilment of the requirements for the award of the degree
of
DOCTOR OF PHILOSOPHY
in
ELECTRICAL ENGINEERING

by

MEGHA AGARWAL

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247 667 (INDIA)
JULY, 2013

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE

CANDIDATE’S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **“ENHANCED QUALITATIVE APPROACHES FOR CONTENT BASED IMAGE RETRIEVAL”** in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy and submitted in the Department of Electrical Engineering, Indian Institute of Technology Roorkee, Roorkee is an authentic record of my own work carried out during a period from July, 2009 to July, 2013 under the supervision of Dr. R. P. Maheshwari, Professor, Department of Electrical Engineering, Indian Institute of Technology Roorkee, Roorkee.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other Institute.

(Megha Agarwal)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date: (R. P. Maheshwari)
Supervisor

The Ph.D. Viva-Voce Examination of **Ms. Megha Agarwal,** Research Scholar, has been held on ………………………

Signature of Supervisor(s) Chairman, SRC Signature of External Examiner

Head of the Department/Chairman, ODC

Abstract

Content based image retrieval (CBIR) paves a way to describe contents within the image based on features and retrieves images accordingly. It resolves the basic problems of text based image retrieval by automatic extraction of low level features from the visual contents of image like color, texture, shape and spatial layout etc. After feature extraction the next step is the measurement of similarity among various images based on these features. Performance of CBIR system substantially depends upon low level visual features. A visual feature can consider only single perception while multiple visual features can perceive an image through different perceptions. The aim of this research is to enhance the performance of retrieval system by designing effective and efficient algorithms for visual features as well as combination of such features. This research work proposes local, global, spatial and transformed domain features for enhanced image retrieval performance. Contributions made towards improvement in the performance of image retrieval systems are summarized as follows:

In one of the developed techniques, information of an image is extracted through local visual features. The local features are computed from small portions of an image. Hence they are capable of capturing minute variations present in the image. Local feature, viz. histogram of orientated gradient (HOG) is enumerated on image blocks. Thus it produces a huge set of feature vectors. This problem is taken care of by the vocabulary tree. Vocabulary tree reduces the complexities in feature indexing. HOG performs better than global feature, viz. Gabor wavelet transform (GWT) as well as local feature, scale invariant feature transform (SIFT).

The next step is towards feature extraction through multiresolution and multiorientation techniques. Transform domain methods allow extraction of image information at multiresolution and multiorientation levels. Log Gabor filter (LGF) is proposed for feature extraction through three scales and four orientations. It facilitates the better analysis of texture information present in the image. Mean and standard deviation are calculated from transformed image to obtain the texture statistics. LGF shows improvement in retrieval performance as compared to the existing GWT.

Another novel multiresolution approach is suggested by binary wavelet transform (BWT). BWT is computationally efficient technique. It decomposes an image into a pyramidal

structure of subimages with different resolutions corresponding to the different scales. It also provides directional information. BWT transformed images provide equal number of gray levels as the original image. This encourages the designing of binary wavelet transform based histogram (BWTH) feature for extracting information of an image. It extracts histogram for each BWT decomposed subbands. Color information is also integrated by computing BWTH on all the three color components of RGB color space. The results of BWTH based image retrieval system are compared with color histogram, auto correlogram (AC), discrete wavelet transform, directional binary wavelet patterns (DBWP) etc. and a significant improvement in the retrieval performance is observed. It is also noted that BWTH does not consider the spatial relationship among neighboring transformed coefficients. Hence, in order to overcome this problem BWTH is further enhanced by integrating it with correlogram features. This combination outperforms optimal quantized wavelet correlogram (OQWC), Gabor wavelet correlogram (GWC), AC and BWTH itself in terms of various performance measures.

Further, á trous wavelet transform (AWT) is utilized to extract multiresolution information. All orientation information is available in a single subband of á trous structure. In the first approach, correlation among á trous wavelet coefficients is employed to analyze the texture statistics. Thus, á trous wavelet correlogram (AWC) is proposed. As orientation information has a great importance in texture detection but it is lost in AWC. Thus in the second approach á trous gradient structure descriptor (AGSD) is designed by extracting the orientation information. In AGSD, orientation information is acquired from á trous wavelet transformed images and then microstructures are used to compute the similarity among orientations within the neighborhood. The microstructured image so obtained is used as a mask to get the corresponding á trous wavelet coefficients. Texture statistics is found out by calculating the correlation of these mapped á trous wavelet coefficients. Retrieval performance of AGSD is found superior to OQWC, combination of standard wavelet filter with rotated wavelet filter correlogram (SWF +RWF correlogram), GWC, combination of GWC with evolutionary group algorithm (EGA) and texton co-occurrence matrix (TCM).

Thereafter in the subsequent chapter, a new spatial method is propounded for texture feature extraction with the help of Haar-like wavelet filters. Haar-like wavelet filters possess simple structure which helps to gather image texture information about an image. From a set of Haar-like wavelet filters, poorer response filters are avoided and dominant filter is selected to propose a feature called cooccurrence of Haar-like wavelet filters

(CHLWF). It permits the consideration of only maximum filter edge response and less prominent directions of intensity variations are prohibited. Cooccurrence of dominant filter is used to capture intensity variations of the most prominent directions. Thus, statistics of dominant edges which incorporate the major information present in the image are utilized to compare various images. Results of this approach is compared with various related works in the literature like cross correlogram (CC), OQWC, GWC, SWF+RWF correlogram, dual tree complex wavelet transform (DT-CWT), dual tree rotated complex wavelet transform (DT-RCWT), DT-CWT + DT-RCWT, Gabor wavelet transform (GWT) etc. and effectiveness of CHLWF is established.

Integration of color as well as intensity properties perceives an image from different perspectives and captures additional image information. Using this concept weight cooccurrence based integrated color and intensity matrix (WCICIM) algorithm is proposed. WCICIM features are combined with integrated color and intensity cooccurrence matrix (ICICM) for final feature construction. In WCICIM suitable weights are assigned to each pixel according to its color and intensity contributions. It finds correlation among color–color, color–intensity, intensity–color and intensity–intensity, based on neighboring pixel variations in the weight matrixes. These features have improved the performance as compared with motif cooccurrence matrix (MCM), ICICM, color correlogram and combination of block bit plane (BBP) with global color histogram (GCH) features.

The performance of the proposed methods is tested on five distinct benchmark image databases (Corel 1000, Corel 2450, MIRFLICKR 25000, Brodatz and MIT VisTex). The results show progressively improved retrieval performance of the proposed algorithms in terms of various performance measures.

Acknowledgements

The successful completion of this manuscript was made possible through the invaluable contribution of a number of people. I would like to thank all people who have helped and inspired me during my doctoral study. To say “thank you” to all of you is not even enough to express my gratitude.

I wish to express my warmest gratitude to my supervisor, Dr. R. P. Maheshwari, Professor, Department of Electrical Engineering, IIT Roorkee, for his proficient and enthusiastic guidance, useful criticism, continuous monitoring and time management throughout the research work, which immensely help me to carry out this research work smoothly with a deep insight in the world of research. I sincerely appreciate him for all the help he rendered to me and his pronounced individualities as well as believing in me and pushing me to my limits to complete this thesis. I humbly acknowledge a lifetime’s gratitude to my mentor for the patience and support he gave me since day one of my research and study at IIT Roorkee. I am indeed fortunate to have him as mentor and supervisor, without his interest and motivation; it is unlikely that this thesis could have taken shape.

I wish to express my sincere gratitude towards research committee members; Prof. S. P. Srivastava, Chairmen of Departmental Research Committee (DRC), Electrical Engineering Department who made constructive recommendations and spared their valuable time in reviewing and critically examining the work.

My heartily gratitude to Dr. R. Balasubramanian, Associate Professor, Department of Mathematics (External Expert) for his humanistic, encouraging and warm personal approach and for the necessary suggestions provided to me to carry out my research work. I acknowledge Dr. R. S. Anand, Professor, Department of Electrical Engineering (Internal Expert) for his kindness in providing me valuable suggestions to complete this research work.

I also wish to extend my sincere thanks to Head of Department of Electrical Engineering, Director, IIT Roorkee and other faculty members of the department for their invaluable encouragement, support and the noblest treatment extended by them for providing the laboratory and computing facility to carry out this work.

Though no amount of money can take out pains during the course of research but it does provide a soothing effect. I acknowledge Ministry of Human Resource and Development, Government of India, for providing me with the necessary financial assistance.

My special thanks goes to my all lab mates Mr. A. B. Gonde, Mr. D. Rajoriya, Mr. P. Arvind, Mr. S. Murala, Mrs. K. Saini, Mr. B. Vyas and all other fellow researchers for their friendly company and direct or indirect help provided by them through research span.

I wish to thank my office superintendent Mr. Mohan Singh, Xerox in charge Mr. Sushil and other office staff who always helpful during my research work. I also wish to thank Instrumentation & Signal Processing lab supporting staff Mr. Joginder Prasad, Mr. Rajiv Gupta and Mr. Veer Chand Ji for all the necessary help provided by them.

I extend special thanks to my inmates in Sarojini hostel and others hostels specially Ms. Monika for her nicest company provided during my stay in campus.

I am thankful to my friend Mr. Shailendra Sagar and his family for constant support and encouragement in all ways and means by them in my research tenure. I feel proud to him for being one of my best counsellors.

My deepest gratitude goes to my family for their unflagging love and support throughout my life; this thesis is simply impossible without them. I am indebted to my father, Mr. J. K. Agarwal, for his care and love, he worked industriously to support the family and spare no effort to provide the best possible environment for me to grow up. I cannot ask for more from my mother, Dr. Sadhana Agarwal, as she is simply perfect. I have no suitable word that can fully describe her everlasting love to me. I remember, most of all, her delicious dishes. Mother, I love you. I sincerely acknowledge the moral support and encouragement from my siblings Moni didi, Gudden and Gopal for their sign of love and affection towards me.

I am proud to humbly dedicate this research work to my family.

Above all, at the first place, I wish to bow before the “**ALMIGHTY”** whose divine light and warmth showered upon me the perseverance, inspiration faith and enough strength to keep the momentum of work high even at tough moments of research.

(Megha Agarwal)

Contents

List of Figures

List of Tables

Introduction

Chapter 1

Digital media is gaining importance day by day due to advances in the ever expanding internet and new digital image sensor technologies. Large image chunks have been created by scientific, educational, medical, industrial, and other entrepreneurial sections. The ease of affordable digital cam-n-camera and expanding internet facilities explore a novel way to upload and share personal photo and video collection on web. Everyday this immense amount of digital media on web creates huge image database and increases complexities in retrieving relevant images according to user's interest. The drastic expansion of digital libraries in the present scenario makes handling of this database by the human extremely cumbersome rather impractical task. Therefore, a dire need of an efficient and automatic procedure for indexing and retrieving images from these databases came into notice. Image retrieval (IR) is widely spread in various fields like medicine, journalism, entertainment, environmental etc. The domain of IR is enormous for example a medical student studying anatomy may be interested in sample instances of a given disease. There are two main frameworks of IR system first one is Text Based Image Retrieval (TBIR) and the other one is Content Based Image Retrieval (CBIR) [1, 2]. In the form of TBIR, IR is commercialized by web image search engine like Google and YAHOO!. Keywords are entered as query and images conforming semantics to keywords are retrieved [2]. TBIR possess several drawbacks. Every human has different interpretation for same image like any image having flower and tree can be annotated by flower, tree, nature or outdoor scene. As well as it takes lot of time to manually annotate image and maintain consistency. While annotating huge database probability of error is also more. So to avoid all such problem it is required to represent images automatically from its visual contents.

The feature extraction is a prominent step and the capability of a CBIR system is dependent upon the method of feature extraction from raw images. The CBIR utilizes visual content of an image which is further classified into general features (color, texture, shape) and domain specific features (human faces and finger prints). There is no single best representation of an image for all perceptual subjectivity, because the user may take the photographs in different conditions (view angle, illumination changes, etc). In the following section CBIR structure is overviewed.

1.1 CONTENT BASED IMAGE RETRIEVAL

In content based image retrieval (CBIR) first of all query image is specified according to user's interest and images are represented by visual contents. The main low level visual contents are color, texture, shape and spatial layout [3, 4]. Visual contents can have various representations to characterize image's visual features from different visual perspectives. Contents of the images in the database are extracted in the form of features and described as feature vectors. These feature vectors of the database images form a feature database. After feature database creation the next step is similarity comparison. It finds similarities between the feature vector of the query image and database images, followed up with the help of an indexing scheme it retrieves images relevant to the input query image. Efficiency and accuracy of retrieval system depends upon how well features are able to match with human visual perception and how effectively similarity function works. Various comprehensive studies of CBIR are incorporated in [1–8]. Fig. 1.1 illustrates basic CBIR structure. Processing of each block of CBIR system shown in Fig. 1.1 is explained in the following section.

1.1.1 Query specification

In the first step of CBIR system a query image is formulated. It specifies what kind of images user wants to retrieve from the database. There are four different ways to specify query for IR [1, 6]. First one is *query by example*. In this method input query image is a sample image and user is interested to find images similar to that input image. Query image can be selected from database itself or it can be any external image to the database. The system first converts the query image into low level visual features and images having similar features are required to be retrieved from image database. Second one is *query by sketch*. In this method user is asked to draw a sketch of the query image with certain properties like color, texture, shape, sizes and locations. Sketch can be drawn with a graphic editing tool provided either by the retrieval system or by some other software. This sketch is input as the query to the retrieval system. Third one is *query by concept*. In this case, input query is the conceptual attributes of the image like signifying events or emotions. For example search for ‘joyful people’ images. This consists of high level semantics. The last one is *query by category*. It specifies the category of image to be retrieved. For example search for ‘car’ images. This method also has high level semantics. In CBIR, *query by example* method is used for query formation.

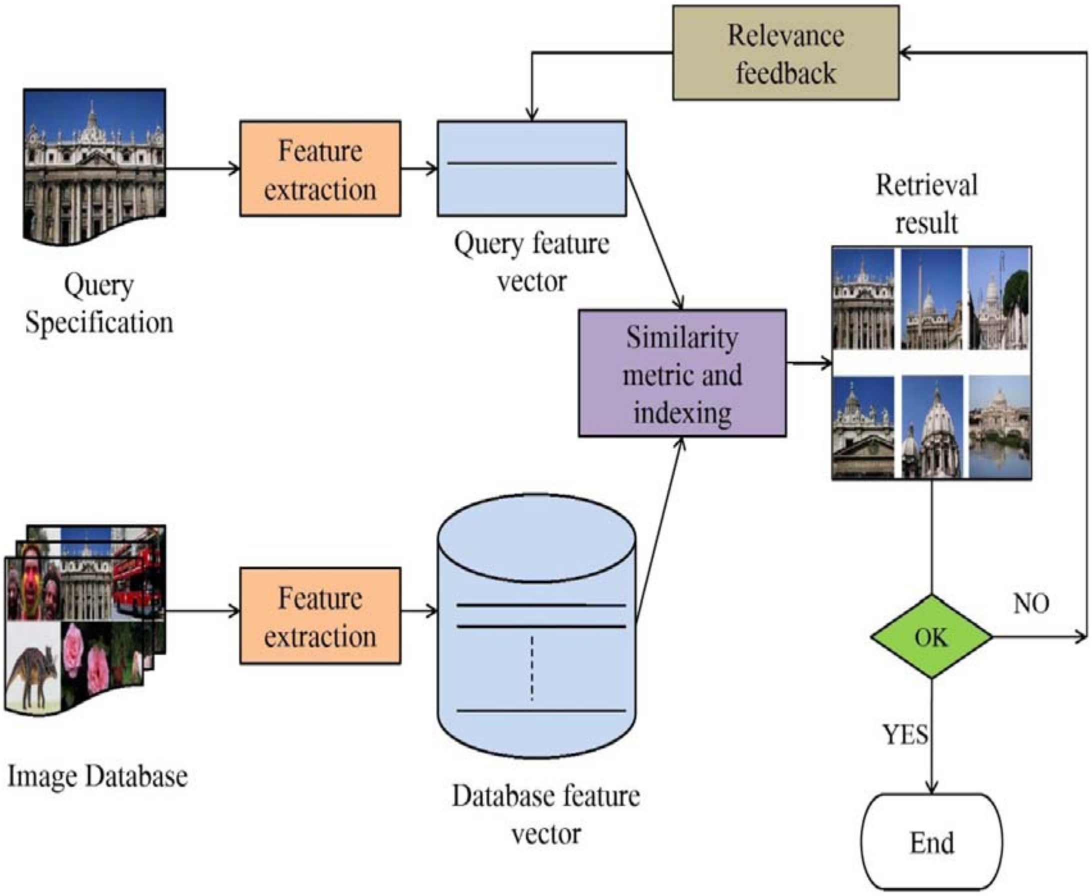


Fig. 1.1: Basic CBIR structure

1.1.2 Low level visual feature

After query formation features are extracted from images. Each image in the database is represented by numeric value called features or descriptors. These features are meant to represent properties of images to allow meaningful retrieval for the user. In CBIR system features are extracted with respect to low level visual contents like, color, texture, shape and spatial layout [5, 6, 7]. Each of these features extracts different information from the image and accordingly relevant images are retrieved.

1.1.2.1 Types of visual features

Depending upon the portion of image considered for feature extraction low level visual features can be classified in two categories; global and local [9]. Both global and local features have their own advantages and disadvantages. Hence, the choice of image feature to be used depends upon the desirability of the system. Global features are computed to capture overall characteristics of an image by only one set of features. It extracts global statistics of the image without involving segmentation or grouping operation. Local descriptor uses the visual features of regions or objects to describe the image content. A set of local features are computed in local manner. A simplest local feature first divides an image into small non-overlapping blocks and then features are computed individually for each block e.g. average pixel value across a small block. There can be various other ways to calculate local features. Like salient feature extraction is a way to get compact image representations. It goes through extraction of salient points and features are computed only for these salient points [3]. This leads to an efficient indexing and good discriminative power, especially in object based retrieval. Sometime user is much interested in particular region rather entire image. In that case image is first segmented into regions. These regions may represent any object too. Then, for each region image features are extracted. Region based approach needs proper segmentation before the feature extraction. Thus it is computationally complex method [1, 6].

1.1.2.2 Visual feature extraction

Various low level visual contents are present in an image like color, texture, shape and spatial layout. Basic idea to extract feature descriptors through these visual contents is briefly overviewed in the following sections.

a. Color

Color is the most significant visual property perceived by human and thus it is extensively being used in CBIR [6, 7]. Color domain allows use of three or four dimensional vector color space. Multidimensional color space provides superior discrimination capability as compared to single dimensional gray space. Also, it is independent to image size and orientation. Various kinds of color spaces are available in literature [10, 11, 12]. Choice of color space also plays an important role in effective feature extraction. Performance of color space is application dependent. However, desirable characteristics of an appropriate color space for image retrieval are its completeness, uniformity and compactness [10, 6].

Most widely used color spaces are RGB (red, green, blue), CMYK (cyan, magenta, yellow, key), HSV (hue, saturation, intensity), CIE L*a*b* and CIE L*u*v* [11]. 'L' corresponds to luminance while a, b and u, v corresponds to two chromatic components. Performance of various color spaces are compared in [10]. Transformation among color spaces is also possible and is given in [11]. Selection of color space for a specific application is difficult. No color space is universally fit for all applications. Sometimes several color spaces perform equally so, weighted subset of color spaces may produce best result for a particular task. The challenge is to obtain a proper fusion algorithm so as to combine their results in an optimal manner. In [13] selection and fusion of color spaces and feature detection algorithm is presented to optimize discriminative power with balance in repeatability.

b. Texture

Texture measures the characteristics of images with respect to changes in certain directions and scales [1, 7, 14]. It extracts important information about structural arrangement of surfaces (like clouds, fabrics, bricks etc.) and their relationship to the surrounding environment. Mostly texture information considers behavior of group of pixels rather than nature of single pixel. Natural images are the best examples of color and texture mosaic. Although most texture descriptors work on gray space, but to optimize the performance of features sometimes they can be applied on color images. Various strategies are followed to extract texture features [8].

Texture features can be characterized in terms of low level statistics of grayscale images. It represents the spatial distribution of pixels. Depending upon the number of pixels concerned for computation of texture features, these statistical methods can be categorized into first order, second order and higher order. First order statistics considers single pixel, second order statistics considers behavior of pixel pairs and higher order statistics considers three or more than that. Any texture feature extracted by mean is supposed to contain first order statistical information of that particular texture. Gray level cooccurrence matrix is computed by analyzing the probability of cooccurrence of pixel-pair values in different distances and orientations [15] hence, it contains second order statistical information. Autocorrelation is also used for analyzing regularity and coarseness of texture. Various texture features like energy, entropy, correlation etc. are extracted from the cooccurrence matrix.

Spatial relationship of pixel pairs is also very popular method for texture feature extraction [15]. Another method to characterize an image texture is by a set of texture primitives (texton) [16]. After primitive identification, their placement rules are identified e.g. statistics of relative vectors that join the primitive's centroid. Such analysis is difficult for natural textures, because primitives and placement rules can be irregular for such kind of images. Image texture can also be modeled in terms of spot, line, edge, corner etc. Statistics of these models capture the properties of textures e.g. local binary pattern (LBP) [17]. Texture features can extract information present at multiple resolution and multiple scales of an image [8]. For this purpose filters are applied on the image. Both spatial domain or frequency domain filters can be used. They differ in the way of getting the filter responses. In spatial domain both filter and input image are convolved to obtain the filter response. In case of frequency domain firstly image and filter are transformed into frequency domain and then multiplied together to obtain the filter response.

c. Shape features

Shape features are not very popular in general CBIR system as compared to color and texture features because most of the shape features require prior proper segmentation. These features are often more sensitive to noise, defects, arbitrary distortion and occlusion. Shape features are generally extracted from shape boundary or interior contents [18]. Contour based method computes shape features only from boundary or contour of the shape. Because this method uses only boundary for feature extraction, thus, it is more sensitive to noise as even small changes in the shape significantly affect the shape contour. Region based methods exploit entire region inside the shape rather than only boundary information. This method is less sensitive to noise because total region is taken into consideration for feature extraction.

d. Spatial layout

Spatial layout considers spatial location of an object (or region) or spatial relationships among the objects (or regions) in an image [6]. Computation of spatial relationship of region and among regions is not easily accessible due to necessity of proper segmentation of regions. Depending upon the regions for spatial relationship computation spatial layout can be of two types [1]. In absolute spatial location method, spatial locations of regions are computed with respect to image itself like top, down, centre etc. For instance, sky, birds are usually at the top of an image and people, building are usually at the centre and sea, grasses are usually at the bottom. Relative spatial location method considers spatial relationships between regions present in an image. It computes the relative positions of objects for differentiation. Like, region A (or object A) is situated left to the region B (or object B) or region A (or object A) is situated below to the region B (or object B).

e. Integrated features

Each low level visual feature explains an image in different perspective. Sometimes two visual features separately do not give enough information for image representation but combination of such features may work great. This introduces the concept of integrated features. Each feature is assigned with some weights according to its importance. Weights can be chosen by trial and error method or can be optimized.

1.1.3 Similarity measure and indexing technique

For image retrieval it is expected to have fast and effective browsing. So, for the features giving high dimensionality it is required to reduce the size of features upto a certain range for an efficient indexing. To solve this purpose offline, high dimensional reduction method is applied [1]. It selects only more effective features for indexing purpose thus feature dimension is decreased [6]. Final dimension of feature depends upon the desirability of system. Dimension reduction is required in case of high dimensional feature vector, so it is an optional block in basic CBIR mechanism. After getting proper visual features the next step is the matching of images according to their visual features [19]. Selection of proper similarity measure also plays a great role in retrieval result. Similarity measure should be selected efficiently to provide better image retrieval result. The successive step to similarity comparison is the indexing of multidimensional features. Indexing is integrated into the CBIR systems so as to obtain retrieval result [7]. CBIR system outputs a list of images relevant to the query image. Number of retrieved images depends upon user’s interest. Retrieval performance of CBIR system is measured in terms of number of relevant images retrieved. Therefore, in an ideal manner it is required to have all the retrieved images to be relevant.

1.1.4 Relevance feedback

Interpretation of images by human is called high level semantic. Human perception is subjective and application dependent. Machine based CBIR system considers low level visual descriptor for image understanding. Although CBIR provides promising direction for image retrieval but sometime it is unable to satisfy human perception and subjectivity. So there is semantic gap between low level visual features and high level semantic. This gap is needed to be reduced. Relevance feedback (RF) is the tool to bridge this gap by incorporating user in the loop of image retrieval [20, 21]. It is an online process which assists user to interact with the retrieval system and retrieval result can be altered according to user's interest by updating the query features [22]. Thus by modifying the retrieval result, finally a perceptually and semantically more meaningful retrieval results are obtained.

1.1.5 Practical applications of image retrieval system

There is a wide range of CBIR application in real world [3, 5, 23]. Few such areas with brief details are listed below.

- *Crime investigation*

  Crime investigation agencies generally archive a wide range of visual evidences. Whenever a new case is lodged then the visual evidences of that crime may be required to compare with entire archival record to find out similar crime and data regarding. In addition to it, it is also required to filter illicit images by comparing with unusual images. There CBIR can play its vital role.
- *Intellectual property*

  In trademark image registration before the registration of any new trademark, it is required to compare it with existing trademarks to ensure that there is no risk of violation. CBIR systems are recognized as a prime resource in this field.
- *Designing*

  In designing applications like architectural, fashion or interior etc., designer is supplied with few constraints to be incorporated in the design. So, designer should be aware of previous designs to create the optimized design by satisfying the constraints. CBIR helps to browse available designs by following the constraints.

- *Medical diagnosis*

  Modern medicine diagnostic techniques such as radiology, computerized tomography have resulted in an explosion in the number of medical images. There is an increasing interest in the use of CBIR techniques to aid diagnosis by identifying the similar past instances of same disease.

- *Cultural heritage*

  Art galleries and museums deal with a wide range of objects. The demand is to develop a tool so as to identify objects sharing some aspects of visual similarity. Visitor may be an art lover interested in particular types of painting or a historiographer following art of any era. CBIR allows ease in the access of huge paintings or sculptures by a visitor or researcher through their image database.

- *Education and training*

  Now a day, classrooms are facilitated with projectors. To explain any complex mechanism or design, mentor may be asked for different representation for easy understanding. Like nebula in astronomical phenomena and dribbling in sport activities. In education and training centers CBIR can be introduced for quality learning and fast explanation.

- *Home recreation*

  Maintaining a digital camera is not a big deal in today's scenario. Home picture collection includes holiday snapshots, glimpses of favorite movie, portfolio of favorite actor etc. In leisure time, one may be interested to recall memories of their holiday spent on a beach or successive scenes of any movie. CBIR assists to retrieve such images from image collection.

- *Web searching*

  Millions of images are available as well as are published continuously on web. It mentions the demand of an efficient image searching tool. Google has commercialized similar image search engine to render relevant images according to one's appeal. This tool offers image retrieval based on query image to fulfill the semantics of user through visual contents of the image.

1.1.6 State of art in image retrieval systems

Various CBIR systems are available with different frameworks [24, 9, 7]. Each system possesses different combination of feature descriptor, similarity measures and indexing structures. Like, *Query by Image Content (QBIC)* system from IBM includes three types of features viz. color histogram, shape feature based on moment and texture feature based on contrast, coarseness and directionality [25]. *BlobWorld* system is developed at the University of California with segmented features [26]. *Search Images by Appearance (SIMBA)* system uses color and texture based rotation and translation invariant features [9]. Many other systems [24] like *Colour Hierarchical Representation Oriented Management Architecture (CHROMA), Computer Aided Search System (Compass), Finding Regions in the Pictures (FRIP), Key-points Indexing Web Interface (KIWI), PicHunter, PicSOM, PicToSeek, Recherche et Traque Interactive (RETIN), Shoebox, RetrievalLab* [27] *Multi-Feature Indexing Network (MUFIN)* [28] are also designed for image retrieval. Recently, Google offered a way to search similar images according to contents. Google image search engine provides a tool to find similar images according to input query image and further refine search results by updating the query image [29].

1.2 LITERATURE SURVEY

CBIR has been attracting researcher's interest from last few decades. In CBIR images are represented by low level visual features like color, texture and shape. Various feature descriptors are available in literature. Performance of these features varies with respect to effectiveness to represent images [30]. Any system cannot consider all the available features to extract information from the image due to an increase in computation time as well as feature dimensions. Similarity comparison of images is based upon their feature vectors. Thus, selection of features results in variation of CBIR system performance. Literature reviewed during this research work has been briefly discussed in the following sections.

At this point it will be pertinent to mention that for image retrieval color and texture features are normally used. While other features like shape and local features are generally used for the purpose of image analysis like object detection, image classification etc. Thus the following sections focus on color, texture and integration of color with texture as well as some other features for image retrieval. Besides them some other features are also discussed to make the presentation complete.

1.2.1 Image retrieval methods using color features

Color of an image is one of the most important content to extract spatial visual features for image retrieval. RGB color histogram [31] is a very popular method of image feature extraction in the field of CBIR. It is obtained by counting the number of times each color occurs in an image but it does not consider spatial relationship among colors. Color histogram is easy to compute and insensitive to small changes in viewpoint. In addition, it is sensitive to both compression artifacts and changes in overall image brightness. Properties of HSV color space match with the human visual perception. So, in [32] histogram is generated from HSV color space by analyzing the relative importance of hue and intensity based on the saturation of a pixel. This histogram is a combination of “true color” and “gray color” components and weights of these components are used to update respective histogram values. In fixed color quantization arithmetic almost same colors are assigned to two different bins, to avoid this problem in [33] an efficient color space quantization arithmetic and histogram similarity function is proposed for HSV histograms computation. Local chromatic distribution and local directional distribution of intensity gradient are combined for color image retrieval [34]. It considers color distribution as well as edge information of the image. For chromatic information extraction, RGB color image is divided into 4X4 non-overlapping blocks and each block is categorized into uniform or non-uniform based on its intensity gradient. For uniform blocks uni-color and for non-uniform blocks bi-color histograms are computed. Directional information is comprised by computing inter-block intensity gradient histogram.

Retrieval process is speeded up as well as quantization problem of histogram is avoided by characterizing the color distribution by a set of color moments [35]. Further performance of color moments are improved by providing higher importance to more informative regions [36]. To incorporate spatial relationship of color pairs, color coherence vector [37] is proposed by splitting each histogram bin into two parts, i.e., coherent, if it belongs to a large uniformly colored region or incoherent, if it does not. Color cooccurrence matrix or color correlogram [38] considers both local as well as global properties of the image. Local properties are extracted by neighborhood gradient directions while global properties are obtained by utilizing local distribution of image. It offers robustness to the appearance changes against background variations, viewpoint changes and zooming. In [39] maximum RGB correlation index of query and database image is used for image retrieval. Images having maximum correlation are retrieved as they are more similar to each other. Correlogram is also calculated in HSV space [40] by quantizing the hue component more precisely than value component so that it becomes more sensitive to changes in color content and less sensitive to illumination and performs better than RGB correlogram [38]. As the color cooccurrence matrix considers only diagonal elements to extract information from homogeneous regions, furthermore color cooccurrence matrix is modified by incorporating non-diagonal elements also [41]. This allows more utilization of edge information. Performance of color correlogram is further improved by avoiding non-edge pixels. Color edge cooccurrence histogram (CECH) [42, 43] captures the distribution of separation between pairs of color edges and solid color regions are eliminated. Phan *et al.* [44] performed CBIR on color image database by using color edge gradient cooccurrence histogram (CEGCH). Vector order statistics are more efficient as compared to simple color pixel difference for color edge detection. By using vector order color gradient and HSV space quantization, CEGCH feature is generated for logos and trademark retrieval. In another method, color and spatial information are combined by clustering the colors according to their values and through these clusters spatial distribution among similar color is extracted [45].

A perceptual color feature is proposed in [46] with all major properties of prominent colors in spatial and color domains. Since, HSV space suffers from discontinuities and RGB color space is not perceptually uniform hence, CIE-Luv color space is used in [46] for image retrieval. Dominant colors are selected in color domain and spatial color distribution is computed by performing quad-tree decomposition on the image. To increase the effectiveness of retrieval system, color distribution (mean, standard deviation) and image bitmap are combined [47]. Color distribution considers global information while image bitmap exploits local characteristics. In [48] through principal components analysis (PCA) a compact color descriptor is proposed for region based localization and it is further used for image retrieval. Other than CBIR color feature are also being used in classification, object detection, pedestrian detection etc. These features can be used directly or with some modifications for CBIR purpose. Like, in [49] color features are combined with motion feature to classify sports events and are suggested for retrieval also. Similarly, color edge cooccurrence histogram (CECH) [42, 43] is proposed for object detection and it's extended version color edge gradient cooccurrence histogram

(CEGCH) is proposed for image retrieval [44]. In the present research work, Chapter 3 and Chapter 6 propose features by incorporating color information into account.

1.2.2 Image retrieval methods using texture features

Texture is second most important property of image representation. It measures intensity variations within the local neighborhood in terms of certain repetitive patterns and finds out the number, types of repetitive patterns and their spatial arrangement on a surface texture image [50]. Texture analysis has been an eye catcher due to its potential values for computer vision and pattern recognition applications. Texture retrieval is a branch of texture analysis that has attracted wide attention from industries since this is well suited for identification of products such as ceramic tiles, marble, parquet slabs, etc. Image texture is represented in terms of roughness, contrast, directionality, regularity, coarseness and line likelihood to satisfy human visual perception [51]. These features are well known as Tamura features. In [52] statistical texture features are computed locally for CBIR by partitioning the gray level image into blocks and probability distribution of all blocks is used to compute texture features i.e. mean, standard deviation, skewness, kurtosis, energy, entropy and smoothness. In [53] before feature extraction, preprocessing is performed by median and Laplacian filters to remove noise and enhance edge information. Further histogram is calculated to determine regions in the image and mean, standard deviation are computed as the final texture feature for CBIR. Other than CBIR, the statistical properties of texture are also being used in image fusion, texture classification etc. Like in [54] statistical properties of neighborhood are computed to select sharper pixel in the neighborhood for image fusion.

Texture is also considered to be a composition of “texture elements” or primitives. Texture features are extracted by analyzing statistical properties of these primitives and rules governing their spatial distribution [50]. After primitive detection, size, shape, area, length, orientation of these primitives and their spatial arrangement is utilized for texture feature computation. In [55] image is considered as a composition of six motif patterns. On the motif transformed image statistical feature, motif cooccurrence matrix (MCM) is computed to provide spatial relationship among motifs. It gives a rich description of low level semantics. MCM is invariant to any change in the image contrast or brightness due to camera because it works on difference in intensities not on pixel values directly. Julesz [16] proposed the concept of “texton”. Textons are defined as a patterns sharing a common property all over the image. As image edge has a close relationship with contour and texture pattern, in [56] texton cooccurrence matrix (TCM) is computed by describing the gradient and quantized color image in terms of textons. Cooccurrence matrix of these texton images is extracted in four directions to extract features. From TCM, energy, contrast, entropy and homogeneity features are computed to describe image features. Liu *et al*. [57] further proposed the extended version of TCM by multi-texton histogram (MTH). It represents the spatial correlation of color and texture orientation. In MTH advantages of cooccurrence matrix and histogram both are combined by representing the cooccurrence matrix using histogram. It is observed that the contents of natural images are composed of many universal microstructures. These microstructures can serve as common base for the comparison and analysis of different images. In microstructure descriptor (MSD) [58], HSV color space is quantized to detect edge orientation and microstructures are defined through edge orientation image. Orientation image is used to define microstructure map, because orientation is invariant to color, illumination, translation and scaling variation. Cooccurrence microstructure features are computed by underlying color information extracted through microstructure map. Statistics of texture pattern are also utilized for image segmentation. In [59] statistical descriptors, adaptive to the variation of texture patterns are proposed for texture segmentation. In this approach a scan-line based method is applied to extract repetitive patterns in each color channel, then mean and standard deviation of each pattern is collected to characterize their local properties.

Local binary pattern (LBP) can also be considered as a type of texture elements [60]. It is not very much popular in the field of CBIR but it is widely used in texture classification, segmentation and face recognition. Like, in [61] multiresolution rotation invariant LBP based texture feature is proposed for texture classification. Most of the time, input image is preprocessed prior to LBP feature extraction. In [62] image is preprocessed by Gabor filter to extract information from broader scales for face detection. Further pyramid domain LBP (PLBP) is computed by cascading the LBP information at hierarchical spatial pyramids to incorporate multiresolution information [63]. LBP feature looses global spatial information so; Guo *et al*. [64] proposed a hybrid scheme with locally variant LBP texture features with globally rotational invariant matching by introducing LBP variance (LBPV) feature. By applying the LBP concept, research is started in the CBIR field also. Murala *et al*. [65] proposed directional binary wavelet patterns (DBWP) for image indexing and retrieval. In this an 8-bit grayscale image is divided into eight binary bit-planes and binary wavelet transform (BWT) is computed on each bit-plane. From the resultant BWT sub-bands, local binary pattern (LBP) features are extracted.

Edges also play vital role in texture analysis. Statistics of these edges are used for texture feature extraction. In [66] texture edge descriptor (TED) is proposed with concern about edges of pedestrian in videos. TED texture feature incorporate both texture and edge information and is independent of neighborhood size. Various kinds of other texture features are also computed by using edges like CECH [42], CEGCH [44] etc. and are used in CBIR. In [67] connected components of edge graph constructed by depth map image of the range data is used for ear detection in 3D. Invariance to linear/non-linear monotonic gray scale transformations can be achieved by analyzing an image through relative rank of pixels rather than their gray scale value. Ranklet transform computes rank of pixels at different resolutions and orientations, and texture features are extracted from ranklet images [68].

Relationship between average intensities within a neighborhood possess good amount of textural information. Oren *et al.* [69] proposed image representation similar to Haar basis functions which represents an image by a set of regions and captures relationship between average intensities of neighboring regions. Considering average intensities of regions, leads to decreased sensitivity to noise and illumination changes. This structure is called Haar wavelet and captures intensity variation in horizontal, vertical and diagonal directions. Viola *et al.* [70] introduced integral image representation and selected small number of visual features for fast computation of Haar-like features for object detection. Viola *et al.* [71] implemented a classifier to avoid background region and put more emphasis on face region for efficient detection. Lienhart *et al.* [72] enhanced the object detection by introducing an efficient set of 45° rotated Haar-like features. Further, in [73] Haar-like features are extended by including some more features with rectangles of same size as well as different sizes and tried to capture more orientation information. Li *et al.* [74] proposed Haar-like features with disjoint rectangles to characterize non-symmetrical objects. Performance of Haar-like features is further improved by assigning optimal weights to their rectangles so as to maximize the ability to discriminate objects [75]. By utilizing the concepts provided in the development of Haar features, in Chapter 5 a novel feature based on Haar-like wavelet filters is proposed for CBIR.

Texture of an image can be analyzed by frequency contents of the image. First the image is filtered by bank of filters with each filter having a specific frequency and orientation and later features are extracted from filtered images. Filtering can be performed in transform as well as spatial domain to extract information at multiresolution levels. The multiresolution analysis consists of two structures, viz. pyramid and á trous. In pyramid structure, images at each scale are down sampled by a factor of two. It causes reduction in subband sizes at each scale. Wavelet transform allows getting details of image at multiple resolutions by filtering and down sampling of image. Mallat [76] has given the concept of separable two dimensional discrete wavelet transform (DWT). In this, filtering is performed at each level first along one dimension and then in the other dimension. DWT mixes diagonal and negative diagonal information in a single subband. Smith *et al.* [77] calculated mean and variance of the three scale wavelet coefficients for CBIR. In [78] images are decomposed by using DWT up to three levels and texture features are extracted by energy of each subbands. These features are forwarded to soft-set theory based algorithm for classification.

In the case of á trous structure, down sampling is not performed. This causes number of approximation coefficients always equal to number of pixels in the image. So, analysis of subbands among different scales is possible. By avoiding the down sampling, translation invariant property is also achieved. Á trous wavelet transform is a type of multiresolution analysis with shift-invariant property. Unlike pyramidal structure wavelet transform, á trous structure wavelet transforms does not provide classification among horizontal, vertical and diagonal directional information [79, 80]. Á trous structure wavelet with the combination of MSD is proposed in Chapter 4 for feature extraction.

Rotated wavelet filter (RWF) set is proposed with the combination of DWT in [81] to distinguish between diagonal and negative diagonal information. Mean and standard deviation are computed from each subband to extract texture information. Concept of color (color correlogram) and texture (wavelet transform) is combined in wavelet correlogram (WC) [82]. Saadatmand *et al.* [83] has stepped up the performance of WC through optimized quantization of wavelet thresholds correlogram (OQWC) and obtained better performance than WC. They used evolutionary group algorithm (EGA) to optimize the wavelet quantization thresholds and correlogram is computed only for horizontal and vertical subband. The correlation between subbands at the same resolution exhibits a strong relationship, this property is utilized to characterize textures, and cooccurrence histogram of DWT decomposed image is proposed in [84]. It captures relationships between each high frequency subband and low frequency subband at the corresponding scale to characterize the texture property. The discrete cosine transform (DCT) is often used for data compression. In DCT most of the information is present in low frequency components. In [85] DCT is applied at multiresolution and multilevel discrete cosine transforms (MDCT) together with Zernike moments is proposed for image feature representation. Texture features are extracted from low resolution image of MDCT to reduce the computational time.

Binary wavelet transform (BWT) yields an output similar to the thresholded output of a real wavelet transform operating on the binary image and it is proposed for character recognition and compression [86]. Further, bit-plane decomposition is proposed to compress grayscale images by using BWT [87, 88]. In-place implementation of BWT is similar to lifting scheme in real wavelet transform with reduced transformation complexity [88, 89]. BWT has several distinct advantages over the real field wavelet transform, such as no quantization is introduced during the transform and it is computationally efficient since only simple Boolean operations are involved. As well as BWT transformed image has same number of grayscale levels as the original image thus, it reduces memory storage requirement. In Chapter 3, BWT based features are proposed for CBIR.

A multiresolution technique for texture analysis is also available with Gabor wavelet transform (GWT) on four scales and six orientations [90]. Frequency and orientation representations of Gabor filters match with human visual perception. Further, texture image retrieval result is enhanced through Gabor wavelet correlogram (GWC) [91] and redundant information is suppressed. GWC computes autocorrelogram of Gabor wavelet subbands along the direction normal to the Gabor wavelet orientations. GWC is enhanced by optimizing the quantization thresholds using EGA [92]. Rotation invariance in GWT is achieved by keeping the dominant directional feature element at the first position and circularly shifting the other elements [93]. To obtain another type of rotation invariance in Gabor texture features, filter response at each scale is generated by summation of all orientations filter responses at corresponding scale. Similarly, scale invariance in Gabor texture features are generated by summation of all scales filter responses at each particular orientation [94]. Maximum bandwidth of a Gabor filter is limited to approximately one octave and Gabor filters are not optimal if broad spectral information with maximal spatial localization is required. To avoid the limitations of Gabor filters, Field proposed the log Gabor filter (LGF) [95]. Extended tail of LGF is able to encode natural images more efficiently by better representation of higher frequency components than Gabor functions which over represent low frequency components and under represent the high frequency components. Log Gabor filters basically consist in a logarithmic transformation of the Gabor filters. In 2007, Gao *et al*. [96] used LGWT for corner detection in gray level images. In Chapter 3, LGF based features are proposed for CBIR application.

Many other transform domain techniques are also available in literature mainly for texture classification. For example Shearlet transform is also proposed to analyze information at multiple scales [97]. Histogram of shearlet coefficients is computed at each scale to estimate the edge responses and also it is used as a feature for texture classification. Various other wavelets like ridgelet transform, curvelet transform etc. are proposed for texture information extraction. Ridgelet transform is proposed in [98]. It provides information along variety of directions so it can represent edges efficiently [99]. Its computation is a two step procedure i.e. radon transform and 1-D wavelet transform computation. Concept of M-band wavelet is combined with ridgelet transform to introduce M-band ridgelet transform [100]. It is capable of detecting rich middle and high frequency components more efficiently as compared to ordinary ridgelet transform. Fingerprint pattern can be considered as an oriented texture of ridges. So, in [101] ridgelet based energy and cooccurrence signatures are used to characterize fingerprint texture. Rotation invariant texture features by ridgelet transform [99], as well as multi scale ridgelet transform [102] are also proposed. Curvelet transform extends the ridgelet transform to multiple scale analysis and was originally proposed for image denoising [103]. Curvelet transform allows capturing edge and linear information. Further it is used for image retrieval purpose also [104]. In contourlet transform, Laplacian pyramid is used to decompose an image to multiple scales and directional filter bank is applied to capture directional details at each scale. In [105] texture feature vector of an image is constructed by computing Gaussian density function of all contourlet decomposed subbands. In [106] a multifocus and multispectral image fusion technique is proposed by multiresolution decomposition of image by using wavelet, wavelet packet and contourlet transform. It utilizes information available at finer levels of the decomposition to assign more weight to the coefficients of shaper neighborhood.

Kingsbury *et al.* [107] introduced dual tree complex wavelet transform (DT-CWT) to extract more directional information of the image. DT-CWT exhibits directionally selectivity with approximate shift invariance, limited redundancy and computational efficiency characteristics. DT-CWT provides texture information strongly oriented in six different directions at each scale, which makes it able to distinguish positive and negative frequencies. In [108] combination of spatial orientation tree (SOT) and DT-CWT is used for feature collection and retrieval of the images from natural as well as texture image database. SOT represents features by parent-offspring relationship among the wavelet coefficients in multi-resolution wavelet sub-bands. These feature vectors are indexed by using vocabulary tree. Texture image retrieval is proposed by DT-CWT in [109] also and texture features are obtained by computing energy and standard deviation on each subband of the DT-CWT decomposed image. Dual tree rotated complex wavelet transform (DT-RCWT) is proposed for texture image retrieval [110]. DT-RCWT is a non-separable wavelet designed by rotating DT-CWT filters with 45°, which gives texture information strongly oriented in six different directions (45° apart from DT-CWT). Compound response of DT-CWT and DT-RCWT [110] furnish textural information in 12 different directions. To incorporate rotation invariant property in DT-CWT and DT-RCWT combination, at each level average of all subband feature values are computed [111]. It improves characterization of rotated textures.

1.2.3 Image retrieval methods using integrated features

Information integration and information fusion techniques are used to produce the most comprehensive and specific datum about an entity from data supplied by several information sources [112]. In the process of information fusion the task of combining the information is performed by aggregator operator. Information from multiple sources is combined by using fuzzy integral technique in [113].

In the case of CBIR color and texture both explain significant information of an image. When only color features are extracted textural behavior is lost and when texture features are extracted from the gray level images color information is lost. So to improve the performance of CBIR it may be desired to include both color and texture features that are complementary to each other without tremendous change in feature vector dimensions. Similarly, there are various ways to combine different visual feature information. One way to utilize both color and texture information is to extract texture features from color images. Color correlogram [38] and MCM [55] are the best examples for texture feature extraction from color images. It facilitates color information from color space and texture features are extracted by spatial relationship of color pixels. Few other examples of combined color and texture features are TCM [56], MTH [57] and MSD [58].

Likewise, various other visual feature combinations can also increase the retrieval performance. Color and shape invariant features are combined into a unified invariant feature for the purpose of image indexing and retrieval. This feature extraction scheme proposes invariant color models and utilizes these color models for edge detection. Shape invariant features are computed from these edges [114]. Color and texture descriptors of MPEG-7 are presented in [115]. In this, color descriptor combines histogram descriptor, dominant color descriptor and color layout descriptor. Similarly, texture descriptors are extracted to characterize homogeneous texture regions and local edge distribution. A spatial-temporal extension of MPEG-7 edge histogram descriptor is proposed in [116] for content based video retrieval.

Combination of multiple features has a problem for assigning weights for these features. In [117] an automatic feature weight assignment approach is proposed for integrating multiple features through genetic algorithm. Weight for each feature is optimized through genetic algorithm in order to improve the retrieval result. In [118] weights for color and texture features can be assigned according to the image texture characteristic. This weight assignment operator assigns weight to the visual features adaptively and realizes image retrieval with combined visual features.

Vadivel *et al.* [119] proposed an integrated color and intensity cooccurrence matrix (ICICM) for CBIR. ICICM captures color and intensity variations around each pixel. For each pair of neighboring pixels, contribution of both color as well as gray level is considered. HSV color space has the property to represent color and intensity perception at the same time thus; images are converted into HSV color space for ICICM feature extraction. In [120] RGB image is converted into HSV color space and with CSS it is used for color and shape information extraction.

Lin *et al.* [121] proposed a smart content based image retrieval system with the help of three image features. The first and second image features of this system are based on color and texture features, called color cooccurrence matrix (CCM) and difference between pixels of scan pattern (DBPSP) respectively. The third image feature is based on color distribution, called color histogram for *K*-mean (CHKM). Color and texture feature are also combined in [122] to introduce cluster correlogram by using MPEG-7 primitive features (color layout, color structure and edge histogram) to improve the indexing performance. Shape features are different from other elementary visual features, such as color and texture so, color, texture and shape features are combined in terms of dynamic dominant color, steerable filter and pseudo-Zernike moments respectively for color image retrieval [123]. In [124] a new entropy function is designed and color, texture features are extracted by using the new entropy function. It is considered for image retrieval with the combination of dominant color feature.

Color and texture features can also be extracted in multiresolution wavelet domain to enhance the effectiveness of retrieval system. In [125] multiresolution information is extracted by applying wavelet transform on HSV color image and color features are extracted by computing color autocorrelogram on the hue and saturation components of HSV wavelet transformed coefficients. Further, texture features are computed in term of BDIP (block difference of inverse probabilities) and BVLC (block variation of local correlation coefficients) moments through intensity value component of HSV wavelet transformed coefficients. Color features are also combined with texture features by applying DWT on color scale image and from vertical, horizontal and diagonal coefficients of each color component, histogram mean and standard deviation are computed [126]. Performance comparison of first order statistics, second order statistics, GWT, DWT and their combination is carried out for CBIR and it is observed that combination outperform their individual results [127]. To improve the retrieval performance color and texture information is also combined through color layout descriptor (CLD) and GWT texture descriptor [128]. CLD computes spatial distribution of colors while frequency distribution of colors is calculated by GWT texture descriptor. These two descriptors extract different properties of an image thus retrieval performance is improved.

Integrated features are also popular in color texture classification. Like, in [129] color, texture features are combined and integrative single channel and multichannel cooccurrence matrix is proposed. Single channel cooccurrence matrix computes gray scale cooccurrence successively to separated color channels while multichannel cooccurrence matrix computes correlation between textures of different color channels. A rotation invariant coordinated cluster based model for color textures classification is proposed in [130]. The basic idea of the method is to split the original color image into a stack of binary images. These binary images are characterized by probability of occurrence of rotation invariant texels and overall feature vector is obtained by concatenating the histograms computed from binary image.

There may be several other methods also to combine multiple features. Two such methods are suggested in Chapter 3 and Chapter 6.

1.2.4 Application of shape features in image analysis

Most of shape features need proper segmentation before feature calculation. Segmentation of natural images is difficult so shape features are mostly used in domain specific images such as in manmade objects [1]. Shape of an object is sensitive to various changes so, a good shape feature of an object should be invariant to translation, rotation and scaling [6]. Simple shape descriptors calculate area, circularity, eccentricity and major axis orientation. These simple shape descriptors are capable of discriminating shapes only with large differences [18]. A Shape signature represents shape by a one dimensional function derived from shape boundary points [131]. Shape signatures are sensitive to noise and slight changes in the boundary can cause large errors in image matching. Fourier descriptors utilize the Fourier transform of object boundary as the shape feature while moment invariants shape descriptor uses region based moments which are invariant to transformations [6, 132]. There is a dramatic increase in complexity with increment in the order in moment invariants.

Gradient based shape signature is proposed by aggregation of gradient information obtained by two dimensional steerable *G*-Filters [133]. It provides gradient information at different orientations and scales. Shape descriptor is computed by Fourier transform of this shape signature. In [134] a curvature scale space (CSS) based shape descriptor is proposed to detect shallow concavities in objects and based on that CBIR is performed. In [135] a novel shape signature, farthest point distance (FPD) is proposed. It includes corner information also to enhance the performance of shape retrieval by using Fourier descriptors.

Many other shape features like Legendre polynomials [136], Zernike polynomials [137], Fourier–Mellin moments [138] are also proposed for shape representation. To construct rotationally invariant moments, multiresolution information is utilized. High and low frequency wavelet coefficients at analysis part are combined to eliminate noise sensitive features [139]. Shape features are also proposed with the help of moment invariants and Zernike moments [140] for image retrieval. *k*-means clustering is used to group similar images into *k* disjoint clusters then neural network compares similarity among images and returns the cluster with largest similarity to the query image. Other than CBIR shape feature is also being used in hand gesture recognition. In [141] performance of Zernike, Tchebichef and Krawtchouk moments are examined for hand gesture recognition and it is observed that Krawtchouk moment achieves better result. In [142] for hand gesture recognition, trajectory features (shape of hand trajectory, length of the extracted trajectory, location of key trajectory points and orientation of hand)are combined with dynamic features (velocity and acceleration).

1.2.5 Application of local features in image analysis

It is possible to extract image visual features in two different manners viz. globally or locally. Global feature are efficient and easy to extract but they are less robust for viewpoint variations. Relative amount of color and texture typically varies in such cases. Hence, it is required to capture image features in a way to make them robust against changes in viewpoint and illumination. Local features, distinctive and invariant to many kinds of geometric and photometric transformations are being used in computer vision research, such as in image retrieval, image registration, object recognition and texture classification [143]. Local features explore the visual properties of regions or objects in an image rather than the whole image as in global features. In the development of local feature descriptor, scale invariant feature transform (SIFT) is proposed by Lowe [144]. A relevance feedback algorithm for SIFT descriptors was proposed in [145] and user is entered into the retrieval system. SIFT feature descriptor is invariant to scale, orientation and partially invariant to affine distortion and illumination changes. In SIFT, the locations of extrema in difference of Gaussian (DoG) correspond to the most stable feature with respect to scale variance, and are identified as the interest feature points or keypoints. For each keypoint, sequel histogram of oriented gradient (HOG) is calculated. HOG computes a histogram of gradient orientation in a certain local region. It is introduced for solving the problem of pedestrian detection in static images [146]. Further its application is expanded to human detection in film and video, as well as to a variety of common animals and vehicles in static image [147]. This task was challenging due to pose, illumination, background and camera position changes. HOG feature descriptor is mainly used in computer vision and image processing for object (person and face) detection. In the algorithm presented by Zhu *et al*. [148], person detection is performed by calculating HOGs for variable size blocks and the most informative block features are selected by Adaboost algorithm automatically. In [149] application of HOG is found in face recognition. Elastic bunch graph matching (EBGM) algorithm is used to recognize the positions of eyes, nose and mouth facial landmarks and every face is represented as graphs with nodes at these facial landmarks. Albiol *et al.* [149] has replaced the Gabor wavelet coefficients by HOG in EBGM to represent each node in face recognition algorithm. Scale space HOG (SS-HOG) feature descriptor is introduced for person detection [150]. In this method, HOG features are calculated for multiple scaled images separately and combined to construct the SS-HOG feature vector. Nanni *et al*. [151] evaluated the performance of HOG descriptor for fingerprint matching. Further also, several contributions are given to improve the human, face and fingerprint detection by using HOG feature descriptor. In the Chapter 2, of this research work, HOG descriptor is applied for image retrieval.

Jayaraman *et al*. [152] combined color as well as texture feature for iris image retrieval. Color indices are obtained by averaging R, G and B intensity values and are applied for indexing by using Kd-tree. Texture feature are also extracted by speeded up robust features (SURF) and it is suggested SURF feature gives good discriminative power with low dimensional feature.

One more local descriptor contrast context histogram (CCH) is proposed for image matching by utilizing the contrast distributions of local regions [153]. CCH descriptor is computationally efficient and is capable of representing local regions by more compact histogram bins with geometric and photometric invariance. Region based algorithm is also proposed for image retrieval [154]. Firstly, image is segmented into several regions and spatial color distribution entropy is designed for each region. It combines local color feature of the region as well as spatial color distribution information of region.

Other than CBIR application, in [155] palmprint detection is performed by partitioning the region of interest into a fixed number of non overlapping windows and features (sigmoid, energy and entropy) are extracted from each window. It is observed that sigmoid and entropy show better accuracy as compared to energy. In [156] the concept of a variant of histogram i.e. histon is introduced. Histon is used as upper approximation and the fusion of rough-set theoretic approximations and fuzzy c-means algorithm is proposed for color image segmentation [156]. Another technique involves image segmentation by using a histon, based on rough-set theory. In this a relation between roughness measure and intensity value is utilized as a basis for image segmentation [157].

Interest point detection also plays important role for local feature extraction. Local image features are computed from these interest points for image matching. Use of color in interest points provides high discriminative power for image matching thus, color interest point is proposed for sparse image representation [158].A signature based symbolic image representation is generated by a set of feature points and is proposed to reduce the number of comparison in the initial stage [159]. Later, Hierarchical decomposition of the feature image is performed to find out the spatial relationship among feature points for image representation.

Local features are also computed from multiresolution images. In [160] GWT is applied to extract frequency and orientation information in fingerprint images and texture features are computed locally through filtered GWT subbands so as to extract the average absolute deviation from mean. Global texture information is also extracted by calculating mean and variance of DWT detail subbands and these local and global features are fused together to obtain the final features of fingerprint. Gabor based local invariant feature is proposed for palmprint recognition [161]. It finds relationship between local lower-layer subblocks and upper-layer blocks based on Gabor features. These features are locally invariant to global noises occurred by variations of position and illumination.

1.2.6 Similarity measures and indexing techniques for image retrieval

Once a decision on the selection of feature descriptors is over, how to use them for accurate image retrieval is the next concern. Instead of exact image matching, CBIR system calculates visual similarities between feature descriptors of query image and database images. There are various similarity measures to find out the similarity based on the extracted features. Each similarity measure shows different retrieval performance. It is required to have fast similarity measurement because the query image is compared with a large set of images to find relevant images. Minkowski-form distance (Manhattan distance and Euclidean distance) is the most widely used similarity metric [1, 6]. Histogram intersection distance is proposed to find out similarity between color images depending upon their histogram [31] feature.

In [162] feature extraction is proposed through Bhattacharyya distance. In this the classification error is approximated by using the error estimation based on the Bhattacharyya distance. It helps to predict the minimum number of features needed for classification without significant information loss. In [163] a fuzzy distance measure is proposed by using the gamma membership function. It calculates the membership values of the gray level histogram for similarity measurement. A normalized information distance (NID) with simple compression based approximations is proposed to measure the similarity among images by bypassing the feature selection step [164]. Some other similarity metric like Canberra distance, Mahalanobis distance, quadratic form (QF) distance, Kullback-Leibler divergence (KLD), earth mover's distance (EMD), Hosdorff distance [1, 4 ,6, 7] are also being used by researchers. Cosine distance and normalization of cosine, Euclidean distance is proposed for trademark image retrieval [165]. It is observed that normalized distance outperform their original results.

In case of large databases it is also required to have efficient indexing structure so as to minimize the retrieval time. Many tree structured indexing algorithms such as K-d tree, quad-tree, R-tree etc. are proposed for this purpose [7]. In Chapter 1, vocabulary tree is proposed for retrieval through large feature database because it can handle feature indexing of large feature database very efficiently [166]. In the indexing process of vocabulary tree, feature descriptors are hierarchically quantized into clusters and are arranged in the form of vocabulary tree. It reduces number of comparison in the similarity measurement.

1.3 THE AUTHOR'S CONTRIBUTIONS

It is concluded from literature survey that the CBIR system performance mainly based on low level visual features. Low level features are selected or designed in such a manner so as to fulfill the user intention. Single low level feature considers only one perception while multiple low level features can consider an image from different perception. Thus, performance of CBIR systems can be improved by considering the proper combination of effective features. Lot of research is carried out in last two decade but, still there is a scope to find better feature representation so as to enhance the performance of CBIR system. This research work proposes several approaches to improve the retrieval performance of CBIR systems.

There are two types of visual content descriptors viz. global and local. A global descriptor computes visual features of the whole image at the same time, whereas a local descriptor calculates visual features of regions or objects to construct the final image feature. Thus, local feature is capable of capturing small intensity variations present in the image. In the first approach a local feature descriptor viz. histogram of orientated gradient (HOG) is proposed to describe the image content. Experiments are performed on Corel 1000 and Corel 2450 standard image databases (Appendix A) [167]. Performance of CBIR system based on the HOG feature is compared with CBIR systems based on the global feature viz. Gabor wavelet transform (GWT) [168] as well as with local feature scale invariant feature transform (SIFT) [144]. Local feature produces a high dimensional feature vector so for indexing of the high dimensional feature vectors vocabulary tree is applied. Performance of features is compared with respect to average precision (*AP*), average recall (*AR*) and average retrieval rate (*ARR*). It is observed that HOG feature performs better than GWT and SIFT with respect to various performance measures.

Since better image information can be extracted by multiscale and multiresolution analysis of images thus, in next approach multiresolution features are proposed to improve the CBIR performance. Log Gabor filter (LGF) is proposed to extract texture feature from three scales and four orientations and its performance is compared with respect to GWT [168]. It is observed that LGF better represents a natural image as compared to GWT because of reducing the overlapping between subbands. Experiments are conducted on Corel 1000 and Corel 2450 standard image databases (Appendix A) [167] and it is verified that LGF based retrieval system improves *AP*, *AR* and *ARR* as compared to GWT based retrieval system.

Another multiresolution feature is proposed by incorporating the binary wavelet transform (BWT). BWT provides image information at multiscale and multiresolution in computationally efficient manner since only simple Boolean operations are involved [89]. BWT returns intensity levels same as the input image. These properties of BWT have encouraged us to propose the binary wavelet transform based histogram (BWTH) feature. Statistical properties of BWT decomposed subbands are extracted in terms of histogram to design the BWTH feature. Further, image color information is also included in BWTH to improve the retrieval performance. The performance of CBIR system with BWTH feature is observed in color as well as gray space on standard image databases, Corel 1000 and Corel 2450 (Appendix A) [167]. It is verified that BWTH performs superior to color histogram (CH) [31], auto correlogram (AC) [38], discrete wavelet transform (DWT) [77], optimal quantized wavelet correlogram (OQWC) [83] etc. with respect to various performance measures. Since, BWTH does not consider relationship among neighboring pixels in subbands so further, histogram and correlogram features are integrated together and integrated feature of BWT (IFBWT) is proposed. It is experimented that retrieval performance is significantly improved by IFBWT as compared to OQWC [83], Gabor wavelet correlogram (GWC) [91], AC [38] and BWTH.

In the next multiresolution approach, first á trous wavelet correlogram (AWC) is proposed. AWC computes relationship among á trous wavelet transformed coefficients. Orientation information is lost in AWC. So, to include it by further extension, á trous gradient structure descriptor (AGSD) is proposed. AGSD facilitates feature extraction with the help of local orientation information of á trous wavelet coefficients and microstructures are applied to calculate relationship between neighboring orientations. The constructed microstructure image is used to map á trous wavelet image and correlation is computed to obtain AGSD feature. The experiments are conducted on three standard image databases viz. Corel 1000, Corel 2450 and MIRFLICKR 25000 (Appendix A) [167, 169] to compare the performance of AGSD with OQWC [83], GWC [91], combination of standard wavelet filter (SWF) with rotated wavelet filter (RWF) correlogram [170], combination of GWC and evolutionary group algorithm (EGA) [92] and texton co-occurrence matrix (TCM) [171] with respect to various performance measures. It is concluded that AGSD has improved the retrieval performance significantly.

In next approach, cooccurrence of Haar-like wavelet filters (CHLWF) is proposed. First and second order Haar-like wavelet filters are utilized to obtain coarse and fine edge information from an image. Out of the set of Haar-like wavelet filters, the dominant filter is selected to capture direction of maximum edge response. Statistical information of dominant Haar-like wavelet filter is extracted for proposed image retrieval framework. Selection of dominant filter for feature extraction allows capturing of the most contributing edges with reduction in the computational complexity. Performance of CHLWF feature is verified on various standard image databases viz. Corel 1000 [167], Brodatz [172, 173] and MIT VisTex [174] (Appendix A) with respect to various performance measures. Experimental results have demonstrated that CHLWF is more effective image features for image retrieval as compared to existing related features like OQWC [83], GWC [91], SWF+RWF correlogram [170], CC [38], GWT [90], DTCWT, DT-RCWT [110] and DTCWT + DT-RCWT [110].

Integrated features are attracting researcher’s interest because of better image representation capability by incorporating different perceptions into account. In Chapter 6, weight cooccurrence based integrated color and intensity matrix (WCICIM) is proposed and its combination with integrated color and intensity cooccurrence matrix (ICICM) is applied for CBIR. It combines color and intensity properties of the image at the same time through HSV color space. Weight matrixes are constructed by applying suitable weights function to decide the contribution of color and intensity for each pixel. WCICIM finds correlation among color–color, color–intensity, intensity–color and intensity–intensity based on neighboring pixel variations in weight matrixes. Proposed integrated feature improves the retrieval performance significantly as compared to motif cooccurrence matrix (MCM) [55], color correlogram (CC) [38], combination of block bit plane (BBP) and global color histogram (GCH) [175] and ICICM [119] features on standard Corel 1000 and Corel 2450 image databases(Appendix A) [167].

1.4 ORGANIZATION OF THESIS

Present chapter discusses the mechanism of CBIR system. Detail description of CBIR algorithm is explained through block diagram and each block is discussed briefly. Types of visual features followed by visual feature extraction techniques and similarity measures are illustrated. Further practical applications of CBIR are overviewed. State of art image retrieval systems are also presented here followed by the relevant literature reviewed throughout this research work. In Chapter 2, a local spatial feature descriptor viz. HOG was applied to describe the image content. In Chapter 3, multiresolution features viz. LGF, BWTH and IFBWT are proposed by using transform domain techniques. In Chapter 4, two new transform domain methods viz. AWC and AGSD are proposed for feature extraction. In Chapter 5, a novel technique is proposed as cooccurrence of Haar-like wavelet filters (CHLWF) for CBIR. In Chapter 6, the retrieval performance is improved by the combination of WCICIM and ICICM features. Finally, In Chapter 7 conclusions from this research work are derived and further directions for future work are also suggested. To compare the performance of various features it is required to check their performances on same standard databases. At last, Appendix A incorporates specifications of all the image databases used in this research work to validate the retrieval performance of proposed features with respect to existing methods.

Histogram of Oriented Gradient Feature

Chapter 2

In CBIR images are represented by visual features (or descriptors) hence, extraction of features plays an important role in the performance of retrieval system. These visual features can be obtained from an image in two different manners, viz. local feature extraction and global feature extraction. Global features concern about overall characteristics of images while local features take into account the variations in small portions of the images. From the literature survey presented in Chapter 1 (section 1.2.5) it is found that use of local features can substantially enhance the performance of image retrieval system. This is due to the property of local feature to capture finer details of the picture. With this aspect, in this chapter the task of image retrieval is proposed by extracting the visual features locally through histogram of oriented gradient (HOG).

The concept that motivates to propose HOG descriptors is that the local object appearance and shape within an image can be well described by the distribution of intensity gradients and edge directions. HOG feature extracts intensity variations as well as direction of intensity variations in the local regions and this information is arranged in the form of histogram of oriented gradient.

Features extracted from local regions of an image together construct the final image feature vector, thus, it generates the high dimension feature vector. In order to avoid the problem of indexing of high dimensional features, it is required to apply an efficient and fast indexing scheme. Thus, in this chapter features are indexed by vocabulary tree. Feature descriptors of local regions are quantized hierarchically and unsupervised training is performed to build the vocabulary tree. Hierarchical approach of vocabulary tree reduces the computational complexity of similarity measurement and provides shift invariance property as well.

The retrieval performance of HOG feature is compared with global feature in order to prove the superiority of local feature over global feature. It is verified that HOG feature performs better than global feature descriptor, viz. Gabor wavelet transform (GWT) [168]. In order to establish the efficacy of HOG features among local features, the performance of HOG is also compared with standard scale invariant feature transform

(SIFT) feature [144]. In the feature extraction process of SIFT, only key points are considered on the other hand in HOG entire image is divided into sub-blocks and features are extracted. Hence, HOG is capable of collecting features from entire image whereas SIFT losses the information available from the locations other than key points. Experimental results illustrate the comparative analysis of retrieval system based on HOG and GWT feature descriptor. All the experiments are performed on Corel 1000 and Corel 2450 standard image databases (Appendix A) [167]. It is verified that on Corel 1000 database, HOG based retrieval system improves the retrieval performance in terms of average precision (*AP*) and average recall (*AR*) (56.75% and 38.45% respectively) as compared to GWT based retrieval system (41.20% and 25.41% respectively) and SIFT (47.08% and 30.79%respectively). Also, it is observed that the average retrieval rate (*ARR*) of HOG based retrieval system is always superior to GWT based retrieval system for all number of retrieved images. In the same manner on Corel 2450 database *AP* of HOG is 34.20% while for GWT and SIFT, *AP* results are 21.65% and 25.62% respectively for 10 retrieved images.

2.1 HISTOGRAM OF ORIENTED GRADIENT

In the present chapter, HOG feature descriptor is proposed for image retrieval. HOG feature descriptor is very effective to represent objects. The concept behind the HOG descriptors is that the local object appearance and shape within an image can be well described by the distribution of intensity gradients and edge directions. Gradient captures variation in intensity and orientation provides direction of intensity variation at any particular pixel. To obtain the local visual descriptors, first an image is divided into parts. The simplest way of dividing an image is to separate the image into tiles of equal size and shape. For the implementation of HOG feature descriptors, input image is divided into small connected regions, called cells and (2 × 2) cells together form a block (refer Fig. 2.1). Gradient and edge orientation information are calculated for each pixel within the cell and are arranged to form HOG descriptor. For making the descriptor invariant to illumination changes, the local histograms are normalized by Gaussian normalization across a larger region of the image (called block). HOG feature calculated from each block represents one component of image descriptor. In a similar manner, HOG feature is computed for the whole image. The HOG descriptor has few key advantages over other descriptor methods. Since the HOG descriptor operates on localized cells, the method upholds invariance to geometric as well as photometric transformations; as such changes would appear only in larger spatial regions.

Fig. 2.1 shows HOG descriptor calculation for sample bus image from group number 4 of Coral 1000 image database (Appendix A) [167]. Each image is divided into cells of size 8 × 8 pixels as shown in Fig. 2.1. For HOG feature construction orientation values $\left(-90^\circ \textit{ to } 90^\circ\right)$ are quantized into 9 equally spaced orientation bins and sum of gradients on particular orientation is computed. If $L(x, y)$ is the intensity of a pixel at coordinate (x, y) then gradients and orientation values can be calculated by using Eq. (2.1) and (2.2), respectively.

$$Grad(x,y)=\sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2} \tag{2.1}$$

$$Orient(x,y)=\tan^{-1}((L(x,y+1)-L(x,y-1))/(L(x+1,y)-L(x-1,y))) \tag{2.2}$$

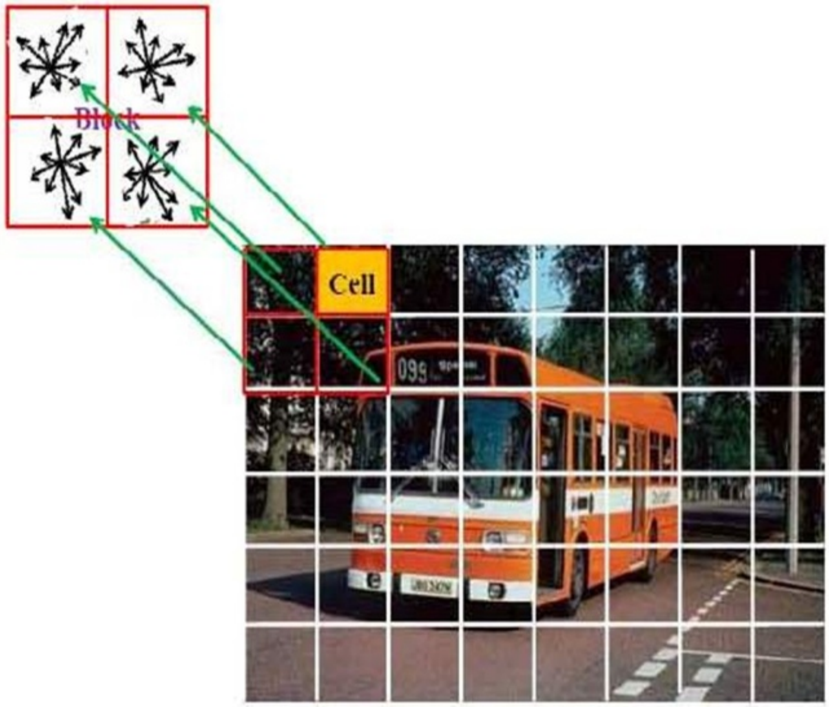


Fig. 2.1: HOG feature descriptor computation

Features are calculated in a sliding fashion of block with overlapping of 50%. For normalization of feature, each group of 2 × 2 cells is integrated as a block and HOG of cells are normalized by Gaussian normalization within the block. The final descriptor of each block is constructed by normalized HOG descriptor of cells within that block. Therefore, each block is represented by a 36-D normalized feature vector. Each cell is normalized four times by different values. Although this increases the redundancy but performance is significantly improved.

2.2 INDEXING AND SIMILARITY MEASUREMENT

Feature extraction process of proposed method generates a large size of feature database. Indexing of high dimensional features is a tedious task. In the presented work, such kinds of features are indexed by vocabulary tree as it can handle large size of feature database very efficiently. In the following section construction of vocabulary tree, indexing and retrieval through vocabulary tree is explained.

2.2.1 Vocabulary tree

Image retrieval becomes a challenging problem when size of the feature database increases. Nister *et al.* [166] introduced vocabulary tree as the solution for handling large amount of feature data, it enables extremely efficient retrieval. Unsupervised training of this tree is performed by *k*-means algorithm on large set of representative descriptor (feature) vectors where, *k* (=10) numbers of cluster centers are defined. The whole training data is then spread into these *k* clusters. Each descriptor is assigned to such a cluster which has minimal distance between descriptor and cluster centre. The same process is recursively applied on each group of descriptors at every cluster centre. The tree is designed level by level upto some fixed number of levels *N*. Fig. 2.2 illustrates the building process of vocabulary tree.

After tree construction, in the mapping phase, every descriptor of the image is propagated down the trained tree by comparing the descriptor with *k* cluster centers at each and every level. Whenever any descriptor is assigned to a leaf node, the count of that node increases by one. After processing the last descriptor of an image, a new descriptor vector for that particular image is obtained in terms of frequencies of leaf nodes. This provokes the sum of new image descriptor values equals to the number of descriptors present in that image.

Likewise new descriptor vectors are generated for the whole image database and query image.

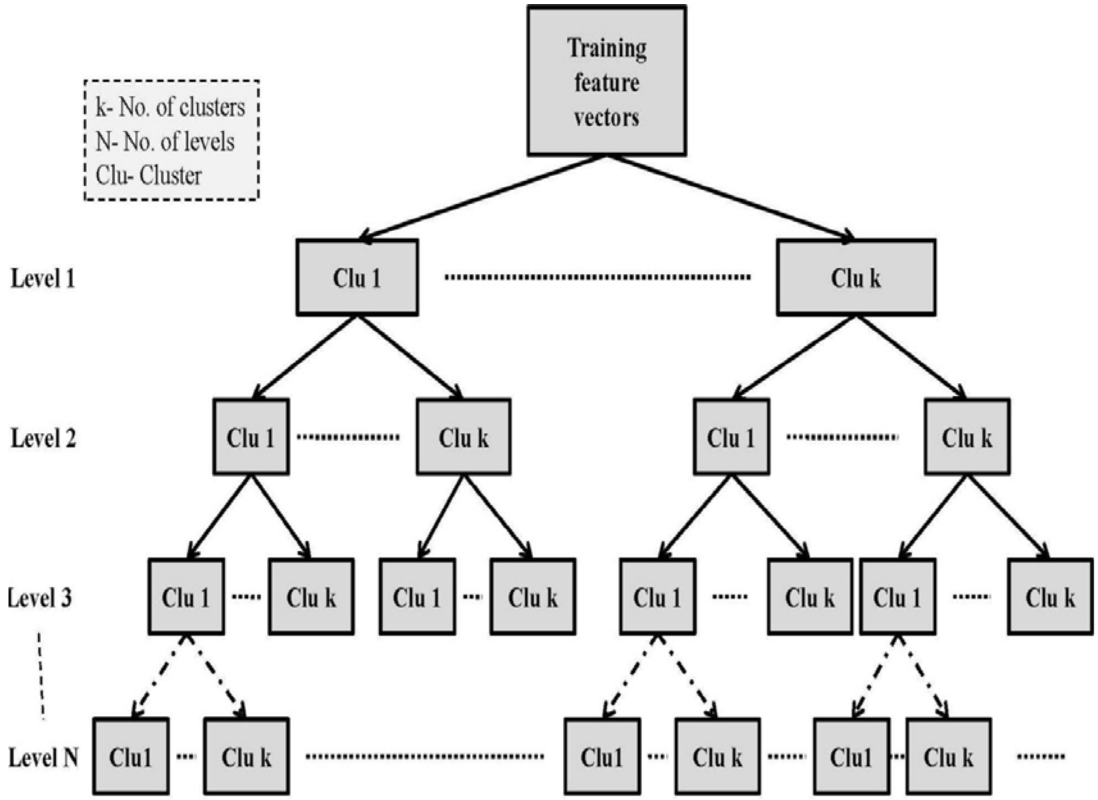


Fig. 2.2: Building process of vocabulary tree

Descriptor vector F_{Ni}, at N^{th} *level* and i^{th} leaf node is given by Eq. (2.3).

$$F_{Ni} = m_i w_i \tag{2.3}$$

Where, $w_i = \ln \frac{DB}{DB_i}$

Where, m_i is the number of descriptors of database image that pass through the leaf node i, w_i is the weight of leaf node i, DB is the total number of images in the database and DB_i is the number of images in the database with at least one descriptor passing through node i.

After the new descriptor vector calculation by using Eq. (2.3), normalization is required to achieve fairness between database images with different descriptor values. Therefore, the final normalized descriptor vector at level N and node *i is* given in Eq. (2.4)

$$F_{Nor} = \frac{F_{N\,i}}{\left\| F_{N\,i} \right\|} \tag{2.4}$$

Algorithm for feature indexing and image retrieval

1. Construct the vocabulary tree by using the HOG feature database, as shown in Fig. 2.2.
2. Map each image feature on vocabulary tree constructed by step 1.
3. Calculate the new feature vector of length $= k^N$, where k is the number of clusters and N is the number of levels (here, k = 10, N = 4) from step 2 for each image.
4. Normalize the features obtained after step 3 and calculate final feature vector.
5. Query feature vector T_i and database image feature vector Q_i are obtained by using Eq. (2.4) and relevant image are retrieved by computing score between them through Eq. (2.5), highest scoring database image is treated as more relevant to query.

$$Score = \sum_{i=1}^{k^N} Q_i.T_i \tag{2.5}$$

Where, T_i is query feature vector, Q_i is database image feature vector, *k=10* is the number of clusters and *N=4* is the number of levels in vocabulary tree.

2.3 EXPERIMENTAL RESULTS AND DISCUSSIONS

In the presented work, authors have conducted all the experiments on the standard Corel 1000 and Corel 2450 image database (Appendix A) [167]. Various performance measures like average precision *(AP)*, average recall *(AR)* and average retrieval rate *(ARR)* are calculated to evaluate the retrieval performance of system. These performance measures are given by following Eq.(2.8), (2.11) and (2.12) respectively. For i^{th} image, precision is given by Eq. (2.6).

$$Precision\,(P_i) = \frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ retrieved\,images\,considered\,(T)} \tag{2.6}$$

$$Group\ precision\,(GP) = \frac{1}{N}\sum_{i=1}^{N} P_i \tag{2.7}$$

$$Average\ precision\,(AP) = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} GP_i \tag{2.8}$$

Similarly, recall of i^{th}image is given by Eq. (2.9).

$$Recall\,(R_i) = \frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ relevant\,images\,in\,database\,(N)} \tag{2.9}$$

$$Group\,Recall\,(GR) = \frac{1}{N}\sum_{i=1}^{N} R_i \tag{2.10}$$

$$Average\,Recall\,(AR) = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} GR_i \tag{2.11}$$

$$Average\,Retrieval\,Rate\,(ARR_i) = \frac{1}{DB}\sum_{j=1}^{DB} \left.\frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ relevant\,images\,in\,database}\right|_{T\le 100} \tag{2.12}$$

Where, N is number of relevant images in the database, Γ is number of groups in the database and DB is the number of images in the database. The retrieval result is not a single image rather a list of images depending upon relevancy. T gives the total number of retrieved images considered $(e.g.10, 20, ...,100)$. Value of T is selected by user.

Results of our proposed method are compared with GWT and SIFT based method on the basis of above mentioned performance measures. In [168] GWT texture features are extracted from Corel 1000 database (Appendix A) [167] for image retrieval. For feature extraction images are decomposed into three scales and four directions. Through GWT texture features are constructed by computing mean and standard deviation from each GWT transformed subbands.

2.3.1 Corel 1000 Database (DB1)

The first experiment is performed on database Corel 1000 image database (Appendix A) [167]. Table 2.1 shows the precision and recall comparison of GWT, SIFT and proposed method for each group of DB1. For group numbers 6, 7 and 10 proposed method is giving lesser result than GWT because the proposed HOG feature is a local feature whereas GWT feature is a global feature. Local features are calculated on small parts of the image whereas global features consider whole image at the same time. Sometimes, small part of one group’s image may match with small part of another group’s image, e.g. flower parts (group number 7) with food parts (group number 10). Because of this property;

sometimes local features show lesser result. But, this property of local features permits extraction of fine details present in the image, so most of the times results are improved as in the case of group numbers 1, 2, 3, 4, 5, 8 and 9.

Table 2.1

Precision and recall comparison for GWT, SIFT and proposed method

Group Number	Group Name	GWT		SIFT		Proposed method	
		P% (T=10)	R% (T=100)	P% (T=10)	R% (T=100)	P% (T=10)	R% (T=100)
1	Africans	32.20	17.37	32.80	22.19	39.50	28.92
2	Beaches	28.80	18.08	21.40	11.30	41.90	25.52
3	Buildings	28.90	12.88	41.50	24.13	61.00	32.29
4	Buses	35.12	29.48	65.30	43.89	88.90	67.06
5	Dinosaurs	80.19	55.14	99.60	93.21	96.80	84.40
6	Elephants	40.15	25.80	28.10	15.14	24.30	18.00
7	Flowers	63.30	43.53	76.90	40.83	72.50	35.72
8	Horses	38.10	16.00	32.40	13.35	85.30	56.96
9	Scenery	31.60	17.25	30.90	16.43	31.80	21.04
10	Foods	33.70	18.60	41.90	27.51	25.50	14.57
Average		41.20	25.41	47.08	30.79	56.75	38.45

It is also observed from Table 2.1 that the average precision of SIFT is lesser than proposed HOG feature. The reason behind is that in SIFT only key points are considered for feature extraction while in HOG entire image is divided into sub-blocks for performing the same task. Hence, HOG is capable of calculating features more efficiently whereas SIFT is losing the information available on the locations other than key points.

For performance evaluation of retrieval methods on the image database, commonly average of all group precision and group recall are being computed by using Eq. 2.8 and 2.11 respectively. From Table 2.1, it is clear that the proposed method gives better performance in terms of *AP%* and *AR%* both (56.75 and 38.45 respectively) when compared with GWT (41.20 and 25.41 respectively) and SIFT (47.08 and 30.79 respectively). Fig. 2.3 and 2.4 show graphical representation of group precision (*GP%*)

and group recall (*GR%*) of proposed method with respect to GWT and SIFT features respectively.

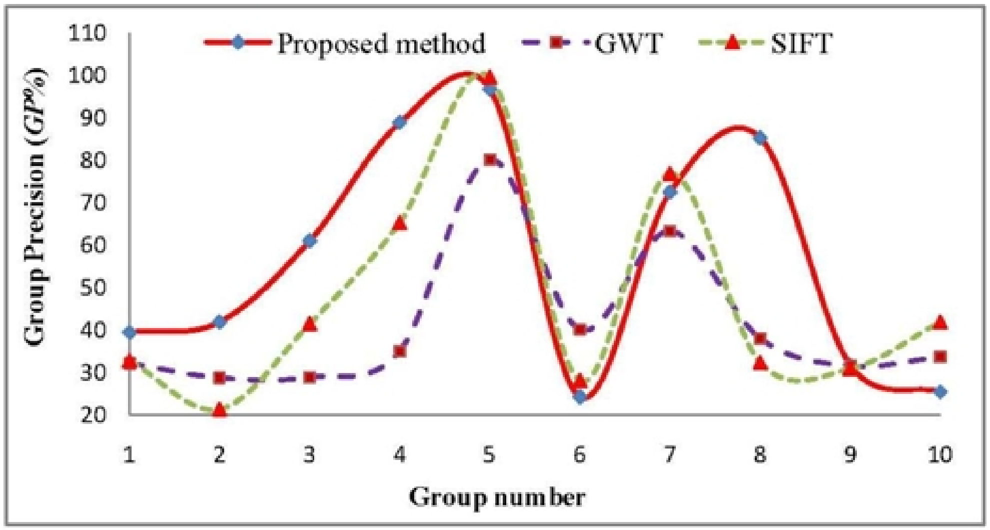


Fig. 2.3: Comparison of group precision

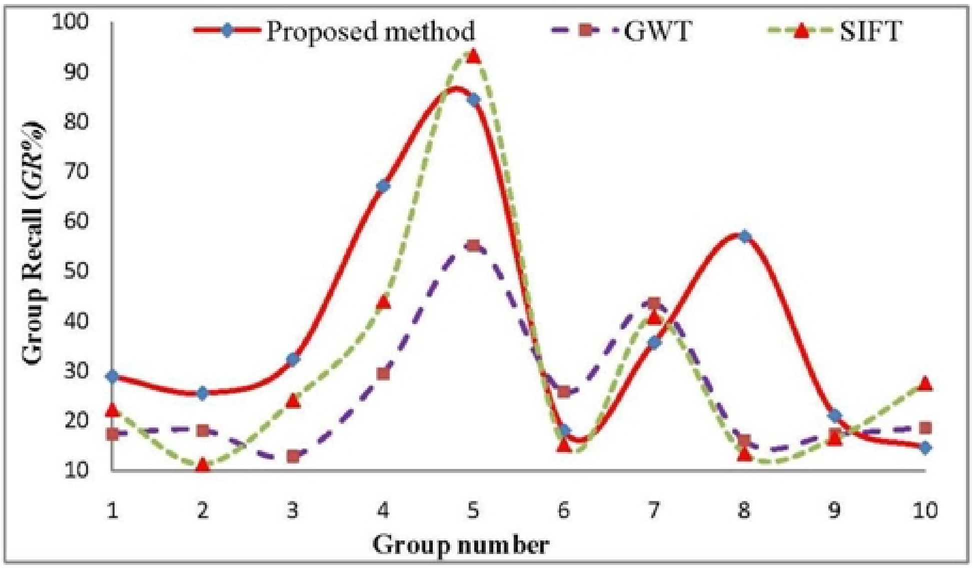


Fig. 2.4: Comparison of group recall

It is evident that *GP%* and *GR%* of proposed method are most of the times better than GWT and SIFT based method. Table 2.2 illustrates the effect of number of retrieved images on *AP%* in case of GWT, SIFT and proposed method (HOG). *AP%* is measured by averaging all the individual group precision values at fixed number of retrieved images. It is verified from Table 2.2 that the proposed method has improved *AP%* in all instances of retrieved images. For example in the case of 10 and 100 retrieved images *AP%* of proposed method is 56.75 and 38.45 respectively, this is far better than *AP%* of GWT based method (41.2 and 25.41 respectively) and SIFT based method (47.08 and 30.79 respectively).

Table 2.2

Average precision (%) according to number of retrieved images

Method	*T*									
	10	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
GWT	41.2	35.28	32.8	31.09	29.72	28.76	27.77	26.81	25.96	25.41
SIFT	47.08	41.80	39.21	37.27	35.84	34.51	33.49	32.52	31.72	30.79
HOG	56.75	51.85	49.13	47.03	45.31	43.72	42.27	40.93	39.67	38.45

For better illustration a plot for *AP%* as a function of number of retrieved images *(T= 10, 20, ... 100)* is drawn in Fig. 2.5. It is apparent from Fig. 2.5 that *AP%* of proposed method is always superior to GWT and SIFT based method.

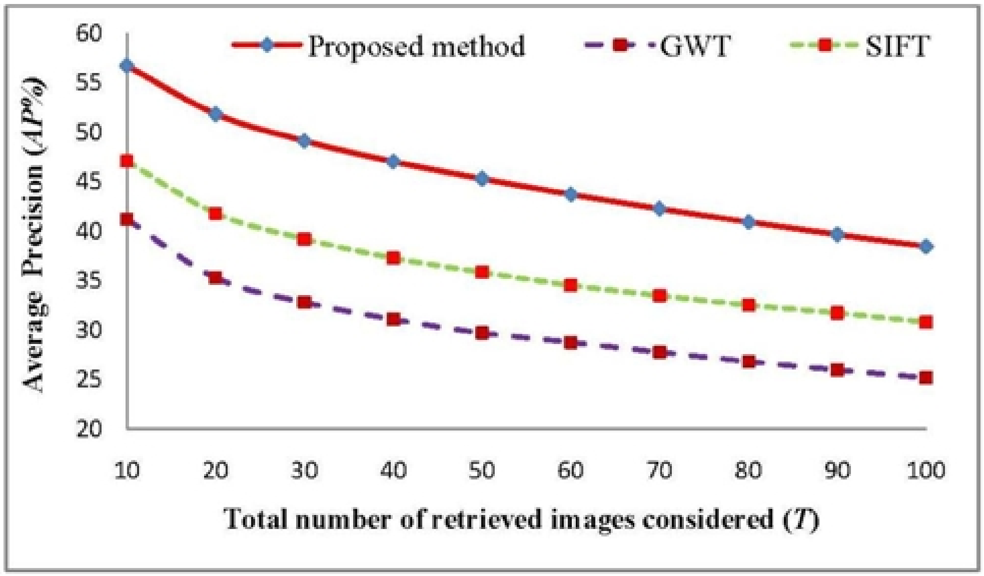


Fig. 2.5: Average precision (%) according to the number of retrieved images

Further *ARR*% is also computed on total image database for performance comparison. Fig. 2.6 shows a graphical representation for *ARR%* as a function of different numbers of retrieved images considered *(T= 10, 20... 100).* Fig. 2.6 proves that for all values of number of retrieved images, *ARR%* of proposed method is consistently better than that of GWT and SIFT based method. For top 10 and 100 matches considered *ARR%* of proposed method (5.67 and 38.45) is higher as compared to GWT based method (4.12 and 25.41) and SIFT based method (4.7 and 30.79) respectively. This verifies the improvement in retrieval efficiency by proposed method in contrast to GWT and SIFT based method on the basis of *ARR%* also.

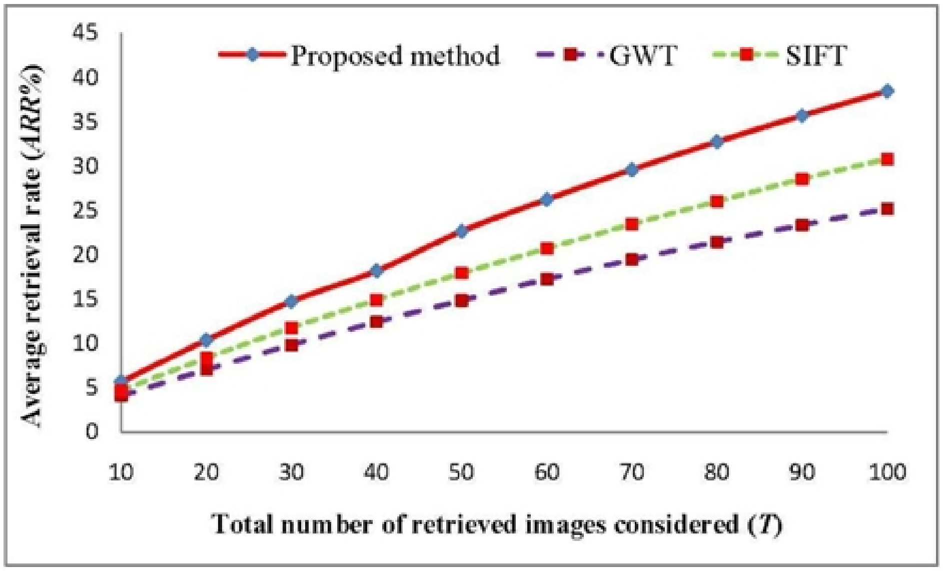


Fig. 2.6: Average retrieval rate (%) according to the number of retrieved images

Fig. 2.7 depicts retrieval results obtained by proposed method for four query images from Bus, Dinosaur, Flower and Horse groups of Corel 1000 image database (Appendix A) [167]; (a) 330; (b) 445; (c) 658; (d) 725. In each case, top left image is the query image and retrieval results for total ten numbers of retrieved images are shown. It is illustrated by Fig. 2.7 that in all cases proposed method is retrieving all relevant images. Thus, efficacy of proposed method is also established by image retrieval result. The main reason for the success of HOG feature is its strength to seize fine intensity variations as well as direction of intensity variations, by operating on local regions. Both intensity variations and direction of intensity variations in an image are highly capable of discriminating images.

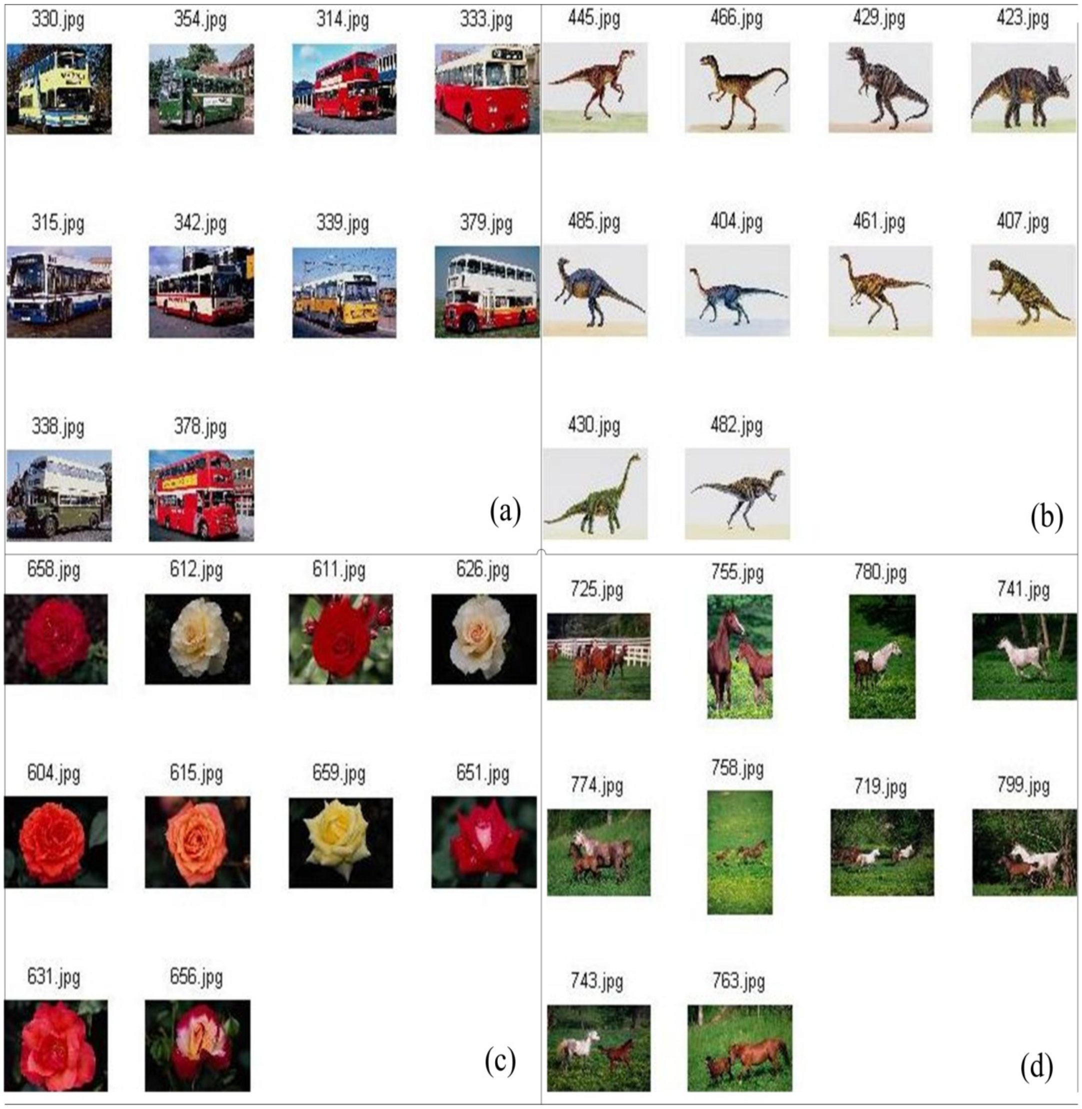


Fig. 2.7: Image retrieval results of proposed method

2.3.2 Corel 2450 Database (DB2)

In order to establish the efficacy of the developed algorithm, it is tested on a larger Corel 2450 (Appendix A) [167] database. Table 2.3 comprises the retrieval results of proposed HOG, GWT and SIFT based methods on DB2 database for various numbers of retrieved images (*T*). It is observed from Table 2.3 that *AP%* of HOG is far better than GWT and SIFT for different number of retrieved images. For top 10 and 100 retrieved images *AP%* of HOG (34.20 and 18.08) outperforms GWT (21.65 and 8.58) and SIFT (25.62 and 12.56) based methods respectively.

Table 2.3

Average precision (%) comparison for database DB2

Method	***T***									
	10	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
GWT	21.65	16.42	13.12	12.76	11.51	10.34	9.51	9.27	8.91	8.58
SIFT	25.62	20.04	17.66	16.26	15.27	14.48	13.83	13.33	12.90	12.56
HOG	34.20	28.61	25.90	24.01	22.61	21.45	20.51	19.65	18.82	18.08

Fig 2.8 comprises the graphical representation of average precision. It is verified that the *AP%* of proposed method is always superior to GWT and SIFT based retrieval methods.

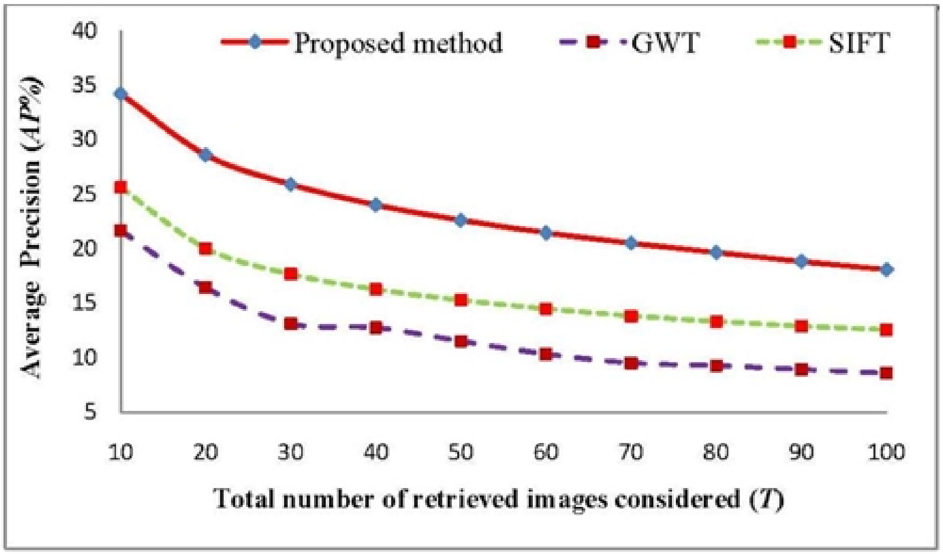


Fig. 2.8: Average precision (%) according to the number of retrieved images

2.4 SUMMARY

In this chapter, a new framework for CBIR is designed by using HOG feature descriptor. Since HOG descriptor generates a large size of feature vector so indexing of HOG feature is performed by vocabulary tree. Vocabulary tree provides an easy way to tackle the problem of large feature database efficiently by reducing the number of comparisons in similarity measurement. Proposed method, GWT and SIFT based methods are compared on the basis of their retrieval performance. Major factors considered for performance evaluation are *AP*, *AR* and *ARR*. The ability of HOG to grab hold of the finer details present in the image makes its suitable for image representation and retrieval. It is evident from the experiments performed on Corel 1000 as well as Corel 2450 standard image databases that the proposed method is found superior in the retrieval performance. *AP* and *AR* of the proposed method are (56.75% and 38.45%, respectively) apparently better than GWT based method (41.20% and 25.41%, respectively) and SIFT (47.08% and 30.79% respectively). In the same manner *ARR%* of proposed method is also far better than GWT based method for different number of retrieved images. It is also verified that on Corel 2450 database, *AP%* of HOG is 34.20 is far better than GWT (21.65) and SIFT (25.62) for 10 retrieved images.

Since, the local features are extracted by operating on local regions or keypoints. Hence, feature extraction as well as similarity measurement is a complicated process. Also, due to partitioning of images some information is lost. Therefore, still there is need to investigate some method which can further improve the performance. To this end in the subsequent chapters the emphasis is towards the implementation of global features.

Log Gabor and Binary Wavelet Transform Based Features

Chapter 3

In the previous chapter HOG based image retrieval algorithm was presented. Although the results were quite promising but still there can be improvement. To that end in this chapter transform based algorithms are developed and investigated.

As natural images do have information at many scales and angles. Thus information available at these scales and angles is important to find out the exact image in various situations. Therefore, it is a good idea to extract image texture information by multiresolution and multiorientation analysis. Thus, in the work reported in this chapter such analysis is performed by log Gabor filter (LGF) and binary wavelet transforms (BWT), and new features have been proposed.

In the first part of this chapter LGF is proposed. LGF basically consists in a logarithmic transformation of the Gabor filters and is capable of avoiding various limitations of Gabor wavelet transform (GWT) such as its extended tails are capable of encoding natural images more efficiently. LGF is capable of representing the higher frequency components better than Gabor functions. It also eliminates the annoying DC component present in Gabor filters. Log Gabor function captures broad spectral information with a compact spatial filter [177]. Thus, all these benefits encouraged applying LGF for the CBIR. Experiments are conducted for comparative analysis of LGF based retrieval system and retrieval system based on GWT feature descriptor. It is verified that the proposed, LGF based retrieval system improves retrieval performance in comparison to GWT based retrieval system on Corel 1000 and Corel 2450 natural image database (Appendix A) [167] with respect to various performance measures.

In the subsequent part of this chapter, another transform domain method, binary wavelet transform based histogram (BWTH) is introduced to extract texture statistics of an image. As simple discrete wavelet transform produces a large dynamic range of subband coefficients hence, quantization of these coefficients is a difficult task. Since BWT conserves intensity levels of the original image in the directional subbands so it is computationally efficient. Owing to this reason BWT is used in this chapter for multiresolution analysis to propose new texture features.

Color of the image plays a significant role in image detection. Hence, the work is further extended in order to facilitate BWTH with color information too. The extended BWTH feature can extract both color as well as texture information. It encouraged the authors to apply it for CBIR. Performance of BWTH based CBIR system is evaluated on Corel 1000 and Corel 2450 standard image databases (Appendix A) [167] in color as well as gray spaces. Results are compared with color histogram (CH) [31], discrete wavelet transform (DWT) [77], auto correlogram (AC) [38], optimal quantized wavelet correlogram (OQWC) [83], motif cooccurrence matrix (MCM) [55] and directional binary wavelet patterns (DBWP) [65] with respect to various performance measures.

BWTH utilizes pixel wise coefficient information but does not consider the spatial relationship among transformed coefficients. Since, this information is very useful to detect texture of an image hence, by further extension correlogram feature is also integrated into BWTH feature. Experimentations suggest that retrieval performance is significantly improved as compared to BWTH itself and to that of AC [38], OQWC [83], Gabor wavelet correlogram (GWC) [91].

3.1 PROPOSED METHOD 1 (PM1)

In this chapter, firstly log Gabor filter (LGF) is proposed for texture feature extraction. Basics of LGF and comparison with GWT are explained in the following section.

3.1.1 Log Gabor Filter (LGF)

Maximum bandwidth of a Gabor filter is limited to approximately one octave and Gabor filters are not optimal if broad spectral information with maximal spatial localization is needed. To avoid these limitations of Gabor filters, Field [95] proposed the log Gabor filters. Log Gabor functions having the extended tails are capable of encoding natural images more efficiently as compared to Gabor functions. Since, log Gabor functions better represent the higher frequency components than Gabor functions, which over represent low frequency components and under represent the high frequency components. It is more apparent for bandwidths more than 1 octave. Log Gabor filters can be constructed with arbitrary bandwidth and this bandwidth can be optimized to produce a filter with minimal spatial extent [176]. Zhitao *et al.* [177] observed that the spatial width of a 3-octave log Gabor function is approximately same as that of a 1-octave Gabor function, it clearly illustrates the capability of log Gabor function to capture broad spectral information with a compact spatial filter. Log Gabor filter basically consists in a logarithmic transformation of the Gabor filters and eliminates the annoying DC component allocated in Gabor filters. Fig. 3.1 illustrates comparison between Gabor and log Gabor filter.

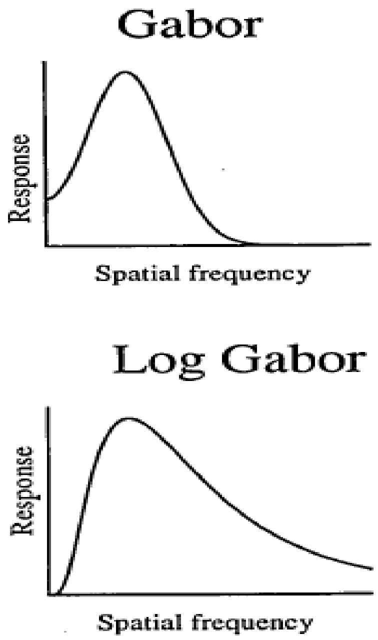


Fig. 3.1: Comparison of Gabor and log Gabor functions in frequency domain [95]

2-D log Gabor filter in frequency domain can be defined by polar coordinate around a central frequency (f_i, θ_i) using Eq. (3.1).

$$\Phi(f,\theta) = R(f) \times A(\theta) \qquad OR$$

$$\Phi(f,\theta) = e^{-\left\{\frac{\left(\log\left(\frac{f}{f_i}\right)\right)^2}{2\left(\log\left(\frac{\sigma_{f_i}}{f_i}\right)\right)^2}\right\}} e^{-\left\{\frac{(\theta-\theta_i)^2}{2\sigma_{\theta_i}^2}\right\}} \tag{3.1}$$

Where, $R(f)$ – Radial component for controlling the frequency band

$A(\theta)$ – Angular component for controlling the orientation of filter

θ_i – Orientation angle of the filter

f_i – Central radial frequency

σ_{fi} , $\sigma_{\theta i}$ – Radial and angular standard deviation respectively

σ_{fi} defines the radial bandwidth B in octaves. $\sigma_{\theta i}$ defines the angular bandwidth in radians. Ratio ${}^{\sigma_{fi}}\!/\!{}_{\sigma_{\theta i}}$ is kept constant for varying f_i in order to make shape of the filter constant. ${}^{\sigma_{fi}}\!/\!{}_{f_i}$ is set to 0.55 to get the filter bandwidth with two octaves. Values of other parameters used in the implementation can be found at [178].

Number of scales and orientations of filter depend upon user’s interest. Fig. 3.2 and Fig. 3.3 show frequency representation of log Gabor filter and Gabor filter respectively. In the presented work filters with three scales and four orientations are implemented to extract texture features from natural images as these are the optimum number of filters for image retrieval.

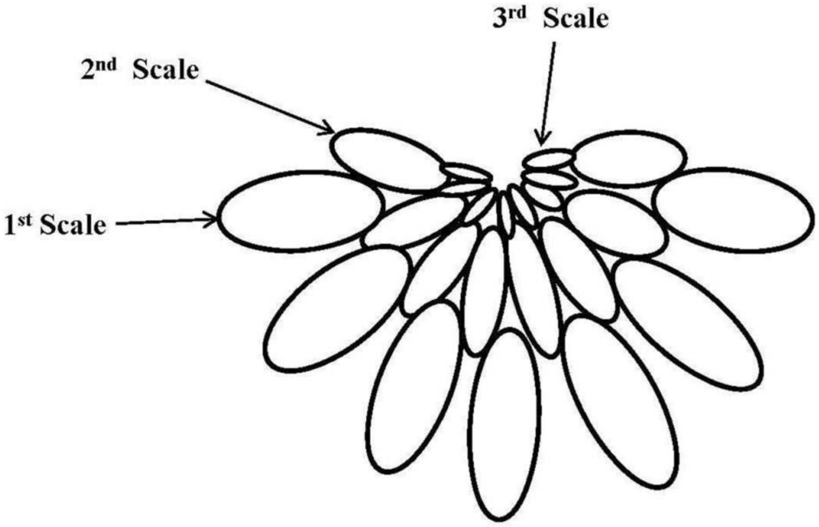


Fig. 3.2: Frequency representation of 2-D log Gabor filters [179]

This leads to the generation of twelve filters $\Phi_{mn} = \{\Phi_{00},\ \Phi_{01},\ldots,\Phi_{23}\}$ representing different scales and orientations (Where, scale $m = 0,1,2$ and orientation $n = 0,1,2,3$). The image filtering is performed in the frequency domain in order to compute LGF response. Image is converted into frequency domain and multiplied by the log Gabor filter. It results in twelve different types of LGF responses i.e. $G_{mn} = \{G_{00}, G_{01},....,G_{23}\}$ for each image. These LGF responses are then transformed back to the spatial domain via *2D* inverse FFT using Eq. (3.2).

$$G_{mn}(x,y) = F^{-1}\{I \times \Phi_{mn}\} \tag{3.2}$$

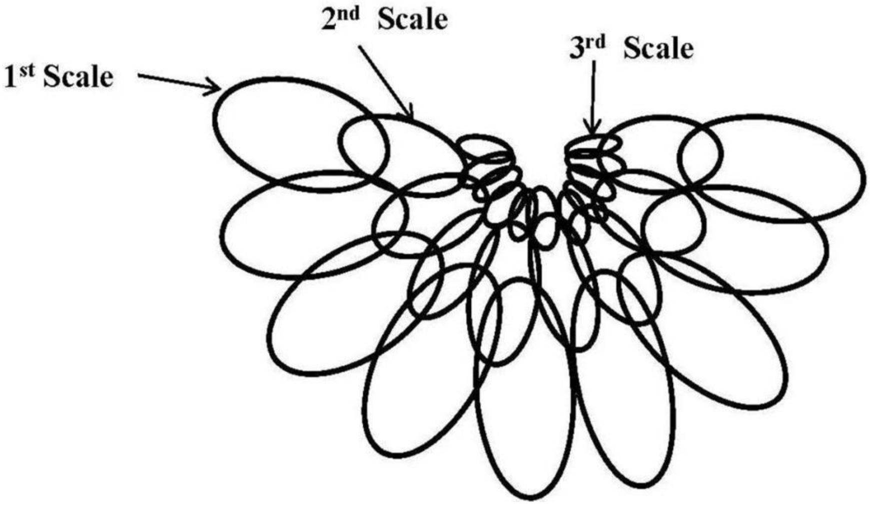


Fig. 3.3: Frequency representation of 2-D Gabor filters [96]

Now, LGF Features are calculated through the filtered images in terms of mean and standard deviation as explained in section 3.1.2.

3.1.2 Feature extraction

After filtering operation, the aim is to calculate features from these filtered images. Same feature descriptors are extracted from LGF and GWT filter responses for comparison. Texture features of an image are computed by mean μ_{mn} and standard deviation σ_{mn} through Eq. (3.3) and (3.4) respectively on transformed coefficients.

$$\mu_{mn} = \frac{1}{XY}\sum_{x=1}^{X}\sum_{y=1}^{Y} G_{mn}(x,y) \tag{3.3}$$

$$\sigma_{mn} = \sqrt{\frac{1}{\left(XY-1\right)^2}\sum_{x}^{X}\sum_{y}^{Y}\left(G_{mn}(x,y)-\mu_{mn}\right)^2} \tag{3.4}$$

Where, $m = 0,1,2$ and $n = 0,1,2,3$ represent scales and orientations respectively. Final feature vector is created by using μ_{mn} and σ_{mn} as the feature components. For three scales and four orientations, the final feature vector is constructed by Eq. (3.5).

$$\boldsymbol{f_1} = (\mu_{00}, \mu_{01}, \ldots.\mu_{23}) \qquad (3.5)$$
$$\boldsymbol{f_2} = (\sigma_{00}, \sigma_{01}, \ldots..\sigma_{23})$$

3.1.3 Similarity measurement

A query image can be any image from the database. In the experiments all images are considered as query image to evaluate average retrieval performance. All images are processed to compute features based on LGF and GWT. Each feature is normalized by using Gaussian normalization. Then d_1 distance metric given by Eq. (3.6) is used to compute dissimilarity or mismatch value between pairs of query image and database images [180].

$$D(Q,T_j) = A\sum_{i=1}^{\lambda}\left|\frac{Q_{1i}-T_{1ji}}{1+Q_{1i}+T_{1ji}}\right| + B\sum_{i=1}^{\lambda}\left|\frac{Q_{2i}-T_{2ji}}{1+Q_{2i}+T_{2ji}}\right| \qquad (3.6)$$

Q_{1i} , Q_{2i} are i^{th} feature vector components of ***f₁*** and ***f₂*** features of the query image respectively. T_{1ji}, T_{2ji} are i^{th} feature vector components of ***f₁*** and ***f₂*** of $j^{th}\left(j=1,....,DB\right)$ database image respectively. λ is the length of feature vector, *DB* is number of images in the database .*A (=0.65)* and *B (=0.35)* are weights for ***f₁*** and ***f₂*** features obtained by experimentation.

Database image having small *D* value is more relevant to the query image. Likewise, required numbers of images relevant to the query image are retrieved.

3.1.4 Experimental results and discussions

Experiments of image retrieval are conducted on standard Corel 1000 and Corel 2450 image databases (Appendix A) [167] and performance of proposed feature is evaluated with respect to various performance measures like average precision (*AP %),* average recall (*AR %*) and average retrieval rate (*ARR %*). These performance measures are computed by Eq. (3.9), (3.12) and (3.13) respectively. Results of proposed method, PM1 are compared with GWT based method on the basis of these performance measures. For GWT based feature extraction images are decomposed into three scales and four directions through GWT and texture features are constructed by computing mean and standard deviation on each GWT transformed subbands. For i^{th} image precision is given by Eq. (3.7).

$$Precision\,(P_i) = \frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ retrieved\,images\,considered\,(T)} \tag{3.7}$$

$$Group\ precision\,(GP) = \frac{1}{N}\sum_{i=1}^{N} P_i \tag{3.8}$$

$$Average\ precision\,(AP) = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} GP_i \tag{3.9}$$

Similarly performance measures are computed in the case of recall by following Eq.

$$Recall\,(R_i) = \frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ relevant\,images\,in\,database(N)} \tag{3.10}$$

$$Group\,Recall\,(GR) = \frac{1}{N}\sum_{i=1}^{N} R_i \tag{3.11}$$

$$Average\,Recall\,(AR) = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} GR_i \tag{3.12}$$

$$Average\,Retrieval\,Rate\,(ARR_T) = \frac{1}{DB}\sum_{j=1}^{DB} R_j \Bigg|_{T \le 100} \tag{3.13}$$

Where, N is number of relevant images in the database, Γ is the number of groups and DB is the number of database images. The retrieval result is not a single image but it is a list of images depending upon relevancy. $T(e.g.10,20,.....,100)$ gives the total number of retrieved images considered. Value of T is selected by user.

3.1.4.1 Corel 1000 Database (DB1)

First experiment is performed on Corel 1000 database. Table 3.1 shows the *GP%* and *GR%* comparison for each group of Corel 1000 image database (Appendix A) [167] in case of LGF and GWT method.

For performance evaluation of retrieval method generally average of group precision and group recall are being considered and *AP% / AR%* are calculated as the average of all individual group precision / group recall values at fixed number of retrieved images. From Table 3.1, it is clear that the LGF has improved retrieval performance in terms of *AP%* and *AR%* (55.46 and 32.03 respectively) as compared with GWT (50.61 and 31.63 respectively).

Table 3.1

Precision and recall for LGF and GWT

Group Number	Group Name	LGF		GWT	
		GP% (T=10)	*GR% (T=100)*	*GP% (T=10)*	*GR% (T=100)*
1	**Africans**	44.90	28.25	39.8	22.72
2	**Beaches**	48.10	31.48	35.2	21.61
3	**Buildings**	51.90	26.09	38.5	20.3
4	**Buses**	84.00	56.48	70.8	47.34
5	**Dinosaurs**	59.40	30.14	81.0	49.86
6	**Elephants**	44.00	24.43	46.9	28.01
7	**Flowers**	78.80	55.65	77.5	60.14
8	**Horses**	56.60	22.85	46.1	24.55
9	**Scenery**	34.40	19.29	29.7	17.66
10	**Foods**	52.50	25.73	40.6	24.19
Average		**55.46**	**32.03**	**50.61**	**31.63**

Fig. 3.4 illustrates the effect of total number of retrieved images considered (*T*) on *AP%* for both the methods. It is evident through Fig. 3.4 that *AP%* of LGF is always higher than the GWT method for all cases of retrieved images.

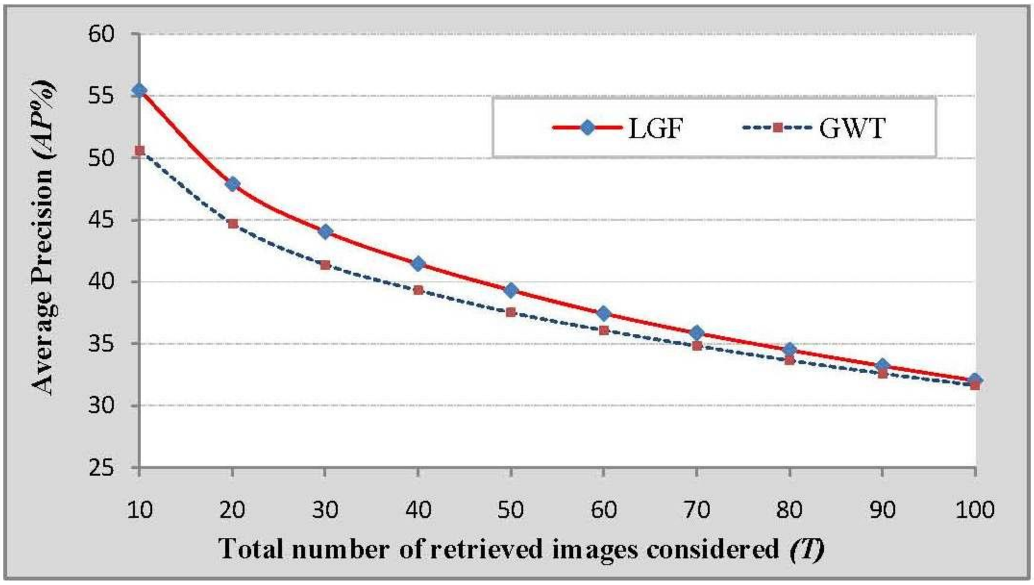


Fig. 3.4: Average precision according to number of retrieved images considered

For more analysis of results in Fig. 3.5, a graph is drawn for *ARR%* values at different number of retrieved images considered (*T*).

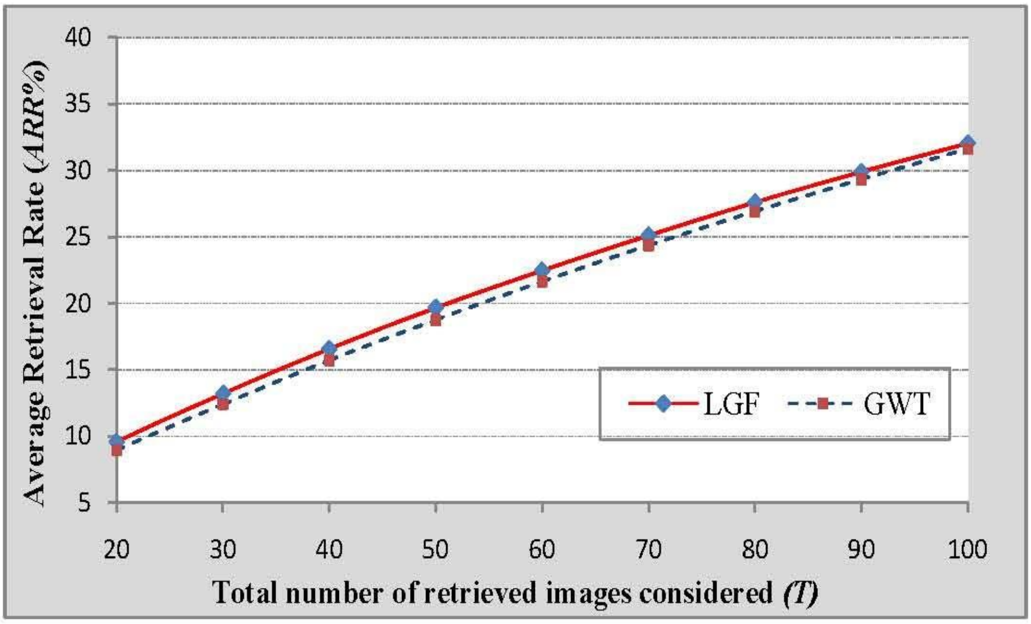


Fig. 3.5: Average retrieval rate according to number of retrieved images considered

Fig. 3.5 illustrates that for all values of retrieved images, *ARR%* of LGF is always better than GWT e.g. in the case of top 100 matches, *ARR%* of proposed method (32.03) is superior to GWT (31.63). It justifies the improvement in retrieval efficiency through proposed method 1. It is observed from Fig. 3.6 that LGF has retrieved relevant images in all cases. Hence, the improvement in retrieval performance also establishes that LGF has the ability to capture more intensity variation present in the image as compared to GWT.

Fig. 3.6 demonstrates four image retrieval results for query image number; (a) 315, (b) 605, (c) 704, (d) 916 obtained through LGF at *T=10* (In all cases top left image is the query image).

3.1.4.2 Corel 2450 Database (DB2)

The second experiment is performed on image database DB2. Fig 3.7 illustrates the *AP%* of LGF and GWT for different number of retrieved image considered (*T*). It is verified that the *AP%* of LGF is far better than GWT. In the case of 10 and 100 retrieved images *AP%* of LGF is 30.79 and 13.1 respectively while for GWT it is 25.95 and 11.12 respectively. Hence, Performance is LGF is found to be superior on Corel 2450 as well.

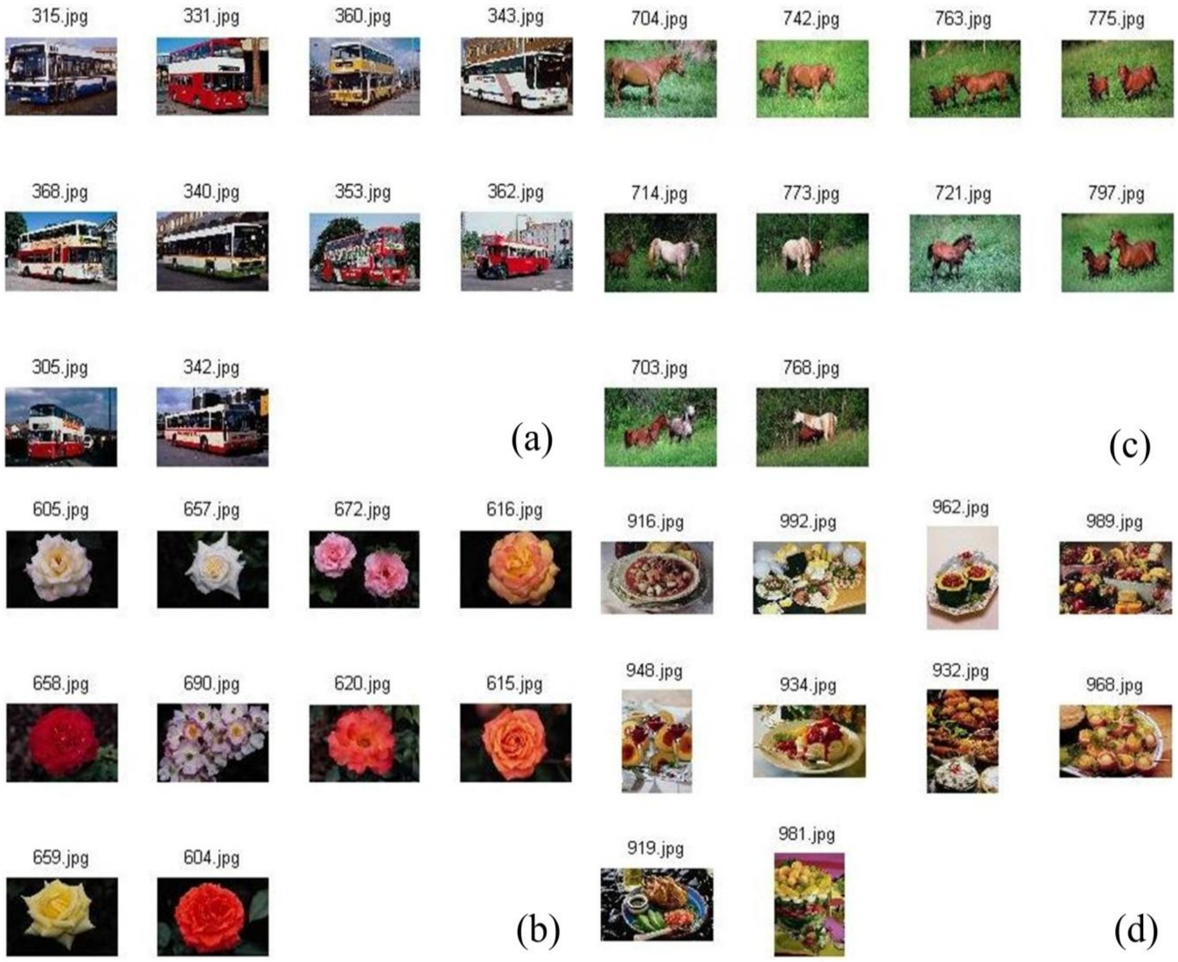


Fig. 3.6: Query retrieval results of proposed method

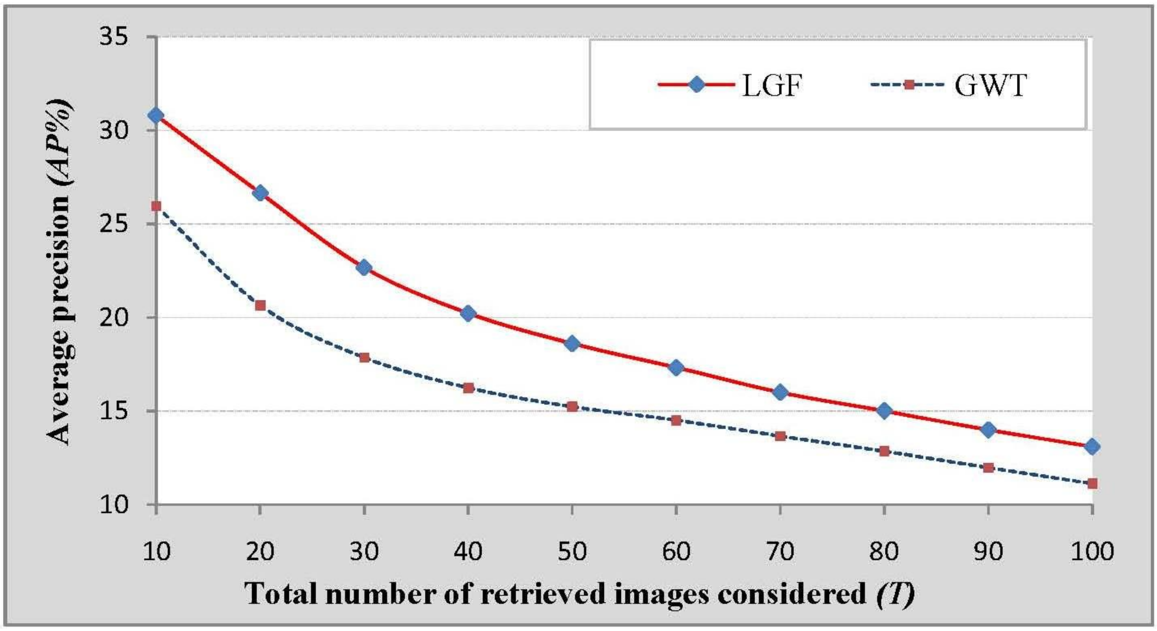


Fig. 3.7: Average precision according to number of retrieved images considered

3.1.5 Summary

This section of chapter has proposed a new framework for image retrieval by employing log Gabor filter for information extraction. It is examined that log Gabor filter performs superior as compared to Gabor wavelet transform based feature descriptors on Corel 1000 and Corel 2450 standard image databases. Experimental results established that the log Gabor filter has significantly improved average precision and average recall (55.46% and 32.03% respectively) as compared to Gabor wavelet transform (50.61% and 31.63% respectively) on Corel 1000 database. In the similar manner retrieval performance of log Gabor filter is far superior to Gabor wavelet transform on Corel 2450 database. In the subsequent section another transform based algorithm is developed. The objective remains the same i.e. to increase the retrieval performance.

3.2 PROPOSED METHOD 2 (PM2)

Simple wavelet transform produces a large dynamic range of subband coefficients thus, quantization of coefficients is difficult. Binary wavelet transform (BWT) preserve the intensity levels and provides same number of intensity levels in the directional subbands as in the input image [89]. Thus, it offers an easy method to work on transformed subbands. It encourages applying BWT for multiresolution analysis.

In this section a new multiresolution feature, binary wavelet transform based histogram (BWTH) is proposed. In the primitive step of proposed algorithm binary wavelet transform (BWT) is applied on each bit of original image to extract information at multiresolution. Then, histogram is calculated on each resultant BWT subband to extract the proposed feature; BWTH. BWT works on simple Boolean operations hence, it is computationally inexpensive. BWTH feature extraction methodologies are tested on RGB as well as on gray scale images. In case of RGB scale every operation is applied individually on every component of image; R, G and B. In case of gray scale, firstly RGB input image is converted into gray scale and then same operations are applied on gray image. It is observed that retrieval performance is further enhanced by including color information. The methodology for BWTH feature extraction is explained in the next sections. The primitive step of BWTH is BWT computation. Thus, BWT in *1–D* and *2–D* are explained in the following section.

3.2.1 Binary wavelet transform (BWT)

Law *et al.* [89] proposed the in-place implementation of BWT. This reduces memory requirement and arithmetic operations. Computation of *1–D* BWT is shown in Fig. 3.8. Odd position values in *1–D* signal give the low pass output while even and odd together with XOR operation gives the high pass or band pass output. Thus, low pass filter output consists only subsampling operation while the high pass filter performs XOR operation between two neighboring samples. Low pass output does not create any change apart from sampling.

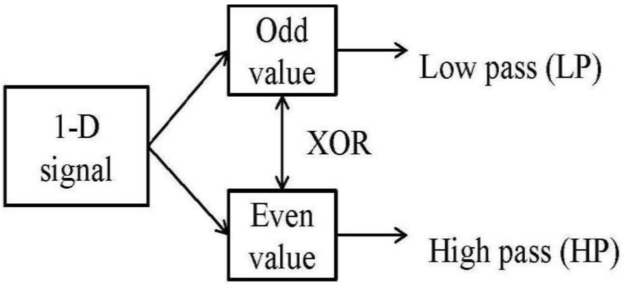


Fig. 3.8: *1–D* BWT

Fig. 3.9 illustrates the *2–D* BWT structure. A separable *2–D* BWT can be computed efficiently in discrete space by applying the associated *1–D* filter bank to each column of the image, and then applying the filter bank to each row of the resultant coefficients.

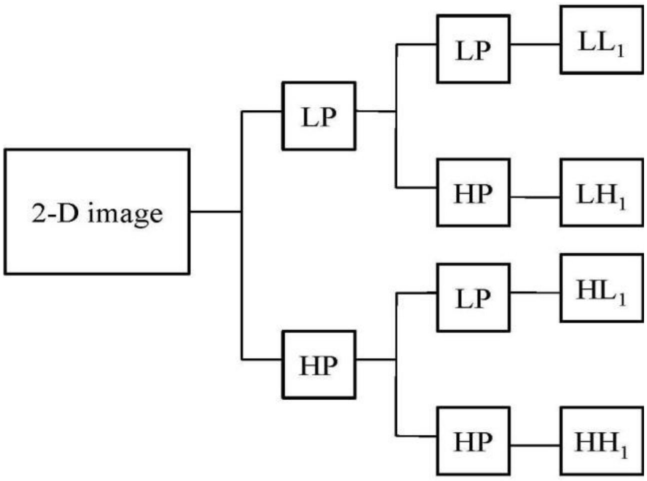


Fig. 3.9: *2–D* BWT

Fig. 3.9 comprises a one level binary wavelet decomposition of a *2–D* image. In the first level of decomposition, one low pass subimage (LL1) and three orientation selective high pass subimages (horizontal LH1, vertical HL1, and diagonal HH1) are created. In second level of decomposition, the low pass subimage (LL1) is further decomposed into one low pass (LL2) and three high pass subimages (LH2, HL2, and HH2). The process is repeated on the low pass subimage to form higher level of binary wavelet decomposition. In other words, BWT decomposes an image into a pyramidal structure of the subimages with various resolutions corresponding to the different scales. Thus, three scale BWT decomposition of an image will create three low pass subbands and nine high pass directional subbands (three; horizontal, vertical and diagonal direction subbands at each scale). On these subband's coefficients, histogram is calculated to extract proposed BWTH feature. Computation of histogram is explained in the following section.

3.2.2 Color Histogram (CH)

Swain *et al.* [31] has proposed a method for color indexing with the help of color histogram (CH). The color histogram represents the count of each color in the image. Histogram is robust to translation, rotation and scale. A color histogram feature *H* for a given image is defined by Eq. (3.14).

$$\boldsymbol{H} = \{H[0], H[1], \ldots.., H[i], \ldots.H[N]\} \tag{3.14}$$

Where, *i* represent the color in color histogram, *H*[i] represent the number of pixels with color *i* in the image and *N* is the number of bins used in *CH*. For comparing the histogram of different sizes, color histogram should be normalized. The normalized color histogram is given as

$$\mathsf{H'} = \frac{\boldsymbol{H}}{p}$$

Where, *p* is the total number of pixels in the image.

3.2.3 Binary wavelet transform based histogram (BWTH)

This section explains about the second proposed method, BWTH for image representation. Algorithm of proposed feature calculation in the case of RGB and gray image is described as follows:

Algorithm:

1. Input RGB/Gray image

2. Repeat a to d steps for *i*; *(i=1 to 3)* in case of RGB image and *(i=1)* in case of gray image

   a. Convert i^{th} component image into 8-bit binary image

   b. On each bit image individually apply BWT upto 3 scales and extract multiresolution information

   c. Convert 8 bit subbands into grayscale back

   d. Now, for each subband calculate histogram feature

3. Histograms of each subband together construct the final BWTH feature vector for the input image

Fig. 3.10 illustrates the process of BWTH feature vector calculation in the case of RGB image. All subbands are quantized into 35 bins of varying width to calculate histogram.

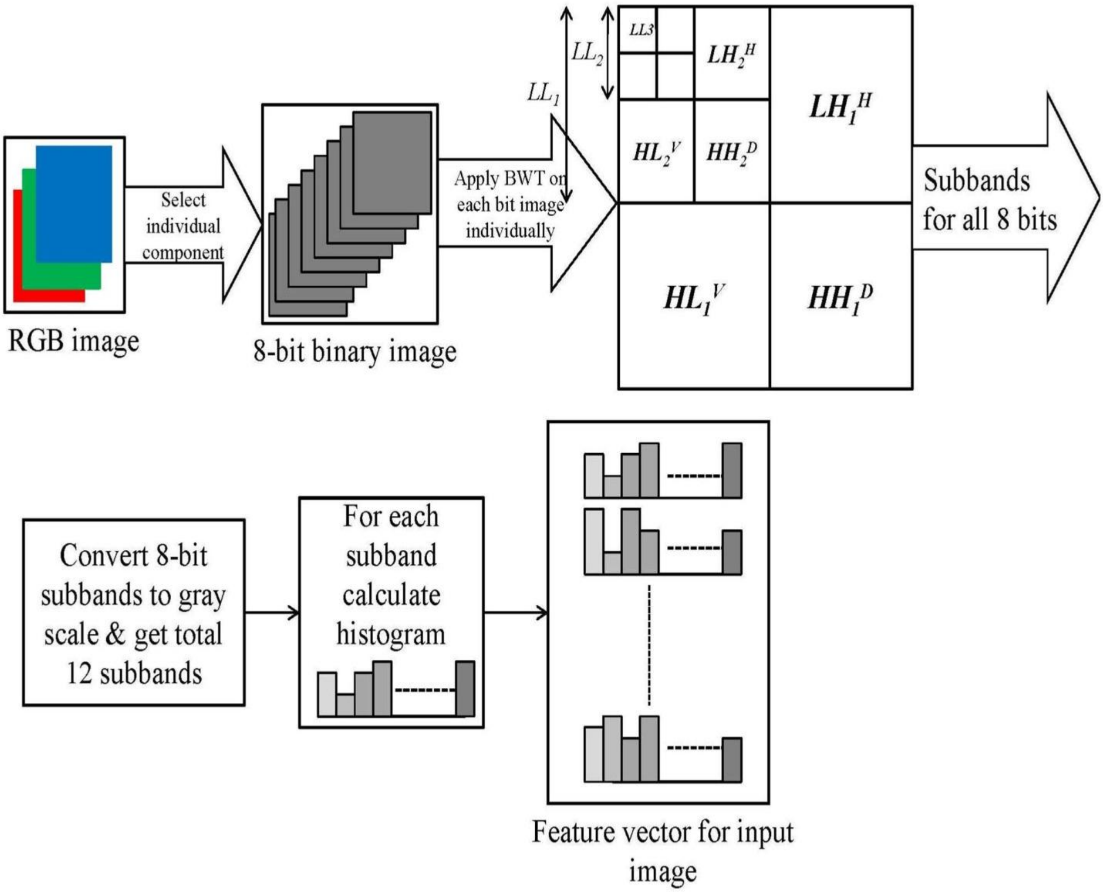

RGB image
Select individual component
8-bit binary image
Apply BWT on each bit image individually
LL_1
LL_2
LL3
LH_2^H
HL_2^V
HH_2^D
LH_1^H
HL_1^V
HH_1^D
Subbands for all 8 bits
Convert 8-bit subbands to gray scale & get total 12 subbands
For each subband calculate histogram
Feature vector for input image

Fig. 3.10: BWTH feature vector construction for RGB image

3.2.4 Similarity measurement

To find out the similarity among query and database images, a query image is selected from the image database. Query image and database images are processed to compute features. In the experiment all images are considered as a query to evaluate average retrieval performance. Distance measure given by Eq. (3.15) is used to compute the mismatch between query image and database image features.

$$D(Q,T) = \sum_{i=1}^{\lambda} \frac{|Q_i - T_i|^2}{|1 + Q_i + T_i|^2} \tag{3.15}$$

Where, Q_i and T_i are i^{th} feature components of the query and database image respectively. λ is the feature vector length. Database image having smallest *D(Q, T)* value is the most relevant to query image. After calculating similarity values on total database images, images relevant to the query image are retrieved.

3.2.5 BWTH based retrieval system

The architecture of CBIR system with proposed feature; BWTH is shown in Fig. 3.11 and the algorithm of respective CBIR system is given as follows:

Algorithm:

1. Input query image (RGB/Gray)
2. Calculate BWTH features for query image and all database images
3. Apply similarity measurement to calculate is match among query and database image features
4. Database images getting less mismatch values will be considered as more relevant to the query image
5. Retrieval result is a list of relevant images present in the database

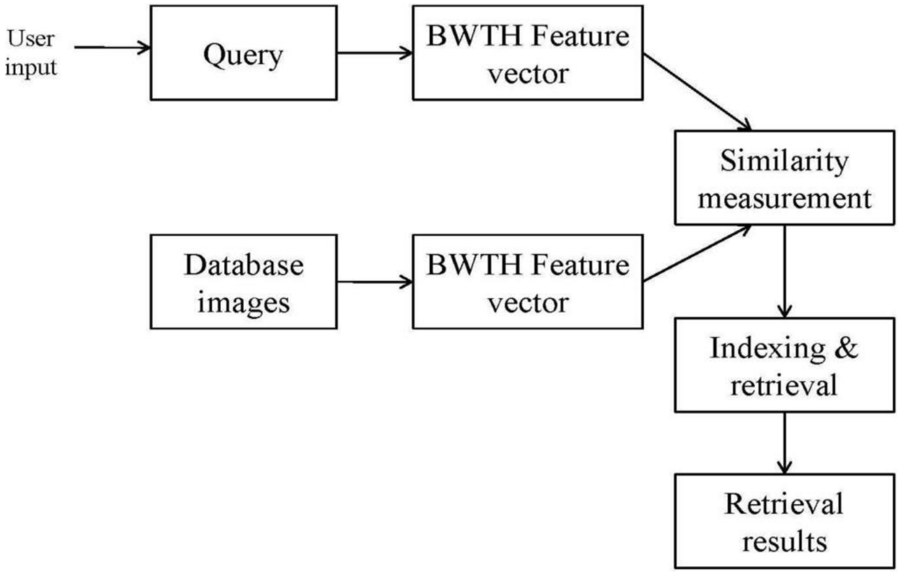


Fig. 3.11: Proposed method 2 (BWTH) based image retrieval system

3.2.6 Experimental results and discussions

All the experiments are performed on the two standard image databases Corel 1000 (DB1) and Corel 2450 (DB2) (Appendix A) [167]. For performance evaluation various performance measures like *AP%* and *AR%* and *ARR%* are calculated by using Eq. (3.9), (3.12) and (3.13) respectively. The abbreviations for extracted features are given below:

BWTH: Binary Wavelet Transform Based Histogram (Proposed method 2)
DWT: Discrete Wavelet Transform
AC: Auto Correlogram
OQWC: Optimal Quantized Wavelet Correlogram
CH: Color Histogram
MCM: Motif Cooccurrence Matrix
DBWP: Directional Binary Wavelet Patterns

Results of BWTH are compared with DWT [77], AC [38], OQWC [83], CH [31], MCM [55] and DBWP [65]. DWT based features are implemented by applying DWT upto three scales. The mean and variance of the all wavelet subbands are considered as texture features for CBIR. AC is implemented by quantizing the input image intensity values into 32 levels and then auto correlogram is calculated in four directions with unit distance. In OQWC computation, DWT of the input image is computed upto three scales and obtained wavelet coefficients are quantized to a limited number of levels by applying an optimization algorithm. Then, autocorrelogram of vertical and horizontal wavelet coefficients is computed. In experiments CH is implemented with 32 bins. For RGB implementation, DWT, AC and CH are calculated on each color channel. Texture features collected from MCM works on difference in intensities computed by motif patterns and finds cooccurrence of these seven types of patterns. In DBWP the 8-bit grayscale image is divided into eight binary bit-planes and binary wavelet transform (BWT) is computed on each bit-plane. Further the local binary pattern (LBP) features are extracted from the resultant BWT sub-bands. BWTH finds the directional and multiresolution information present in the image and computes the distribution of intensities in these subbands. It permits extraction of features from different scales and orientations in a simple manner. Thus, advantages of wavelet and histogram are combined together which ultimately force BWTH to perform better.

3.2.6.1 Corel 1000 Database (DB1)

First experiment of PM1 is performed on database DB1 (Appendix A) [167]. Results obtained on database DB1 are incorporated in Table 3.2, Table 3.3 and Table 3.4. Table 3.2 illustrates the retrieval performance measures; *AP%* and *AR%* comparison of proposed method 2 (BWTH) with DWT, AC, OQWC and DBWP in gray space (*T*: Total number of retrieved images considered).

Table 3.2

Retrieval performance comparison for database DB1 with gray scale

Group Name	**BWTH**		**DWT**		**AC**		**OQWC**		**DBWP**	
	P% *(T=10)*	**R%** *(T=100)*	**P%** *(T=10)*	**R%** *(T=100)*	**P%** *(T=10)*	**R%** *(T=100)*	**P%** *(T=10)*	**R%** *(T=100)*	**P%** *(T=10)*	**R%** *(T=100)*
Africans	76.1	38.42	39.00	20.02	74.2	49.7	57.7	31.1	66.7	35.06
Beaches	43.1	22.94	37.10	22.77	30	15.68	49.3	28.6	44	26.08
Buildings	45.8	26.39	29.70	19.45	40.4	20.25	50.9	30.5	45.3	26.51
Buses	72.6	47.99	56.40	34.38	39.7	22.69	87.1	64.0	82.6	48.1
Dinosaurs	99.9	94.84	95.60	62.25	100	97.73	74.6	28.8	98.4	85.96
Elephants	65	34.58	48.50	27.83	55.1	31.41	55.7	30.7	57.8	25.69
Flowers	93.9	65.39	90.60	71.17	86.7	47.57	84.3	65.3	91.6	65.79
Horses	74.9	28.68	57.10	30.81	70.9	33.48	78.9	39.9	74.3	29.61
Mountains	30.8	16.43	32.80	18.26	21.9	12.54	47.2	25.1	35.3	20.64
Foods	64.1	30.74	40.20	22.82	57.8	31.07	57.1	36.4	59.5	31.3
Average	**66.62**	**40.64**	**52.70**	**32.98**	**57.67**	**36.21**	**64.3**	**38.0**	**65.55**	**39.47**

It is verified that of *AP* and *AR* of proposed method (66.62% and 40.64%) are significantly improved upon DWT (52.70% and 32.98%), AC (57.67% and 36.21%), OQWC (64.3% and 38.0%) and DBWP (65.55 and 39.47)respectively.

It is evident from Table 3.3 that in RGB space also *AP* and *AR* (73.82% and 44.16%) of BWTH is better in comparison to DWT (60.83% and 38.25%), AC (68.52% and 40.81%), CH (70.85% and 42.16%) and DBWP (71.27 and 42.68). Table 3.4 illustrates the *AP* comparison of proposed method 2, BWTH with MCM in the case of 20 retrieved images considered. It is observed that the BWTH shows higher precision for image database DB1. BWTH achieves magnificent improvement in average precision (66.39%) as compared to MCM (52.64%) on total database.

Table 3.3

Retrieval performance comparison for database DB1 with color scale

Group Name	BWTH		DWT		AC		CH		DBWP	
	P% *(T=10)*	R% *(T=100)*	P% *(T=10)*	R% *(T=100)*	P% *(T=10)*	R% *(T=100)*	P% *(T=10)*	R% *(T=100)*	P% *(T=10)*	R% *(T=100)*
Africans	83.1	41.86	45.70	24.25	85.1	56.06	83.10	49.34	75	37.82
Beaches	52.0	26.78	48.00	26.90	46.1	23.01	52.40	24.64	43.6	27.1
Buildings	59.3	29.91	36.20	21.45	53.7	23.96	54.50	27.35	60.5	31.51
Buses	79.9	54.14	59.00	38.35	46.8	30.28	62.10	41.98	85	55.46
Dinosaurs	99.8	95.03	97.80	70.00	100.0	97.91	100	96.90	98.7	88.41
Elephants	72.1	34.25	58.00	31.32	68.0	35.38	67.10	34.82	60.3	25.35
Flowers	92.8	67.14	91.30	74.75	83.6	45.84	84.30	50.57	92.1	66.37
Horses	84.7	37.23	78.00	44.06	89.5	41.96	90.50	42.60	84.4	38.3
Mountains	38.6	20.17	43.30	23.22	37.1	18.34	38.20	18.98	45.6	23.43
Foods	75.9	35.15	51.00	28.15	75.3	35.35	76.30	34.43	67.5	33.1
Average	**73.82**	**44.16**	**60.83**	**38.25**	**68.52**	**40.81**	**70.85**	**42.16**	**71.27**	**42.68**

Table 3.4

Average precision (%) comparison for database DB1 with color scale (*T=20*)

Group no.	1	2	3	4	5	6	7	8	9	10	Average
BWTH	75.25	44.85	50.70	75.00	99.55	61.15	89.75	73.05	31.20	63.35	66.39
MCM	45.25	39.75	37.35	74.10	91.45	30.40	85.15	56.80	29.25	36.95	52.64

Fig 3.12 shows *AP%* for various number of retrieved images (gray level) of DB1. It is observed that *AP%* of BWTH feature is always superior in comparison to DWT, AC, OQWC and DBWP. For top 10 and 100 matches retrieved, *AP%* of BWTH (66.62% and 40.64%) is substantially improved compared to DWT (52.70% and 32.98%), AC (57.67% and 36.21%), OQWC (64.3% and 38.0%) and DBWP (65.55 and 39.47) respectively. In the similar manner Fig 3.13 shows *AP%* as a function of various number of retrieved images for color images of database DB1.

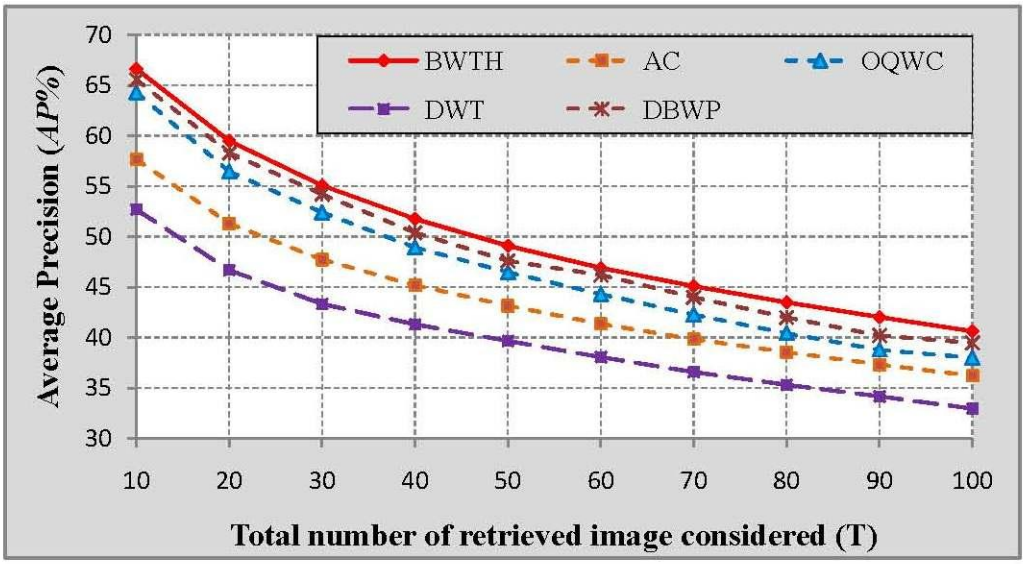


Fig. 3.12: Average precision (*AP%*) for database DB1 with gray scale

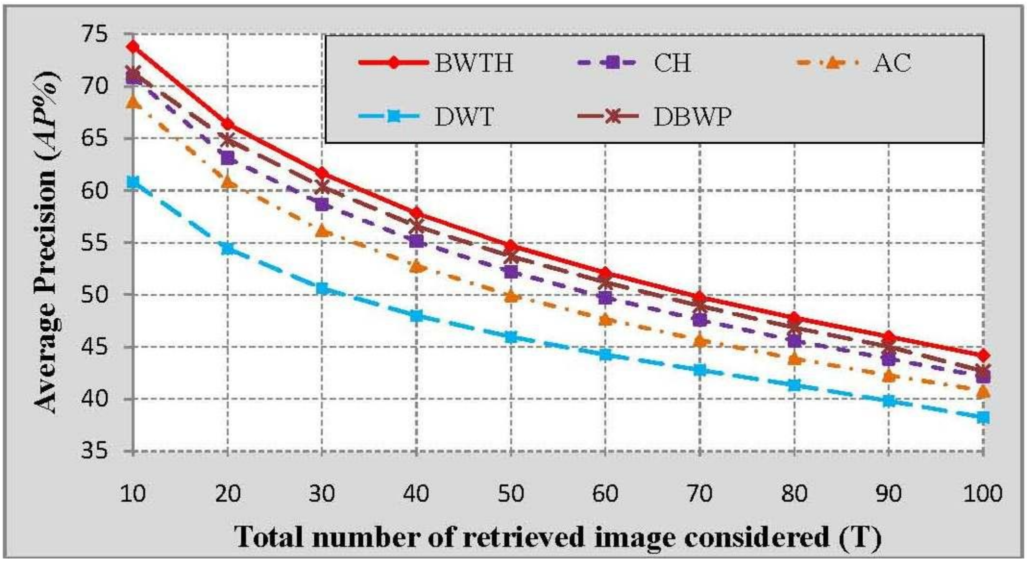


Fig. 3.13: Average precision (*AP%*) for database DB1 with color scale

It is verified that BWTH yields *AP%* of (73.82% and 44.16%) and it is far better than DWT (60.83% and 38.25%), AC (68.52% and 40.81%), CH (70.85% and 42.16%) and DBWP (71.27% and 42.68%) for top 10 and 100 matches retrieved. Further performance is justified by *ARR%* computation on DB1 database. Fig 3.14 and Fig 3.15 show *ARR%* of BWTH and its comparison with other existing methods in gray as well as color space respectively, for database DB1. It is illustrated in Fig 3.14 that BWTH shows significantly better *ARR%* (6.66 and 40.64) as compared to DWT (5.27 and 32.98), AC (5.67 and 36.21), OQWC (6.43 and 38) and DBWP (6.55 and 39.47) for top 10 and 100 retrieved images respectively.

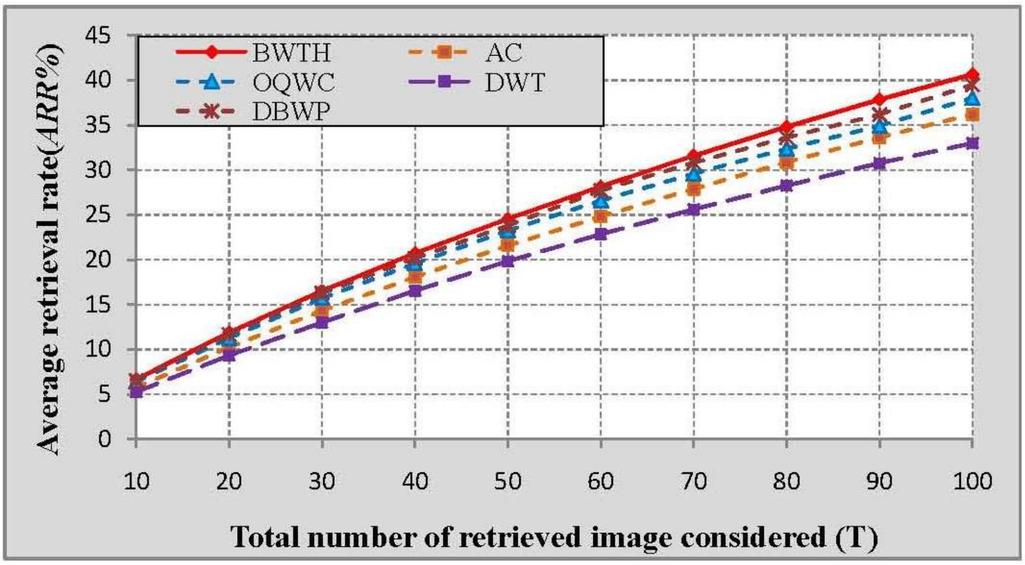


Fig. 3.14: Average retrieval rate (*ARR%*) for database DB1 with gray scale

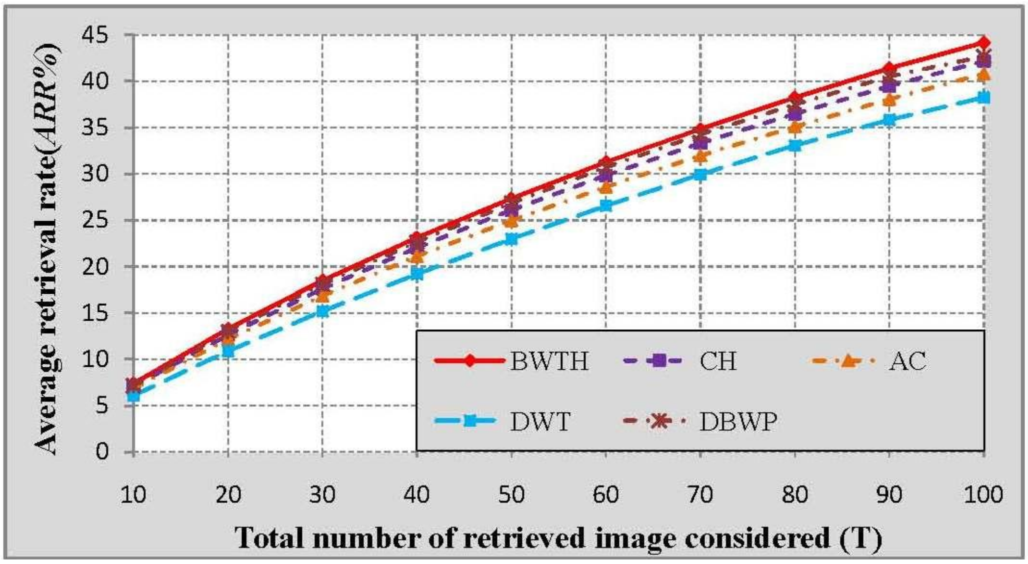


Fig. 3.15: Average retrieval rate (*ARR%*) for database DB1 with color scale

Similarly, Fig 3.15 also justifies improvement in retrieval performance by BWTH. It is analyzed that *ARR%* for BWTH (7.38 and 44.16) is far better than DWT (6.08 and 38.25), AC (6.85 and 40.81), CH (7.08 and 42.16) and DBWP (7.12 and 42.68).

3.2.6.2 Corel 2450 Database (DB2)

Next experiment, on a larger database, is conducted on image database DB2. Table 3.5 comprises the results of BWTH, DWT, AC, OQWC and DBWP in gray scale for DB2 database. It is observed from Table 3.5 that *AP%* of BWTH is far better than other existing methods for different number of retrieved image considered (*T*). For top 10 and 100 retrieved images *AP%* of BWTH (56.07 and 36.68) outperform DWT (42.38 and 27.96), AC (51.60 and 33.34), OQWC (54.55 and 36.55) and DBWP (55.7 and 35.7) respectively.

Table 3.5

Average precision (%) comparison for database DB2 with gray scale

T	10	20	30	40	50	60	70	80	90	100
BWTH	56.07	49.12	45.16	42.27	40.12	40.02	39.50	39.25	37.96	36.68
DWT	42.38	36.00	32.96	31.02	29.58	29.74	29.46	29.46	28.68	27.96
AC	51.60	44.20	40.55	37.87	35.89	35.82	35.42	35.33	34.35	33.34
OQWC	54.55	47.96	44.42	41.58	39.49	39.53	39.04	38.87	37.64	36.55
DBWP	55.7	48.4	44.3	41.5	39.1	39.1	38.5	38.2	36.9	35.7

Table 3.6 illustrates comparison between BWTH and DWT, CH, AC and DBWP on database DB2 for various numbers of retrieved images (*T)* in color scale.

Table 3.6

Average precision (%) comparison for database DB2 with color scale

T	10	20	30	40	50	60	70	80	90	100
BWTH	64.78	56.97	52.25	48.64	45.92	45.40	44.56	44.13	42.50	40.98
DWT	47.44	40.43	36.98	34.73	33.05	33.13	32.72	32.59	31.58	30.64
CH	58.13	49.88	45.48	42.41	39.99	39.71	39.02	38.78	37.51	36.31
AC	59.36	51.02	46.44	43.28	40.77	40.50	39.81	39.54	38.15	36.94
DBWP	61.96	54.58	50.12	46.78	44.23	43.87	43.1	42.83	41.27	39.84

It is apparent from Table 3.6 that *AP%* performance of BWTH (64.78 and 40.98) is much better than other existing methods like DWT (47.44 and 30.64), CH (58.13 and 36.31), AC (59.36 and 36.94) and DBWP (61.96 and 39.84) for top 10 and 100 retrieved images respectively.

Fig 3.16 and Fig 3.17 describe the *AP%* comparison among BWTH and various existing methods in gray and color scale respectively, for database DB2. It is verified from Fig 3.16 that *AP%* of BWTH is always higher than DWT, AC, OQWC and DBWP.

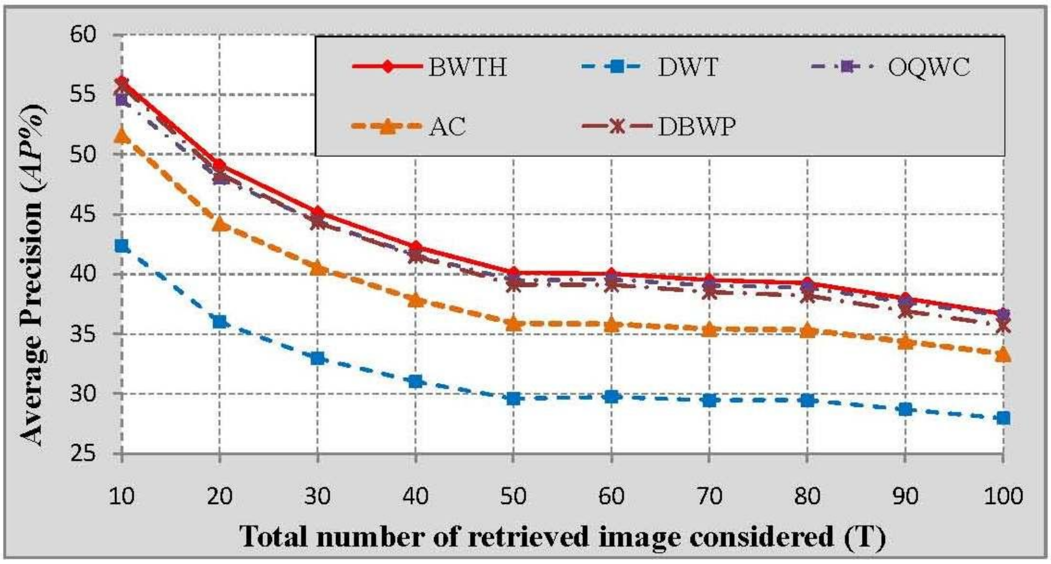


Fig. 3.16: Average precision (%) for database DB2 with gray scale

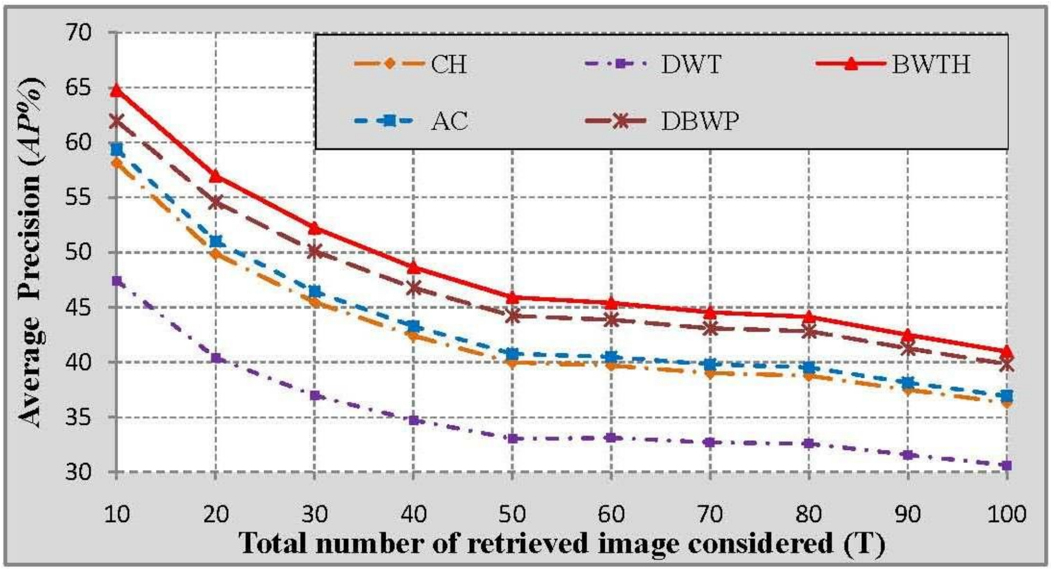


Fig. 3.17: Average precision (%) for database DB2 with color scale

Similarly, from Fig 3.17 it is evident that *AP%* of BWTH always shows better performance compared to DWT, AC, CH and DBWP in the case of different number of retrieved images.

3.2.7 Summary

In this section a novel visual feature BWTH is proposed. It utilizes the characteristics of color as well as texture feature. BWTH extract directional edge information by applying BWT on each bit-plane image and its features are computed by histogram on generated subbands. The experimental results demonstrate that the proposed BWTH feature is capable of extracting useful information for representing gray as well as color scale images. Results on database DB1 and DB2 verify that the proposed method 2 (BWTH) outperform to the existing methods in gray as well as color space. In case of gray level DB1, *AP%* and *AR%* of BWTH (66.62 and 40.64) are better than DWT (52.70 and 32.98), AC (57.67 and 36.21), OQWC (64.3 and 38.0) and DBWP (65.55 and 39.47). The results analysis of DB1 database in RGB space also illustrates better *AP%* and *AR%* of BWTH (73.82 and 44.16) as compared to DWT (60.83 and 38.25), AC (68.52 and 40.81), CH (70.85 and 42.16) and DBWP (71.27 and 42.68). It is also verified that in case of database DB2, *AP%* of proposed method (56.07 and 36.68) for top 10 and 100 retrieved images are significantly better as compared to DWT (42.38 and 27.96), AC (51.60 and 33.34), OQWC (54.55 and 36.55) and DBWP (55.7 and 35.7) respectively in gray scale and in color scale also BWTH has improved *AP%* (64.78 and 40.98) as compared to DWT (47.44 and 30.64), CH (58.13 and 36.31), AC (59.36 and 36.94) and DBWP (61.96 and 39.84) respectively. Similarly *ARR%* of BWTH is substantially better than other existing methods.

3.3 PROPOSED METHOD 3 (PM3)

Histogram gives information about pixel intensities while correlogram comprises interrelation among the pixel intensities. Relationship among subbands coefficients exhibits good approximation of image texture. So, by combining these two techniques more information can be extracted to represent images. Further a new feature extraction technique is proposed by integrating the properties of histogram and correlogram of binary wavelet coefficients. Methodology for integrated features of binary wavelet transform (PM3) is explained below:

3.3.1 Integrated features of Binary wavelet transform (IFBWT)

PM3 performs the task of image retrieval by using the histogram and auto-correlogram features extracted from the binary wavelet subbands. So, the primitive step of PM3 feature extraction is BWTH calculation. BWTH computation is already explained in section 3.2.

Further, in PM3 implementation BWT high pass subbands coefficients are quantized into 9 levels and auto correlogram feature is calculated in their respective subband direction. Correlogram characterizes not only the distributions of pixel intensities, but also the spatial correlation between pairs of intensities. A correlogram can be interpreted as a table indexed by intensity pairs, where the k^{th} entry for *(i, j)* specifies the probability of finding a pixel of intensity *j* at a distance *k=1* from a pixel of intensity *i* in the image. Let *I* represents the entire set of image pixels and $I_{c(i)}$ represent the set of pixels whose intensities are *c(i)*. Then, the correlogram feature is defined by Eq. (3.16).

$$\gamma_{i,j}^{(k)} = \underset{p_1 \in I_{c(i)}, p_2 \in I}{Pr} \left[p_2 \in I_{c(j)} \| p_1 - p_2 | = k \right] \qquad (3.16)$$

$$i = j \ for\, autocorrelogram$$

The combination of histogram and auto correlogram work well because it contains distinguish properties to represent any image.

Fig. 3.18 illustrates the computational process of proposed method 3 and step wise algorithm of feature calculation is described as follows:

Algorithm:

1. Input gray image
2. Convert the input image into 8-bit binary image
3. On each bit image individually apply BWT up to 3 scales and obtain multi resolution binary images
4. Convert 8 bit binary subbands into gray scale back
5. For each subband calculate histogram feature (***F₁***)
6. Compute auto-correlogram feature (***F₂***) in the direction of subbands
7. Construct the final feature vector of the given image by giving weights to histogram and auto-correlogram features.

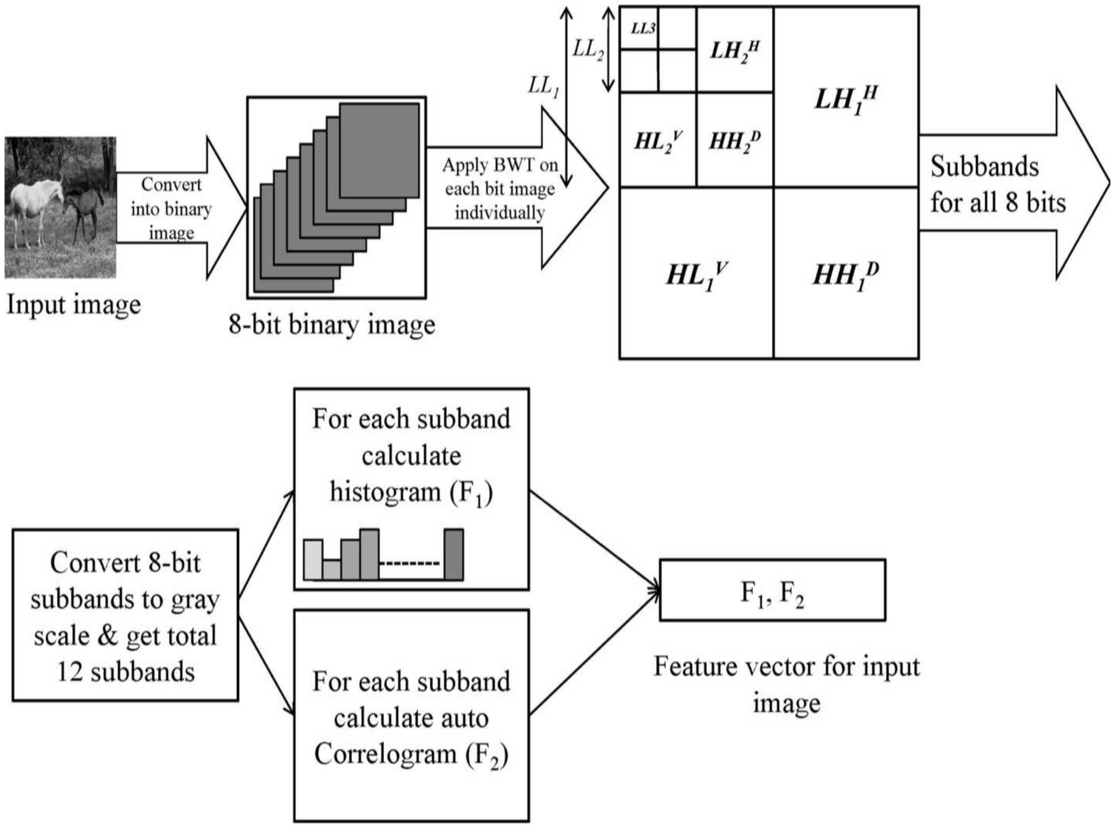

Input image
Convert into binary image
8-bit binary image
Apply BWT on each bit image individually
LL_1
LL_2
LL3
LH_2^H
LH_1^H
HL_2^V
HH_2^D
HL_1^V
HH_1^D
Subbands for all 8 bits
Convert 8-bit subbands to gray scale & get total 12 subbands
For each subband calculate histogram (F_1)
For each subband calculate auto Correlogram (F_2)
F_1, F_2
Feature vector for input image

Fig. 3.18: Proposed method 3 computation

3.3.2 Similarity measurement

To find out similarity among query and database images, a query image is chosen from image database. Query image and database images are processed to compute features. In the experiment all images are considered as query to evaluate average retrieval performance on the image database. Distance measure given by Eq. (3.17) is used to compute the difference between query image and database image features.

$$D(Q,T) = A\sum_{i=1}^{\lambda_1} \frac{\left|Q_{1i}-T_{1i}\right|^2}{\left|1+Q_{1i}+T_{1i}\right|^2} + B\sum_{i=1}^{\lambda_2}\left|\frac{\left|Q_{2i}-T_{2i}\right|^2}{\left|1+Q_{2i}+T_{2i}\right|^2}\right| \qquad (3.17)$$

Where, Q_{1i} and Q_{2i} are i^{th} feature components of query features vector $\boldsymbol{F_1}$ and $\boldsymbol{F_2}$ respectively. T_{1i} and T_{2i} are i^{th} feature components of database image features $\boldsymbol{F_1}$ and $\boldsymbol{F_2}$ respectively. λ_1 and λ_2 are feature vector lengths of $\boldsymbol{F_1}$ and $\boldsymbol{F_2}$ features respectively. $A = 0.4$ and $B = 0.6$ are weights of $\boldsymbol{F_1}$ and $\boldsymbol{F_2}$ features selected by experimentation. Database image having small *D (Q, T)* value is more relevant to query image. Similarly, a list of images is retrieved depending upon relevancy.

3.3.3 Experimental results and discussions

Experiments are conducted on the standard Corel 1000 and Corel 2450 databases (Appendix A) [167]. Various performance measures like *AP %*, *AR %* and *ARR %* are calculated by using Eq. (3.9), (3.12) and (3.13) respectively. Results of proposed method 3 are compared with auto-correlogram (AC), OQWC [83], GWC [91] and BWTH (proposed method 2). AC is implemented by quantizing the input image intensity values into 32 levels and then auto correlogram is calculated in four directions with unit distance. In OQWC computation, DWT of the input image is computed upto three scales and $0°$, $\pm 45°$ and $90°$ directions. Obtained wavelet coefficients are quantized to a limited number of levels by using the optimization algorithm. Then, auto-correlogram of vertical and horizontal wavelet coefficients is computed. GWC avoids the problem of diagonal mixing and computes the wavelet coefficients by using Gabor wavelets in $0°, +45°, -45°$ and $90°$ directions and three scales. The coefficients are discretized through the experimentally obtained quantization thresholds for better performance. Finally, the autocorrelogram of the quantized coefficients is computed along the direction normal to Gabor wavelet orientation with unit distance. Implementation of BWTH is given in section 3.2.3.

3.3.3.1 Corel 1000 Database (DB1)

The first experiment is conducted on DB1 (Appendix A) [167]. Table 3.7 illustrates the precision of proposed method 3, AC, OQWC, GWC and BWTH for each group of database. (*T: Total Number of Retrieved Images Considered). It is verified from Table 3.7 that the *AP%* and *AR%* of proposed method 3 (69.43 and 41.81) are significantly improved as compared to AC (57.67 and 36.21**),** OQWC (64.3 and 38.00), GWC (64.1 and 40.6) and BWTH (66.62 and 40.64). Fig. 3.19 illustrates the effect of different numbers of retrieved images on *AP%*. It is evident from graph of *AP%* that proposed method 3 always perfromes better than AC, OWQC, GWC and BWTH.

Fig 3.20 shows the *ARR %* of proposed method 3 and comparison with AC, OQWC, GWC and BWTH. It is clearly depicts from Fig 3.20 that proposed method 3 always shows better performance in case of different number of retrieved images. For top 10 and 100 retrieved images *ARR %* of proposed method 3 is (6.94 and 41.81) while for AC is (5.67 and 36.21) for OQWC is (6.43 and 38) for GWC is (6.41 and 40.6) and for BWTH is (6.62 and 40.64) respectively. Fig 3.21 shows four query retrieval results (a) 320 (Bus), (b) 420 (Dianosaur), (c) 670 (Flower), (d) 750 (Horse) of proposed method 3.

Table 3.7

Retrieval performance comparison of proposed method 3

Group Name	PM 3		AC		OQWC		GWC		BWTH	
	P% (T=10)	*R% (T=100)*	*P% (T=10)*	*R% (T=100)*	*P% (T=10)*	*R% (T=100)*	*P% (T=10)*	*R% (T=100)*	*P% (T=10)*	*R% (T=100)*
Africans	74	37.54	74.2	49.7	57.7	31.1	52.9	33.2	76.1	38.42
Beaches	44.2	23.29	30	15.68	49.3	28.6	42	26.2	43.1	22.94
Buildings	55.6	29.46	40.4	20.25	50.9	30.5	47.8	26.5	45.8	26.39
Buses	83.7	53.03	39.7	22.69	87.1	64.0	88.3	65.1	72.6	47.99
Dinosaurs	100	94.84	100	97.73	74.6	28.8	96.2	65	99.9	94.84
Elephants	64.8	32.44	55.1	31.41	55.7	30.7	65.9	37	65	34.58
Flowers	93.7	66.25	86.7	47.57	84.3	65.3	75.5	50.4	93.9	65.39
Horses	78.2	30.39	70.9	33.48	78.9	39.9	73	39.5	74.9	28.68
Scenery	34.9	19.06	21.9	12.54	47.2	25.1	35.2	20.1	30.8	16.43
Foods	65.2	31.83	57.8	31.07	57.1	36.4	63.2	43.1	64.1	30.74
Average	**69.43**	**41.81**	**57.67**	**36.21**	**64.3**	**38.0**	**64.1**	**40.6**	**66.62**	**40.64**

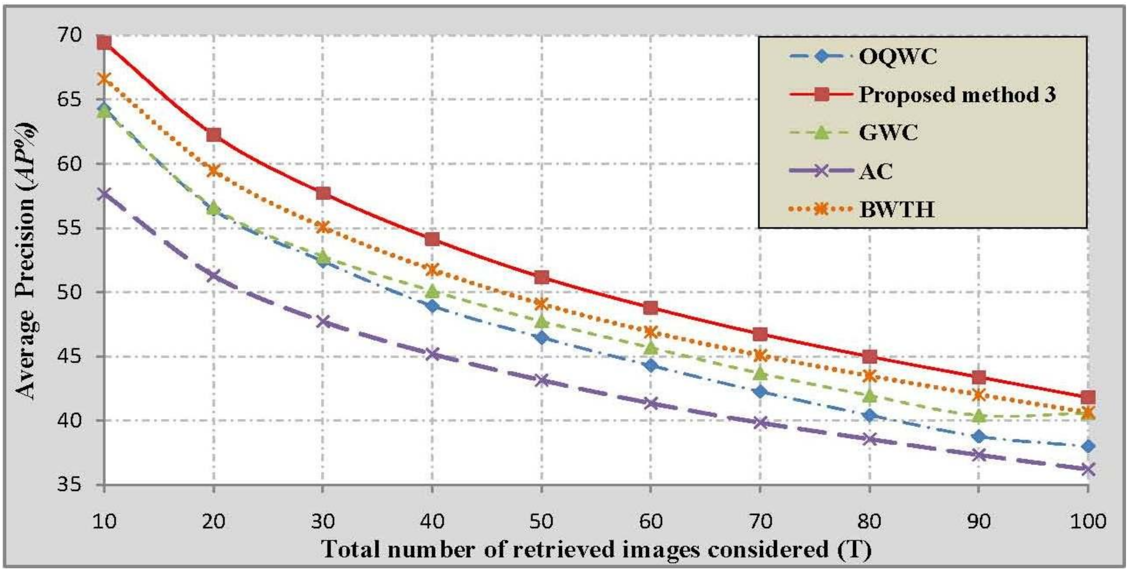


Fig. 3.19: Average precision (*AP%*) according to different numbers of retrieved images

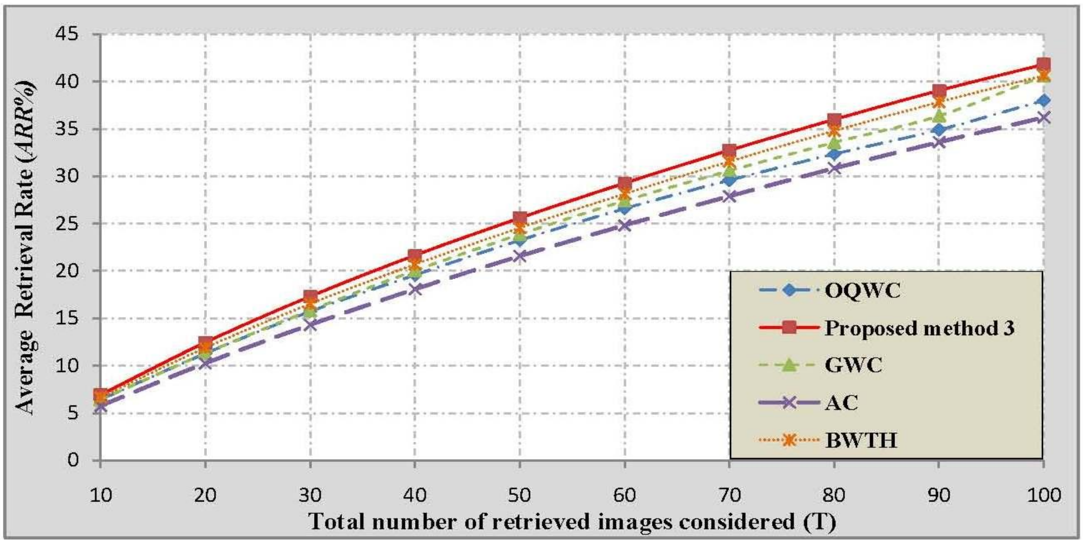


Fig. 3.20: Average retrieval rate (*ARR%*) according to different number of retrieved images

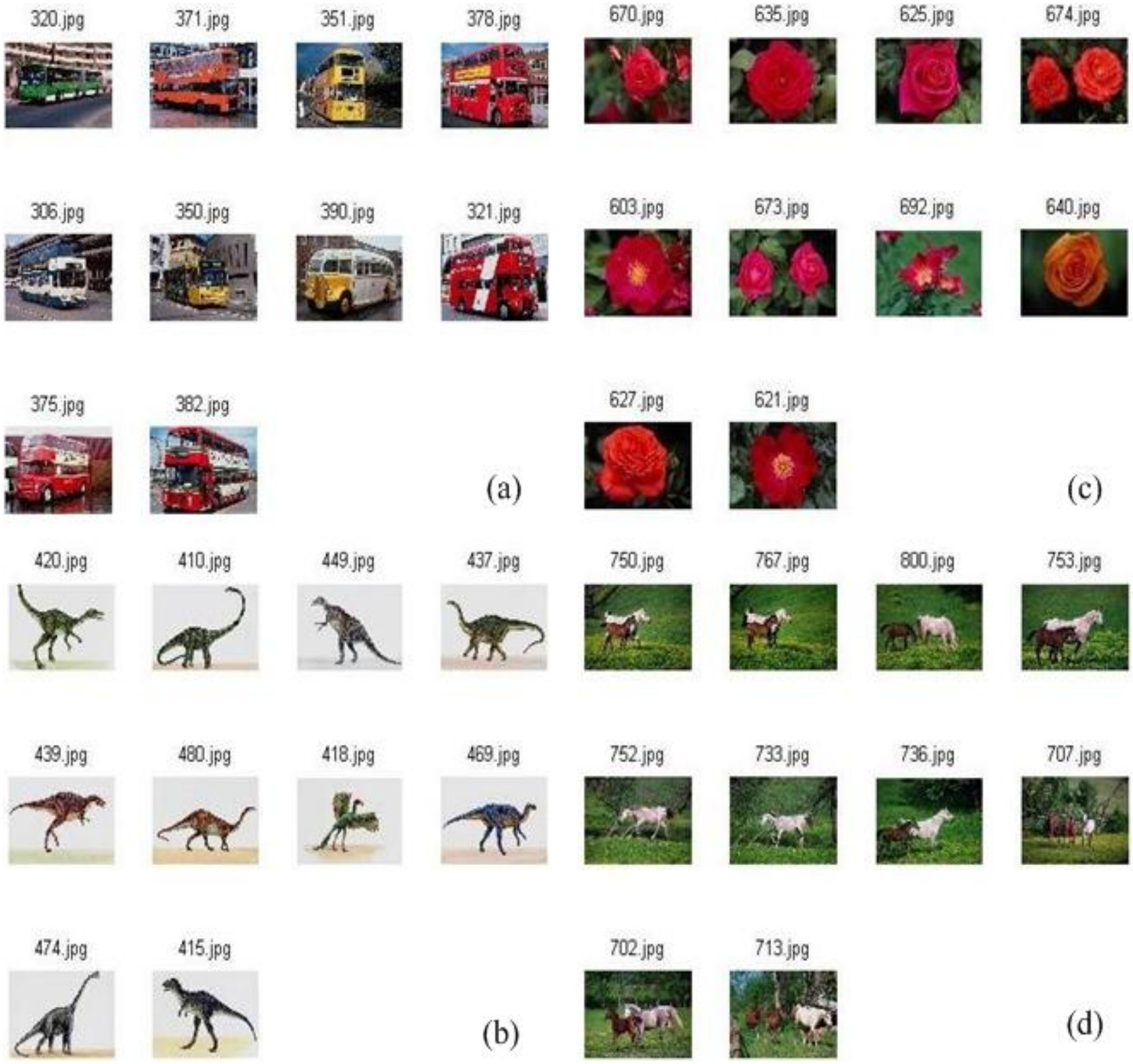


Fig. 3.21: Retrieval result for four query images

In Fig 3.21 first image is the query image. It is clear from Fig 3.21 that retrieved images are relevant to the query image. Thus, performnce of proposed method 3 is justified by retrieval result as well.

3.3.3.2 Corel 2450 Database (DB2)

The next experiment is perfromed on Corel 2450 Database (Appendix A) [167]. Average precision of PM3 in comaprison to AC, OQWC, BWTH, DBWP is shown in Fig. 3.22 with respect to various number retrievd images. It is illutrated from the Fig. 3.22 that the PM3 (59.24 and 38.39) is giving better *AP%* as comapred to AC (51.60 and 33.34), OQWC (54.55 and 36.55), DBWP (55.7 and 35.7) and BWTH (56.07 and 36.68).

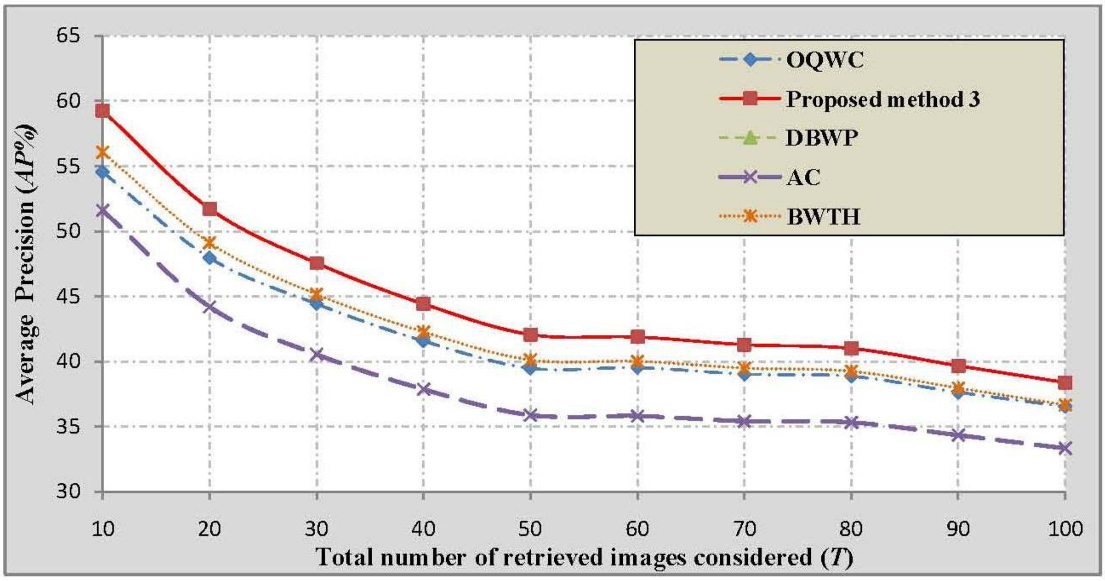


Fig. 3.22: Average precision (*AP%*) according to different numbers of retrieved images

3.3.4 Summary

This section has proposed a new, integrated feature descriptor by using BWT. In the primary step of feature calculation binary wavelet coefficients are extracted and further spatial relationship among binary wavelet coefficients are calculated using auto correlogram. As well as histogram of binary wavelet coefficients is also concatenated into proposed feature. This allows incorporating more information about an image into feature descriptor. It is verified through the experimental analysis that proposed method 3 (69.43

and 41.81) has significantly improved *AP%* and *AR%* compared to AC (57.67 and 36.21), OQWC (64.3 and 38.0) and GWC (64.1 and 40.6) on standard Corel 1000 image database respectively. In the similar manner proposed method 3 outperforms AC, OQWC and BWTH on Corel 2450 as well. In the further work the performance of PM3 can be enhanced by operating on color levels.

In the next chapter subbands decomposition is performed in á trous manner. It avoids the down sampling operation of pyramidal structure and hence translation invariant property is achieved. Thus, marked improvement in the retrieval results is expected.

Á trous Wavelet Transform Based Features

Chapter 4

In the preceding chapter subbands are decomposed in pyramidal manner thus, down sampling is necessary. It causes the drawback of translation variance. In order to avoid this problem in this chapter subbands are decomposed in the á trous manner.

Since, multiresolution techniques allow decomposition of an image at multiple scales hence analysis is performed to extract features from different scales. Each resolution characterizes different information about an image. Coarse resolutions tell about large structural design and as we move towards fine resolution, minute details about structures are acquired. Decomposition can be executed in two different manners; pyramid and á trous. In contrast to pyramidal structure, different spatial resolutions of á trous wavelet transform (AWT) have the same number of pixels as the original image. So, analysis among subbands of different scales is possible. Also, by avoiding the down sampling, translation invariant property is achieved. These properties encourage authors to perform multiresolution analysis by AWT and extract texture statistics by its subbands. This chapter presents two different methods based on á trous wavelet.

Firstly a new multiresolution approach named as á trous wavelet correlogram (AWC) is proposed for texture feature extraction. In the designing of AWC feature, distribution of intensity variations in AWT subbands is extracted to get the textural information. AWT provides information about all orientations in a single subband.

To overcome the problem of single subband confinement of orientation information by AWT, in the second method á trous gradient structure descriptor (AGSD) is proposed for CBIR. AGSD provides not only intensity variations but also the information regarding orientations of intensity variations in á trous wavelet subbands. The local information regarding statistics of orientations in the image is extracted through microstructure descriptor (MSD). It finds the relation between neighborhood pixel’s orientations. Finally, with the help of á trous quantized coefficients and MSD mask, AGSD feature is computed. Experiments are conducted to approve the retrieval performance of proposed features on three benchmark image databases; Corel 1000, Corel 2450 and MIRFLICKR 25000 (Appendix A).

Various performance measures like average precision, weighted precision, standard deviation of weighted precision, average recall, standard deviation of recall, rank etc. of proposed methods are calculated to evaluate the retrieval performance.

The results are compared with other existing transform based texture features like, optimal quantized wavelet correlogram (OQWC) [83], Gabor wavelet correlogram (GWC) [91], GWC with evolutionary group algorithm (EGA) [92], combination of standard wavelet filter with rotated wavelet filter correlogram (SWF+RWF correlogram) [170] and texton co-occurrence matrix (TCM) [171]. It is worth to notice that the proposed methods have improved the retrieval performance significantly on the basis of various performance measures.

Out of the two proposed methods present in this chapter the AWC is being called as PM1 and the other AGSD is called as PM2 in the subsequent discussions of this chapter. The first step in the implementation of these methods is á trous wavelet transform (AWT). Computation of AWT is explained in the following section

4.1 Á TROUS WAVELET TRANSFORM (AWT)

The multiresolution analysis can be performed in two manners, pyramid structure and á trous structure. In pyramidal structure images at each scale are down sampled by a factor of two. It causes reduction in subband sizes at each and every scale. But in case of á trous structure down sampling is not performed. It results in the number of approximation coefficients always equals to the number of image pixels. So, analysis of subbands among different scales is possible. By avoiding the down sampling translation invariant property is also achieved. In contrast to pyramid structural wavelet transform no classification among horizontal, vertical, and diagonal subbands are present in á trous structure [80]. These properties of á trous wavelet transform are utilized in the proposed features extraction. Given an image *I* of size $X \times Y$ and resolution 2^j, at each scale of á trous wavelet transform, approximation of image *I* with coarser spatial resolution is obtained. Likewise, by dyadic decomposition approach at the N^{th} scale the resolution of the approximation image is 2^{j-N}. The size of each approximation image is always equal as the image *I*. The scaling function used for calculating approximation image is B_3 cubic spline and is given by low-pass filter shown by Eq. 4.1:

$$h=\frac{1}{256}\begin{bmatrix}1&4&6&4&1\\4&16&24&16&4\\6&24&36&24&6\\4&16&24&16&4\\1&4&6&4&1\end{bmatrix} \tag{4.1}$$

The approximation image at first scale is given by Eq. (4.2).

$$I_{2^{j-1}}=I_{2^j}\otimes h \tag{4.2}$$

In order to obtain coarser approximations of the original image, the above filter must be filled with zeros, to match the resolution of desired scale. The detailed information lost between the 2^j and 2^{j-1} images are collected in one wavelet coefficient image W_2^{j-1}, by subtracting corresponding approximation coefficients at consecutive decomposition scales. This wavelet plane represents the horizontal, vertical and diagonal detail between 2^j and 2^{j-1} resolution as given by Eq. 4.3.

$$W_{2^{j-1}}=I_{2^j}-I_{2^{j-1}} \tag{4.3}$$

The original image I_2^j can be reconstructed exactly by adding approximation image I_2^{j-N} at N^{th} scale to the wavelet subbands up to N^{th} scale and it is given by Eq. (4.4).

$$I_{2^j}=I_{2^{j-N}}+\sum_{i=1}^{N}W_{2^{j-i}} \tag{4.4}$$

4.2 PROPOSED METHOD 1 (PM1)

Structural properties at AWT subbands are utilized for AWC computation. Á trous coefficients have a wide dynamic range of real numbers, so directly it is not suitable for correlogram calculation. Á trous wavelet coefficients are calculated up to three scales and to avoid the problem of large dynamic range, each scale is quantized into 16 levels using the quantization thresholds given in Eq. 4.5. $L1$, $L2$,...,$L15$ define the quantization thresholds. Fig. 4.1 shows quantization thresholds and levels for á trous wavelet coefficients at different scales. AWC is proposed by utilizing correlation among á trous wavelet coefficients. After quantization of á trous wavelet coefficients, for AWC extraction auto correlogram is computed on quantized á trous wavelet coefficients at each scale and four directions.

$$\left.\begin{array}{l}
\text{First scale} \\
L1=-30,\, L2=-20,\, L3=-15,\, L4=-10,\, L5=-6,\, L6=-3,\, L7=-1,\, L8=-0.15, \\
L9=0.15,\, L10=1.8,\, L11=4.5,\, L12=10,\, L13=20,\, L14=35,\, L15=50. \\
\text{Second scale} \\
L1=-20,\, L2=-18,\, L3=-15,\, L4=-12,\, L5=-8,\, L6=-4,\, L7=-1,\, L8=-0.15, \\
L9=0.15,\, L10=4,\, L11=8,\, L12=15,\, L13=25,\, L14=40,\, L15=55. \\
\text{Third scale} \\
L1=-15,\, L2=-12,\, L3=-10,\, L4=-9,\, L5=-6,\, L6=-3,\, L7=-1.5,\, L8=-0.4, \\
L9=0.4,\, L10=2.5,\, L11=5.5,\, L12=11,\, L13=21,\, L14=36,\, L15=56.
\end{array}\right\} \quad (4.5)$$

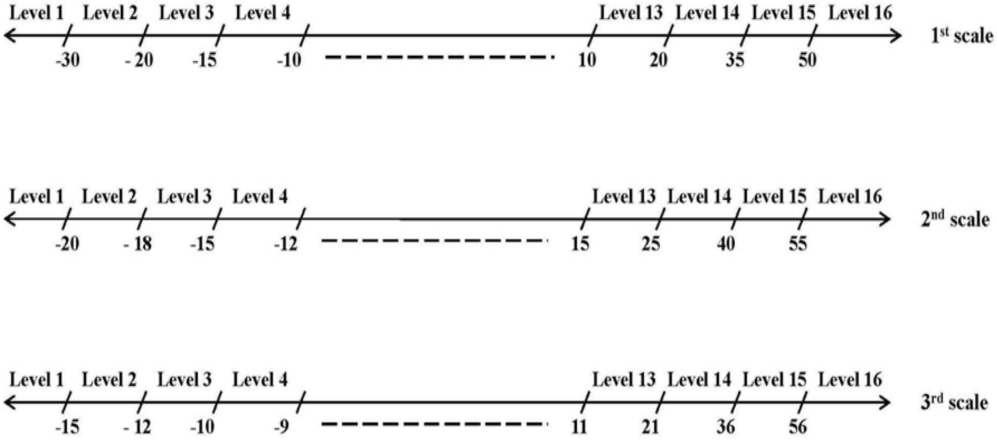


Fig. 4.1: Quantization threshold for á trous wavelet decomposition scales

In Fig. 4.2 structure for PM1 is drawn and algorithm for PM1 is given as follows:

Algorithm 1:

Input: RGB image

Output: PM1 feature vector for the input image

1. Take RGB image and convert it into gray image.
2. Calculate á trous wavelet transform up-to three scales using Eq. (4.3).
3. Quantize á trous wavelet coefficients at each scale as given by Eq. (4.5).
4. On quantized á trous wavelet transformed image I at each scale (s), auto correlogram is calculated in four directions by using Eq. (4.6).

$$F_{i,j}{}^{(k)}{}_{s} = \Pr_{p_1 \in I_{c(i)},\, p_2 \in I}\left[\, p_2 \in I^{s}{}_{c(j)} \,\middle\|\, p_1 - p_2 \right| = k \,\Big] \qquad (4.6)$$

$$i = j \;\; for\; autocorrelogram, k = 1$$

$I_{c(i)}^{\ s}$ represents the set of pixels whose values are *c(i), s(=1,2,3)* represents the scale of á trous wavelet transformed image and $|p_1 - p_2|$ is the distance between p_1 and p_2.

5. For each scale $F_{i,j}^{(k)}{}_s$ is calculated and combined for the construction of PM1 feature vector.

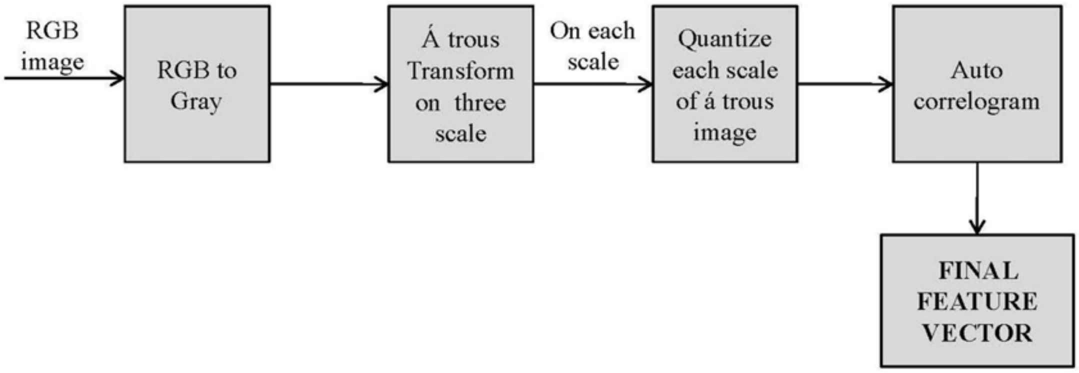


Fig. 4.2: Á trous wavelet correlogram extraction

4.3 PROPOSED METHOD 2 (PM2)

Orientation of image pixels possess immense amount of information regarding image texture. Orientations are invariant to many changes like color, illumination and scale. By further work in PM1 direction á trous gradient structure descriptor (AGSD) is introduced as PM2 by incorporating the orientation information. In the calculation of AGSD the first step is to obtain á trous wavelet transformed image (section 4.1). From á trous wavelet transformed coefficients, orientation information is obtained. Relationship among á trous transformed image and its orientations information is extracted and MSD image is obtained. In the final step of feature calculation autocorrelogram is calculated from MSD image at each scale. The following sections illustrate all the processing steps.

4.3.1 Orientation angle calculation

In order to calculate PM2, from á trous transformed images, orientation information is extracted. If *L(x, y)* is the value of pixel at coordinate *(x, y)* then orientation value can be calculated using Eq. 4.7.

$$Orient(x,y) = \tan^{-1}((L(x,y+1) - L(x,y-1)) / (L(x+1,y) - L(x-1,y))) \tag{4.7}$$

It gives the orientation values ranging from $-90^o\ to + 90^o$. These orientation values are quantized into 17 levels.

4.3.2 Microstructure descriptor (MSD)

Textures are created by simple primitive texture elements. A typical example is Julesz's texton theory [16], but it emphasizes on regular texture images. To avoid this problem, the microstructures are used here for image representation. Contents of natural images can be considered as constructed by many microstructures [58]. The microstructures compute the similarity of orientation in the neighborhood. The quantized orientation image is having values ranging from 1 to 17. For the implementation of MSD, the quantized orientation image is divided into 3 × 3 regions, called blocks. Flow diagram for MSD calculation is shown in Fig. 4.3 and algorithm for MSD calculation is as follows:

Algorithm 2:

Input: Quantized orientation image, á trous transformed image

Output: MSD image

1. Select á trous wavelet transformed image and quantized orientation image at first scale
2. Quantized orientation image (Fig. 4.3(a)) is partitioned off into 3×3 block size. The center pixel *(CP)* of each block is compared with eight neighborhood pixels *(BK)*. If any one of the eight nearest neighbors has the same value as the center pixel, then keep that pixel value unchanged; otherwise it is set to empty. Pseudo code is given by

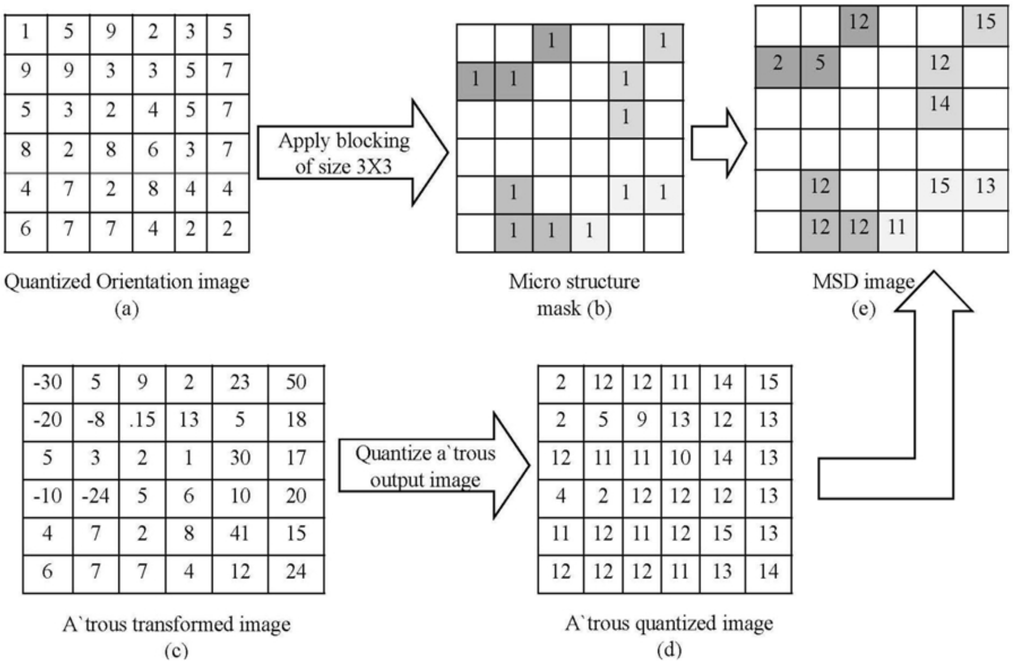
```
For i=1:8
    If CP==BK (i)
      b (i)=CP=1
    Else
      b (i)=0
    End
End
```

Likewise, microstructure mask (Fig. 4.3(b)) is obtained.

3. Á trous transformed image (Fig. 4.3(c)) is quantized into 16 levels using Eq. (4.5) and Fig. 4.3(d) is obtained.
4. The *MSD* image (Fig. 4.3(e)) is obtained by keeping á trous quantized image values (Fig. 4.3(d)) and using Fig. 4.3(b) as a mask. Image in Fig. 4.3(b) is imposed on á trous quantized image (Fig. 4.3(d)) and non-zero values are retained as shown in Fig. 4.3(e).

```
If b (x, y) !=0
  MSD image (x, y) = á trous quantized image (x, y)
Else
  MSD image (x, y) =0;
End
```

   This generates final *MSD* image (Fig. 4.3(e)).
5. In a similar manner *MSD* image is calculated for all scales of á trous transformed image.



1	5	9	2	3	5
9	9	3	3	5	7
5	3	2	4	5	7
8	2	8	6	3	7
4	7	2	8	4	4
6	7	7	4	2	2

Quantized Orientation image
(a)

		1			1
1	1			1	
				1	
	1			1	1
	1	1	1		

Micro structure
mask (b)

		12			15
2	5			12	
				14	
	12			15	13
	12	12	11		

MSD image
(e)

-30	5	9	2	23	50
-20	-8	.15	13	5	18
5	3	2	1	30	17
-10	-24	5	6	10	20
4	7	2	8	41	15
6	7	7	4	12	24

A`trous transformed image
(c)

2	12	12	11	14	15
2	5	9	13	12	13
12	11	11	10	14	13
4	2	12	12	12	13
11	12	11	12	15	13
12	12	12	11	13	14

A`trous quantized image
(d)

Fig. 4.3: MSD image construction

Fig. 4.4 shows the PM2 (AGSD) framework for image feature extraction where the area without dotted line covers PM1.

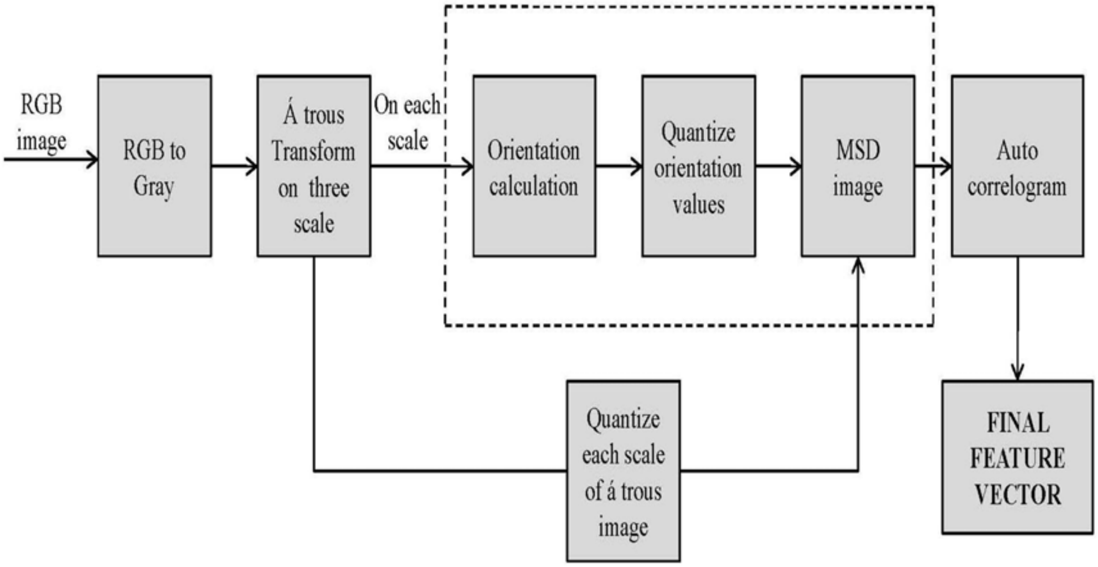


Fig. 4.4: Á trous gradient structure descriptor extraction

Implementation algorithm of the PM2 is given below.

Algorithm 3:

Input: RGB image

Output: PM2 feature vector for the input image

1. Convert input RGB image into gray image *(I)*
2. Find the á trous wavelet transform of image *I* up to three scales using Eq. (4.3)
3. From each scale of á trous wavelet transformed image extract the orientation information using Eq. (4.7)
4. Quantize the orientation image into 17 levels
5. Calculate the *MSD* image from each scale of quantized á trous wavelet transformed image and orientation image (Refer *Algorithm 2*)
6. From each *MSD* image *M,* calculate the auto correlogram for four directions by using Eq. (4.8).

$$F_{i,j}{}^{(k)}{}_{s} = \underset{p_1 \in I_{c(i)}, p_2 \in I}{Pr}\left[p_2 \in M_{c(j)}{}^{s} \left\| p_1 - p_2 \right| = k \right] \tag{4.8}$$
$$i = j \ for\ autocorrelogram, k = 1$$

$M_{c(i)}{}^{s}$ represents the set of pixels whose values are *c(i), s(=1,2,3)*represents the scale of á trous wavelet transformed image and $\left| p_1 - p_2 \right|$is the distance between p_1 and p_2.

7. For each scale ${F_{i,j}^{(k)}}_s$ is calculated and combined for the construction of PM2 feature vector.

4.4 SIMILARITY MEASUREMENT

To find the resemblance among query and database images, query and database images are processed to compute features. (Query image can be any image selected by user from the image database. In the retrieval experiments each image is considered as a query to obtain average result on the database.) Distance measure *d₁*, given by Eq. (4.9) is used to compute the difference between query image and database images based on extracted features. Database images having small *D (Q, T)* value are considered as more relevant to query image and are retrieved by the system.

$$D(Q,T) = \sum_{i=1}^{\lambda} \frac{|Q_i - T_i|}{|1 + Q_i + T_i|} \tag{4.9}$$

Where, Q_i is i^{th} feature vector of query image, T_i is i^{th} feature vector of database image and λ is the feature vector length.

4.5 EXPERIMENTAL RESULTS AND DISCUSSIONS

All the experiments are performed on three standard image databases viz. Corel 1000 (DB1), Corel 2450 (DB2) and MIRFLICKR 25000 (DB3) (Appendix A). Various performance measures like precision, recall, and rank, etc. are calculated to compare the performance of proposed features with some of the already existing features OQWC [83], GWC [91], GWC+EGA [92], SWF+RWF correlogram [170] and TCM [171]. In OQWC computation, discrete wavelet transform (DWT) of the input image is computed upto three scales and resultant wavelet coefficients are quantized to a limited number of levels by using the optimization algorithm. Then, auto-correlogram of vertical and horizontal wavelet subbands is computed. GWC excludes the problem of diagonal mixing in DWT and computes the wavelet subbands by using Gabor wavelets in $0°, +45°, 90°$ and $-45°$ directions and three scales. The subbands coefficients are discretized by using the experimentally obtained quantization thresholds for better performance. Finally, the autocorrelogram of the quantized coefficients is computed along the direction normal to Gabor wavelet orientation with unit distance. Further in [92] GWC coefficients are quantized by EGA before correlogram calculation. Subrahmanyam *et al.* [170] proposed the combination of SWF and RWF to collect the four directional $\left(0°,\ +45°,\ 90°\,\text{and}\,\text{-}45°\right)$ information from the image and further used it for correlogram feature calculation. To accomplish this, $0°$ and $90°$ subbands information of the image are collected from SWF and $+45°$ and $\text{-}45°$ are collected from RWF. In [171] AWT is used to decompose the image into successive scales and texton elements are used to search for particular texture in the image. In order to generate TCM features, correlogram on texton image is computed in four different directions. AGSD extracts local orientation information at each á trous wavelet subband and analyze the orientation statistics. Orientation statistics is evaluated locally to find out the resembling orientation portions and only corresponding wavelet coefficients are utilized for correlogram computation, thus, it confines information at local as well as global levels. This property permits the AGSD to perform better as compared to other existing methods.

Performance measures used for retrieval performance comparison are as follows:
Precision of any query image I_q can be obtained by Eq. (4.10)

$$P\,(I_q) = \frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ retrieved\,images\,considered\,(T)} \tag{4.10}$$

The weight precision of query image I_q is obtained by assigning weight to each relevant image retrieved and is given by Eq. (4.11).

$$WP\left(I_q\right) = \sum_{k=1}^{T} \frac{1}{k}\frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ retrieved\,images\,considered\left(T\right)} \tag{4.11}$$

Average precision for each group or database is formulated by Eq. (4.12)

$$AP = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} P_i \tag{4.12}$$

Γ is number of images; in case of group precision its total number of images in a particular group whereas in case of average precision of database it is total number of images in the database. Similarly, average weighted precision for each group or database is given by Eq. (4.13)

$$P_{wt} = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} WP_i \tag{4.13}$$

Likewise, same parameters are calculated for recall using Eq. (4.14) and (4.15).

$$R(I_q) = \frac{No.of\ relevant\, images\, retrieved}{Total\ no.of\ relevant\, images\, in\, database} \tag{4.14}$$

$$AR = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} R_i \tag{4.15}$$

Weighted recall and average weight recall can be calculated in a manner similar to Eq. (4.11) and (4.13).

Total average retrieval rate is given by Eq. (4.16)

$$ARR = \frac{1}{DB}\sum_{j=1}^{DB} R_j \Bigg|_{T \le 100} \tag{4.16}$$

Where, *DB* is total number of images in the database.

Rank of query image I_q is given by Eq. (4.17)

$$C\left(I_q\right) = \frac{1}{\Gamma}\sum_{\delta(I_i)=\delta\left(I_q\right)} Rank\left(I_i, I_q\right) \tag{4.17}$$

$\delta(x)$ is the category of x^{th} image. $Rank\left(I_i, I_q\right)$ returns the rank of image I_i (for the query image I_q) among all images of *DB.* Average rank C_{avg} can be calculated in a manner similar to Eq. (4.15).

Standard deviation of precision for each group or total database is another performance parameter obtained by Eq. (4.18).

$$P_{std} = \sqrt{\frac{1}{\Gamma - 1}\sum_{i=1}^{\Gamma}\left(P\left(I_i\right) - AP\right)^2} \tag{4.18}$$

Standard deviation of recall for each group or total database is obtained by Eq. (4.19)

$$R_{std} = \sqrt{\frac{1}{\Gamma - 1}\sum_{i=1}^{\Gamma}\left(R\left(I_i\right) - AR\right)^2} \tag{4.19}$$

Similarly, standard deviation of weighted precision and standard deviation of average rank can be obtained. Average of precision, weighted precision and recall should attain higher values whereas, average value of rank, standard deviation of all evaluation measures should achieve lower values to get good performance of retrieval system.

4.5.1 Corel 1000 Database (DB1)

Experiments are performed on database DB1 (Appendix A) and parameters given by (Eq. 4.10 to 4.19) are calculated. The authors have first applied PM1 for CBIR application. Tables 4.1 to 4.5 justify the performance of proposed methods with respect to other existing methods. It is observed that PM1 results are better than [83, 91, 92, 170 and 171]. Tables 4.1, 4.2, 4.3, 4.4 and 4.5 show parameters P_{wt}, $P_{\text{wt\_std}}$, *AP*, P_{std}, *AR*, R_{std}, C_{avg} and C_{std} values for OQWC [83], SWF+RWF correlogram [170], TCM [171], PM1 and PM2 respectively.

Table 4.6, Table 4.7 gives P_{avg}, R_{avg} and C_{avg} results for GWC [91] and GWC+EGA [92]. From Table 4.4 it is evident that the average precision (*AP%*) and average recall (*AR%*) of PM1 (70.24 and 44.60) surpass than OQWC (64.3 and 38), GWC (64.1 and 40.6), GWC+EGA (65.01 and 41.18), SWF+RWF correlogram (66.5 and 41.3), TCM (67.41 and 42.32). But, again with Table 4.5, it is verified that PM2 (70.54% and 46.12%) has improved the retrieval performance than PM1 itself. As observed from Tables 4.1, 4.2, 4.3, 4.4, 4.5, 4.6 and 4.7 that the performance of PM2 is better than that of other methods [83, 91, 92, 170, 171].

Table 4.1

OQWC results in terms of various performance measures

Category Name	**P_{wt}%**	**P_{wt_std}%**	**P%**	**P_{std}%**	**R%**	**R_{std}%**	**C_{avg}**	**C_{std}**
Africans	68.2	25	57.7	29.2	31.1	12.7	282	78
Beaches	61.9	23.4	49.3	28.2	28.6	16.5	335	131
Buildings	63.2	20.6	50.9	23.7	30.5	12.1	308	141
Buses	91.2	15.3	87.1	20.5	64	16.4	108	79
Dinosaurs	82.8	22	74.6	28.5	28.8	10.6	410	91
Elephants	70.7	17.5	55.7	20.9	30.7	8.4	235	44
Flowers	88.3	19.4	84.3	24.3	65.3	19.9	125	82
Horses	85.9	19.5	78.9	23.1	39.9	13.9	264	99
Mountains	60	21.2	47.2	23.7	25.1	9.7	324	79
Food	67.5	26	57.1	31.8	36.4	14.3	236	57
Total	**74**	**23.9**	**64.3**	**29.4**	**38**	**19.6**	**263**	**127**

Table 4.2

SWF+RWF correlogram results in terms of various performance measures

Category Name	**P_{wt}%**	**P_{wt_std}%**	**P%**	**P_{std}%**	**R%**	**R_{std}%**	**C_{avg}**	**C_{std}**
Africans	74.2	24	64.9	29.3	34.7	14.5	254.8	73.7
Beaches	56	20.7	43.5	24.9	27.8	14.8	350.4	131
Buildings	64.9	22.4	52.4	26.8	31.1	13.7	294.5	110.5
Buses	93	15	90.3	19.4	69.5	19.5	100.3	70.1
Dinosaurs	87.5	20.1	81.2	26.6	45.9	19.2	208.5	103.9
Elephants	71.9	17.9	58.4	20.1	33.7	7.6	240.3	57.4
Flowers	91.2	15.6	86.7	20.9	58.5	19.5	172.4	103.2
Horses	86.1	16.5	80	20.6	47.2	16.1	201.7	95.2
Mountains	53.6	18.8	37.5	19.1	20	8.4	391.8	79.4
Food	75.6	23.7	67.8	28.2	44.1	15.3	194.3	74.6
Total	**75.5**	**19.5**	**66.5**	**23.6**	**41.3**	**14.9**	**240.9**	**89.9**

Table 4.3

TCM results in terms of various performance measures

Category Name	**P_{wt}%**	**P_{wt_std}%**	**P%**	**P_{std}%**	**R%**	**R_{std}%**	**C_{avg}**	**C_{std}**
Africans	70.32	22.30	66.70	27.25	36.17	13.83	299.84	98.40
Beaches	70.39	21.93	50.80	26.67	28.82	18.62	265.05	141.17
Buildings	65.23	23.23	50.40	27.49	25.11	10.54	312.21	81.13
Buses	91.38	15.13	89.10	18.93	61.76	14.36	94.21	45.04
Dinosaurs	97.79	9.41	96.70	13.03	83.81	16.18	75.86	53.78
Elephants	67.66	18.95	52.80	21.28	25.56	5.94	275.56	36.08
Flowers	90.58	14.55	86.50	17.62	57.67	18.98	91.17	79.86
Horses	82.15	20.86	77.30	26.59	43.88	11.92	260.70	99.91
Mountains	53.91	18.06	34.80	19.12	20.70	9.75	315.94	90.88
Food	75.82	21.96	69.00	27.76	39.70	12.91	225.18	65.36
Total	**76.52**	**18.64**	**67.41**	**22.57**	**42.32**	**13.30**	**221.57**	**79.16**

Table 4.4

PM1 results in terms of various performance measures

Category Name	**P_{wt}%**	**P_{wt_std}%**	**P%**	**P_{std}%**	**R%**	**R_{std}%**	**C_{avg}**	**C_{std}**
Africans	77.50	26.10	68.30	32.57	34.25	14.96	282.37	99.30
Beaches	62.77	22.07	49.70	25.16	28.90	13.69	313.00	113.16
Buildings	70.65	24.05	59.70	29.39	32.94	15.84	264.43	117.08
Buses	93.86	13.30	91.90	16.98	66.97	14.27	99.32	54.41
Dinosaurs	99.95	00.48	99.90	01.00	95.92	07.34	53.55	14.47
Elephants	73.60	17.40	58.60	18.80	27.11	06.53	290.05	52.49
Flowers	96.53	10.37	93.30	14.64	59.21	17.75	154.28	114.82
Horses	87.84	17.23	81.20	23.96	46.18	17.12	198.76	95.10
Mountains	56.67	16.70	41.80	16.78	22.60	07.13	350.56	89.45
Food	69.52	22.84	58.00	27.71	31.92	12.45	249.75	73.57
Total	**78.89**	**17.05**	**70.24**	**20.70**	**44.60**	**12.71**	**225.61**	**82.38**

Table 4.5

PM2 results in terms of various performance measures

Category Name	**P_{wt}%**	**P_{wt_std}%**	**P%**	**P_{std}%**	**R%**	**R_{std}%**	**C_{avg}**	**C_{std}**
Africans	78.23	24.14	68.80	30.16	36.11	16.01	267.50	99.06
Beaches	60.41	20.56	49.70	23.59	30.01	12.68	278.16	87.84
Buildings	70.62	24.80	59.80	28.85	32.66	17.09	281.08	120.26
Buses	93.89	14.79	90.60	18.63	68.88	17.08	99.96	71.60
Dinosaurs	100.00	0.00	100.00	0.00	96.81	05.01	52.32	4.72
Elephants	73.13	19.63	58.70	22.86	28.09	06.09	272.99	38.64
Flowers	95.75	08.38	93.40	12.49	62.00	12.27	122.09	62.28
Horses	89.10	14.64	81.90	22.41	50.07	19.77	222.92	117.59
Mountains	62.70	19.75	49.80	23.74	27.73	09.94	296.02	89.78
Food	65.99	22.62	52.70	26.05	28.91	10.00	262.27	69.96
Total	**78.98**	**16.93**	**70.54**	**20.88**	**46.12**	**12.59**	**215.53**	**76.17**

Table 4.6

GWC results in terms of various performance measures

Category Name	**P%**	**R%**	**C_{avg}**
Africans	52.9	33.2	264.9
Beaches	42	26.2	357.7
Buildings	47.8	26.5	332.8
Buses	88.3	65.1	111.6
Dinosaurs	96.2	65	135.3
Elephants	65.9	37	242.6
Flowers	75.5	50.4	194.4
Horses	73	39.5	269.3
Mountains	35.2	20.1	372.4
Food	63.2	43.1	192.7
Total	**64.1**	**40.6**	**247.3**

Table 4.7

GWC+EGA results in terms of various performance measures

Category Name	**P%**	**R%**	**C_{avg}**
Africans	56.90	28.38	329.34
Beaches	43.10	24.75	370.56
Buildings	41.90	23.87	379.87
Buses	87.42	62.41	125.67
Dinosaurs	96.00	83.34	62.28
Elephants	65.91	35.85	258.46
Flowers	89.94	53.14	171.79
Horses	69.17	38.62	291.81
Mountains	36.60	23.82	386.68
Food	62.62	37.62	298.91
Total	**65.01**	**41.18**	**267.53**

In Fig. 4.5, average precision for Corel 1000 image database is drawn as a function of various numbers of retrieved images.

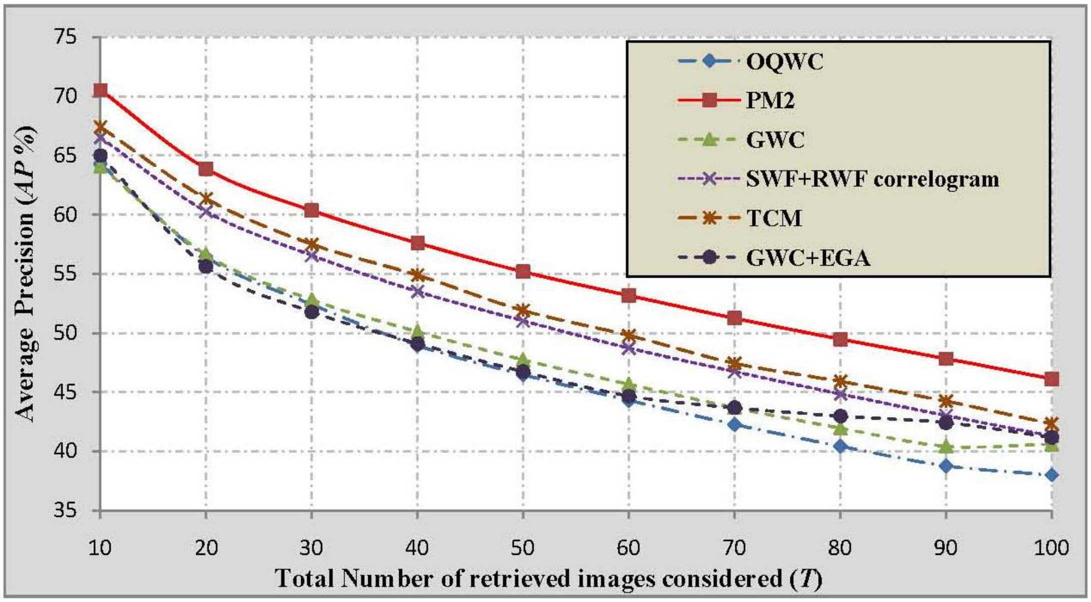


Fig. 4.5: Average precision of PM2 for DB1 database

It is verified that irrespective of the number of retrieved images PM2 always demonstrates better *AP%* as compared to OQWC, GWC, GWC+EGA, SWF+RWF correlogram and TCM. In Fig. 4.6, average retrieval rate (*ARR %*) of PM2 is compared with OQWC, GWC, GWC+EGA, SWF+RWF correlogram and TCM.

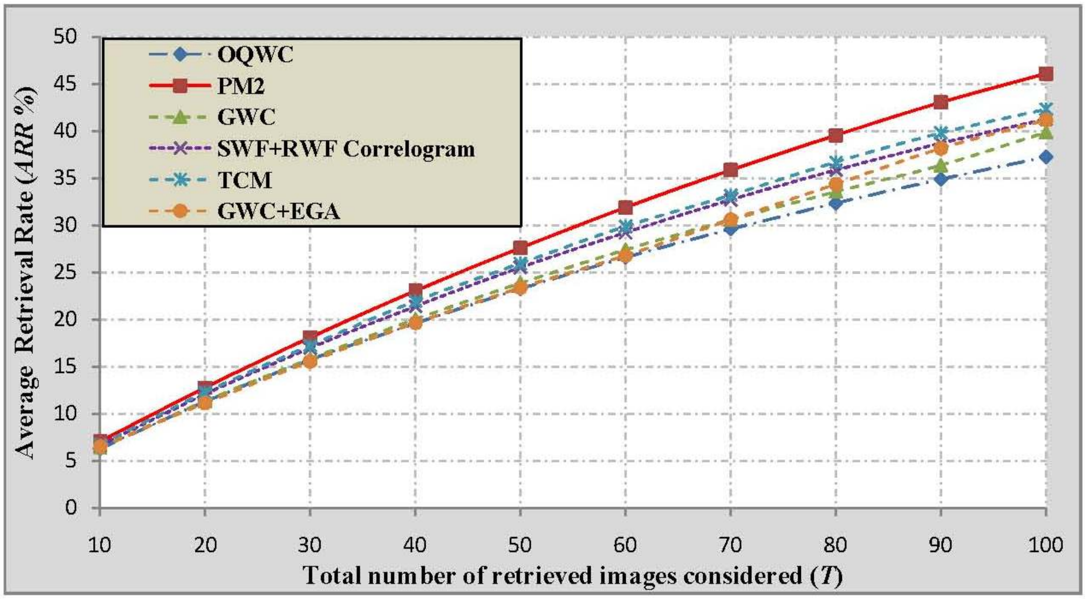


Fig. 4.6: Average retrieval rate of PM2 for DB1 database

It is clear from Fig. 4.6 that the *ARR%* for PM2 is always higher than OQWC, GWC, GWC+EGA, SWF+RWF correlogram and TCM for different number of retrieved images. For 10 and 100 retrieved images, *ARR%* of PM2 is 7.054 and 46.12, for OQWC it is 6.43 and 38, for GWC it is 6.41 and 40.6, for GWC + EGA it is 6.50 and 41.18, for SWF+RWF correlogram it is 6.65 and 41.3 and for TCM it is 6.74 and 42.32 respectively.

Retrieval result of PM2 is also incorporated in Fig. 4.7 for four query images (a) 333, (b) 427, (c) 613 and (d) 789. In each case of Fig. 4.7 first image is the query image and total ten numbers of retrieved images are shown. It is clear from Fig. 4.7 that all retrieved images are relevant to the query images. Thus, better performance of PM2 is justified from retrieval result too.

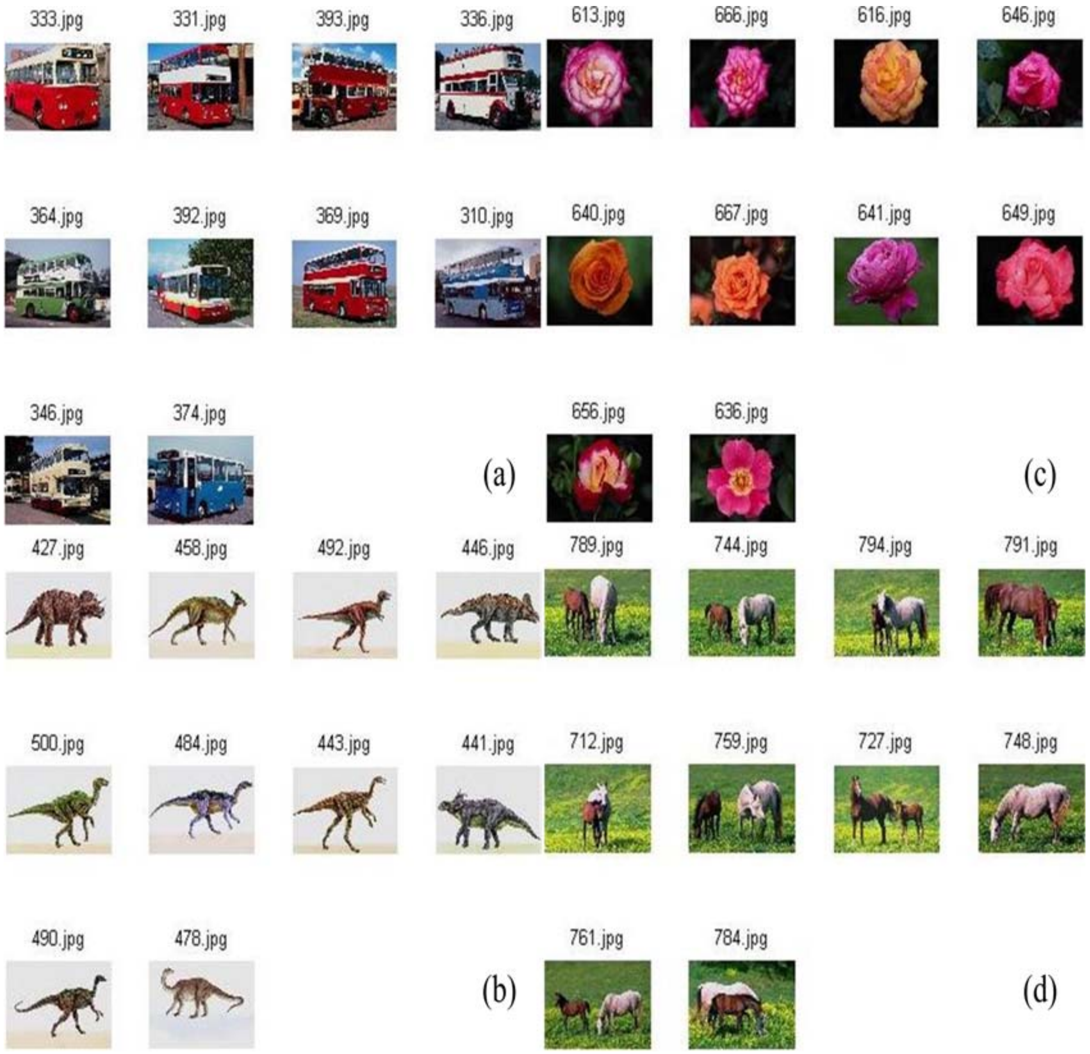


Fig. 4.7: Retrieval results of PM2 for four query images

4.5.2 Corel 2450 Database (DB2)

Next experiment is conducted on DB2 (Appendix A). Average precision is calculated on database DB2 by using Eq. 4.12. In Fig. 4.8 nature of average precision according to number of retrieved images is shown and it is verified that the performance of proposed method PM2, is better in all instances. It is examined that average precision (*AP%*) of PM2 (61.54) is significantly boosted upon OQWC (55.35), SWF+RWF correlogram (59.10) and TCM (57.75) for 10 retrieved images. Hence, it is observed that the proposed method PM2 proves to be the most successful for the image database DB2 also.

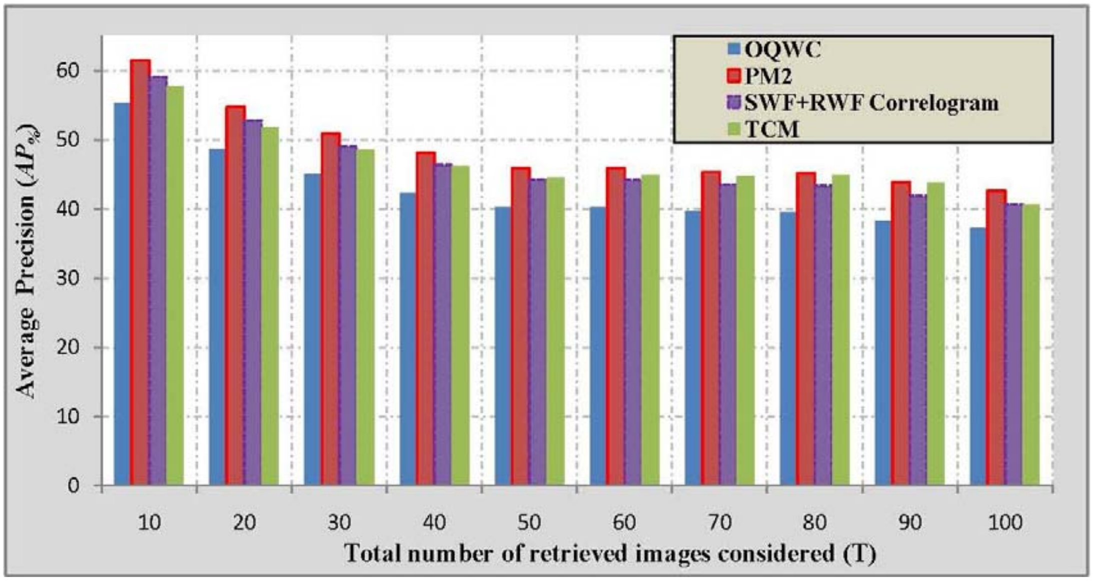


Fig. 4.8: Average precision of PM2 for DB2 database

4.5.3 MIRFLICKR 25000 Database (DB3)

The last experiment is performed on DB3 (Appendix A). On database DB3, better performance of PM2 is justified by average precision calculation through the Eq. 4.12. It is observed from Fig. 4.9 that performance of PM2 is better for various numbers of retrieved images. It is verified that *AP%* of PM2 (25.85) is improved upon OQWC (22.42), SWF+RWF correlogram (23.30) and TCM (23.63) for 10 retrieved images. From the Fig. 4.9, it is inferred that the proposed method PM2 emerges as the most effective and efficient method as compared to the OQWC, SWF+RWF correlogram and TCM even for a huge image database like DB3.

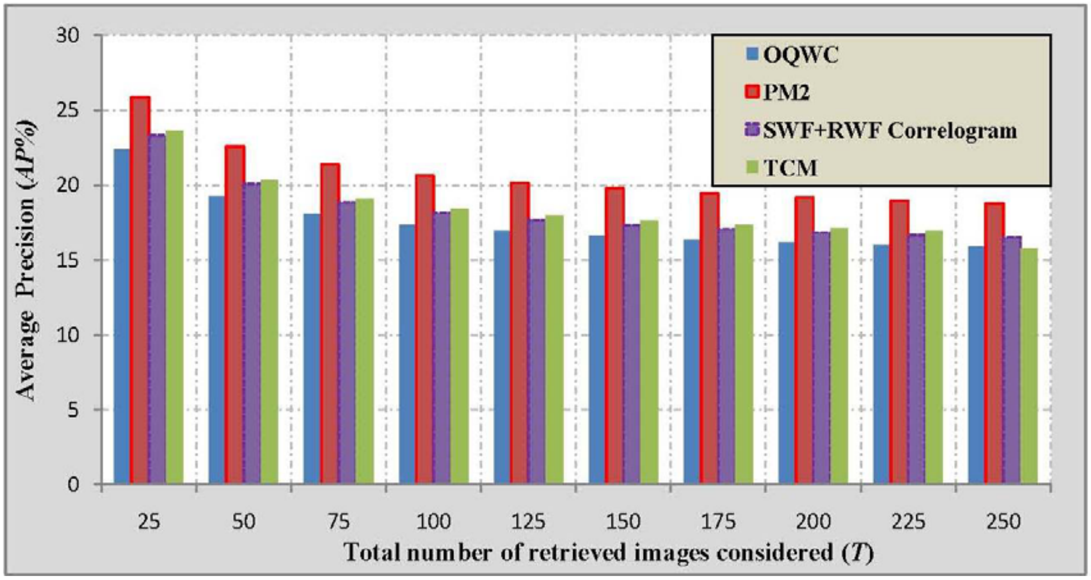


Fig. 4.9: Average precision of PM2 for DB3 database

4.6 SUMMARY

This chapter proposes two methods, namely AWC (PM1) and AGSD (PM2) for CBIR application, where PM2 is the extended version of PM1. PM1 finds the correlation among á trous wavelet coefficients, while in the primitive step of PM2 á trous wavelet coefficients are extracted and further spatial relationship among orientation of á trous wavelet coefficients are utilized. Experiments are performed on three databases viz. Corel 1000, Corel 2450 and especially MIRFLICKR 25000. Performance of proposed method is compared with OQWC, GWC, GWC+EGA, SWF+RWF correlogram and TCM with respect to various parameters. It is concluded that the results with proposed methods are significantly better, irrespective of image databases. Further, the performance of proposed methods can be improved by optimizing the quantization thresholds of á trous wavelet coefficients through any optimization algorithms like genetic algorithm, particle swarm optimization etc.

Even though the present chapter gives very high retrieval performance but still it suffers from the problem of deciding quantization thresholds. This problem is avoided in the next chapter by automatic quantization of coefficients based on the direction of maximum intensity variation.

Cooccurrence of Haar–like Wavelet Filters

Chapter 5

In the previous chapter algorithms presented were based on á trous transform. Although the results are better than what obtained from the work already reported in this thesis as well as available from the literature. But the algorithms developed suffer with the problem of quantization. In the present chapter further attempts are made to bridge this limitation.

An image is a layout of different intensity values. Average intensities within a neighborhood possess good amount of information for texture detection. Since, Haar–like wavelet filters represent an image by a set of regions and capture average intensities within the neighboring regions. Thus, in the work reported in this chapter Haar–like wavelet filters are used for texture feature extraction. The sensitivity to noise and illumination changes is also reduced by working on average intensities of the regions. A large set of Haar–like wavelet filters consists of rectangles in such a manner to detect intensity variations in different directions. Haar–like wavelet filters provide simple, efficient and better information extraction for image representation.

In this chapter a novel technique; cooccurrence of Haar–like wavelet filters (CHLWF) is presented for content based image retrieval (CBIR). CHLWF obtains correlation among dominant Haar–like wavelet filters to capture structural similarity among images. Selection of dominant Haar–like wavelet filter avoids less prominent directions of intensity variation for feature computation and reduces complexities as well. Most of the correlogram based techniques need to decide quantization thresholds beforehand but CHLWF excludes this as the quantization is performed automatically based on directions of maximum intensity variations. Huang *et al*. [38] has used correlogram for feature extraction. Their technique works on pixels intensities directly and does not consider directional edge information. Therefore, in this edge information is lost. But edges play a vital role in texture detection. CHLWF solves this problem by incorporating Haar–like wavelets for edge response calculation before the correlogram computation.

Performance of proposed feature is justified through three benchmark image databases viz. Corel 1000 (DB1), Brodatz (DB2) and MIT VisTex (DB3) (Appendix A). To establish the robustness of the developed method testing has been done on three different types of image database. Experimental results on database DB1 have demonstrated that average precision and average recall is significantly improved by CHLWF (73.14% and 46.53%) in comparison to optimal quantized wavelet correlogram (OQWC) (64.3% and 38%) [83], Gabor wavelet correlogram (GWC) (64.1% and 40.6%) [91], combination of standard wavelet filter with rotated wavelet filter correlogram (SWF+RWF correlogram) (66.5% and 41.3%) [170] and cross correlogram (CC) (67.40% and 43.21) [38]. Likewise, on DB2 and DB3 databases, average retrieval rate of CHLWF (81.54% and 84.08%) is better than CC (72.64% and 79.16) [38], dual tree complex wavelet transform (DT-CWT) (74.73% and 80.78%) [110], dual-tree rotated complex wavelet transform (DT-RCWT) (71.17% and 75.78%) [110] and DT-CWT + DT-RCWT (77.75% and 82.34%) [110] etc. respectively.

5.1 PROPOSED METHOD

CHLWF extracts most dominant directional filter at pixel level for feature calculation thus it considers only prominent edge responses for feature computation. It provides better image representation with reduced complexities. Huang *et al*. [38] has also used correlogram for feature extraction but their technique works on pixels intensities only and does not consider directional edge information for correlogram calculation hence, edge information is lost. Edges play a vital role in texture detection. CHLWF solves this problem by incorporating Haar–like wavelets for edge response calculation. Adding more to it, CHLWF does not suffer from decision of quantization thresholds which persists in most of correlogram feature. In the following section Haar–like filter sets are briefly discussed and proposed retrieval system architecture is overviewed.

5.1.1 Haar–like wavelet filters

Haar–like wavelet filters signify the existence of certain characteristics in a particular area of the image, such as edges or changes in texture. The basic Haar–like wavelet filter takes into account the adjacent rectangular regions at a specific location and calculates intensity differences between them. For example, a 2-rectangle filter can indicate edges. A large set of Haar–like filters consist of filters with different number of rectangular regions and orientations to highlight diverse texture information of the image [75]. Fig 5.1 comprises a Haar like filter (HLF1) used in proposed feature (CHLWF). In Fig. 5.1 (a), (b), (c) and (d) two rectangles are used while in Fig. 5.1 (e) four rectangles are used. Scalar values inside the rectangles represent weights assigned to average intensities of rectangles. These filter masks provide information in $0°, 90°, 135°, 45°$ and $\pm 45°$ respectively.

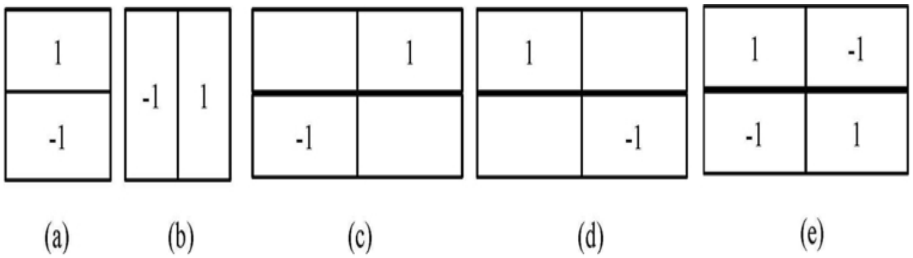


Fig. 5.1: Haar–like filter set 1 (HLF1)

Fig. 5.2 illustrates another set of Haar–like filters viz., HLF2 (first order). It is derived from HLF1 by considering the disjoint rectangles. HLF1 is capable of extracting relationship among close regions but HLF2 can capture relation between distant regions.

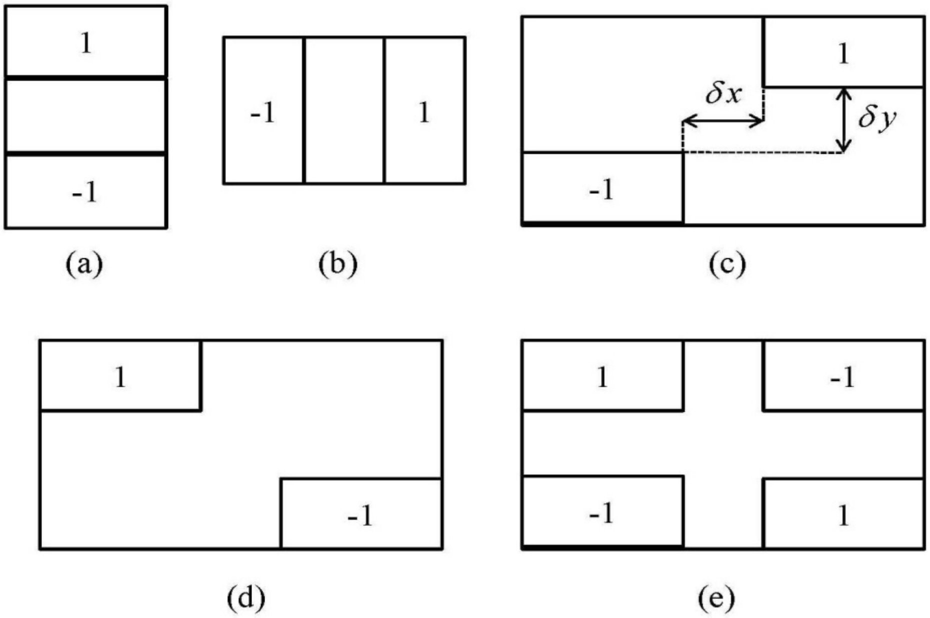


Fig. 5.2: Haar–like filter set 2 (HLF2)

Fig. 5.3 incorporates respective response of HLF2 filters for frontal face image available at [181]. Face image is considered for response representation as it encompasses more visual information for illustration. First row of Fig. 5.3 has original face image considered while second row of Fig. 5.3 contributes responses in $0^\circ, 90^\circ, 135^\circ, 45^\circ$ and $\pm 45^\circ$ (left to right). It is observed in Fig. 5.3 that HLF2 is capable of extracting good directional edge information from images. Fig. 5.4 describes the second order responses for HLF2.

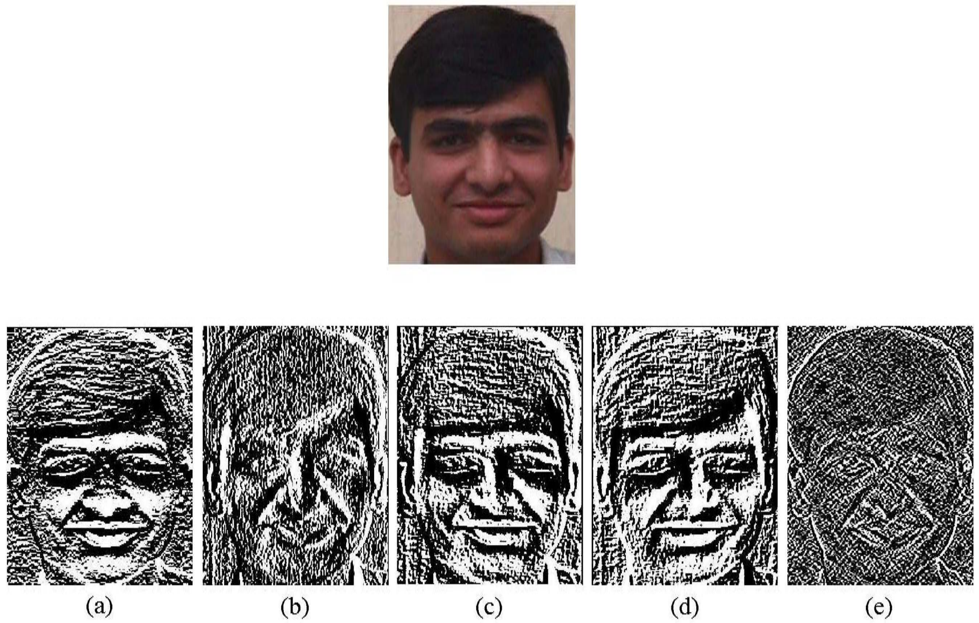


Fig. 5.3: Filter responses of HLF2 first order filter

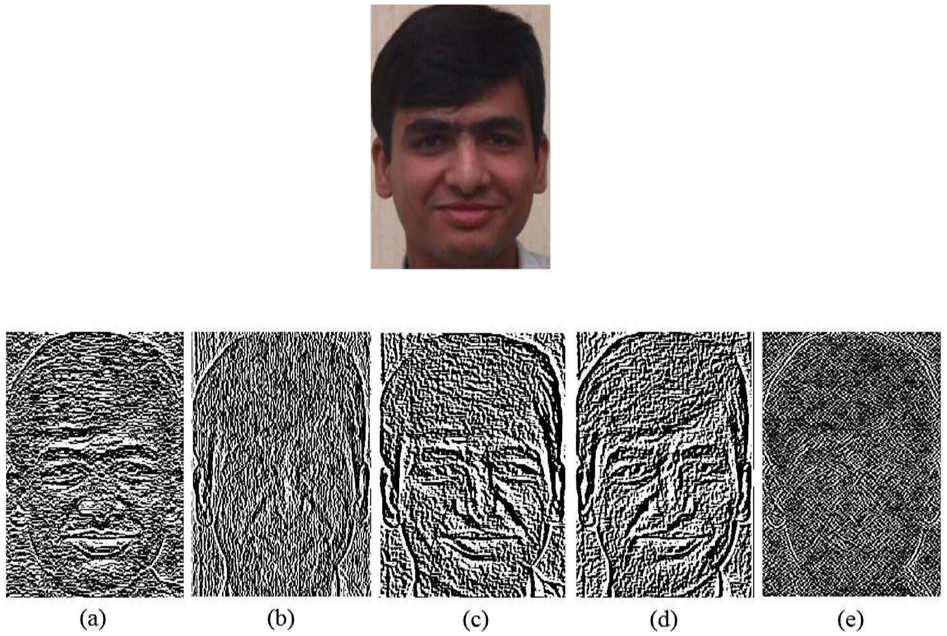


Fig. 5.4: Filter responses of HLF2 second order filter

It is clear from Fig. 5.4 that second order responses provide fine directional edge information. So, both 1st order and 2nd order filter responses of HLF1 and HLF2 filter sets are enclosed for proposed feature computation. Size of blocks used in the implementation of Fig. 5.1 and Fig. 5.2 is 1×2 or 2×1

5.1.2 Cooccurrence of Haar–like wavelet filters (CHLWF)

Let k^{th} filter mask $w|_k$ is used for computation of filter response $g(x,y)|_k$ on image $I(x,y)$ by using *2D* convolution through Eq. (5.1)

$$g(x,y)\big|_k = w(x,y)\big|_k * I(x,y) = \frac{1}{ST}\sum_{s=0}^{S-1}\sum_{t=0}^{T-1} w(s,t)\big|_k \, I(x-s,y-t) \tag{5.1}$$

$where, x=0,1,.....,S-1, y=0,1,........,T-1, k=1,........,5, SXT\ is\ the\ size\ of\ window$

Response for each filter set is given by Eq. (5.2)

$$G(x,y) = \left[g(x,y)\big|_1, g(x,y)\big|_2, g(x,y)\big|_3, g(x,y)\big|_4, g(x,y)\big|_5 \right] \tag{5.2}$$

The next step is construction of dominant filters image (*DFI*) by selecting the filter giving maximal response. This reduces the consideration of non-prominent responses. *DFI* is of same size as the input image and is given by Eq. (5.3).

$$DFI(x,y) = \arg_k\left(\max\left(G(x,y)\right)\right) \tag{5.3}$$

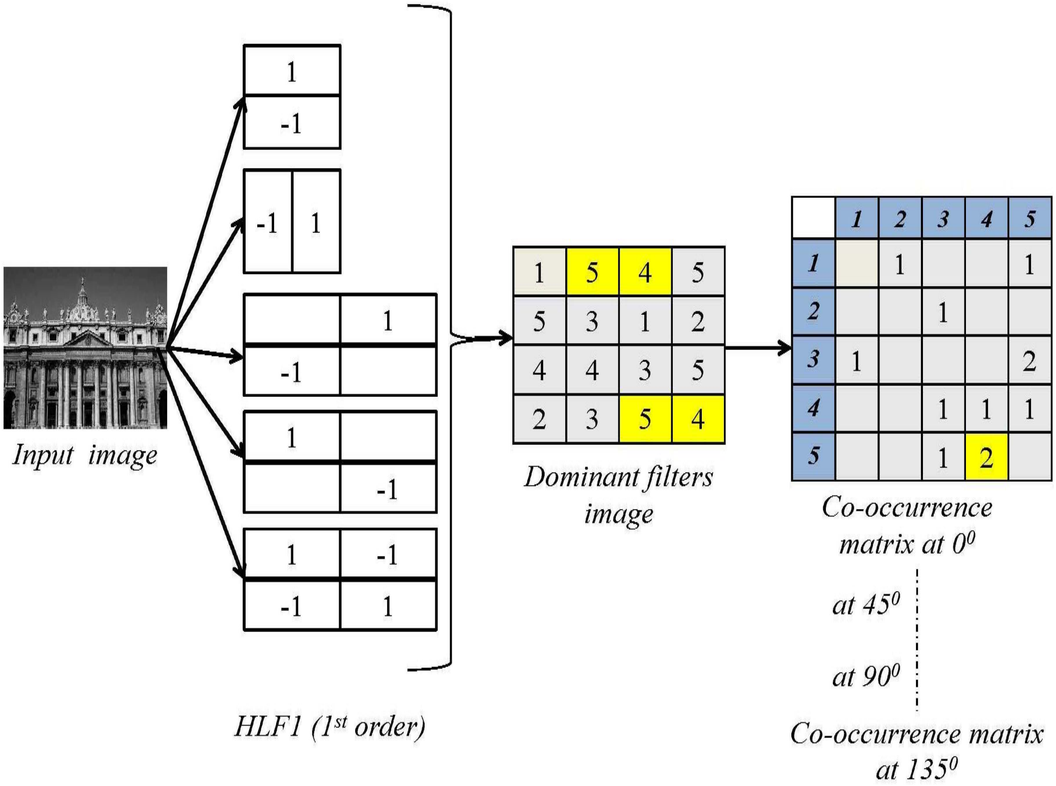
Let $DFI(x_1,y_1)=v_i$ and $DFI(x_2,y_2)=v_j\ ;v_i,v_j \in 1,........5$ then, probability of occurrence of v_i and v_j at distance *d*, is extracted for proposed feature vector (CHLWF) computation by using Eq. (5.4).

$$CHLWF^{(m,n)}_{(\theta,d)}(v_i,v_j) = \Pr\left[\left(v_i \Lambda v_j\right)\big| d\right] \tag{5.4}$$

Where, $d=\max\left(|x_1-x_2|,|y_1-y_2|\right)$, $m(=1,2)$ represents filter sets, $n(=1,2)$ represents filter orders and $\theta(=0°,45°,90°,135°)$ represents direction for cooccurrence computation. In the proposed feature implementation $d=1$ is used. Cooccurrence matrixes obtained from HLF1 (1st& 2nd order) and HLF2 (1st& 2nd order) together construct the final CHLWF feature vector as given by Eq. (5.5).

$$CHLWF = \left[CHLWF^{(1,1)}, CHLWF^{(1,2)}, CHLWF^{(2,1)}, CHLWF^{(2,2)}\right] \tag{5.5}$$

Design strategy for CHLWF extraction is depicted in Fig. 5.5 with HLF1. For CHLWF extraction, after the completion of HLF1 and HLF2 response calculations, filters which give maximum edge response are selected from each filter set separately. As the second order statistics have the ability to provide fine edge distribution in an image thus, in CHLWF computation second order statistics is also used. Cooccurrence is applied on dominant filters image for feature extraction. For illustration, cooccurrence of 5 and 4 in *DFI* is highlighted in Fig. 5.5. 5 and 4 are present twice at $\theta = 0°$ and $d=1$ so, in 0° cooccurrence matrix value at position (5, 4) is incremented by two. Likewise, cooccurrence matrix is computed in four neighboring directions.



1	5	4	5
5	3	1	2
4	4	3	5
2	3	5	4

	1	*2*	*3*	*4*	*5*
1		1			1
2			1		
3	1				2
4			1	1	1
5			1	2	

Fig. 5.5: CHLWF architecture

Algorithmic representation of CHLWF computation is as follows:

Algorithm:

Input: Image

Output: Proposed feature vector corresponding to image

1. Input image and convert it into gray scale
2. Convolve the image with HLF1 (1^{st} order)
3. Responses of each filtered image are normalized

4. For each pixel obtain the dominant filter
5. Apply cooccurrence matrix on dominant filters image with directions, θ and distance, *d*
6. Similarly, obtain cooccurrence matrix for *2nd* order response of HLF1.
7. Construct the final feature vector (CHLWF) by combining cooccurrence matrixes, extracted from HLF1 and HLF2.

5.2 SIMILARITY METRIC

Once the features are extracted the next step of image retrieval process is to perform similarity comparison among query image and database images based on extracted features. A query image can be any image selected by user from the image database. In the experiments conducted each image is considered as query for analyzing average result. The goal is to retrieve images relevant to the query image. Distance measure d_1 given by Eq. (5.6) is used to compute the difference between query and database image features [180].

$$D(Q,T_j)=\sum_{i=1}^{\lambda}\frac{\left|Q_i-T_{ji}\right|}{\left|1+Q_i+T_{ji}\right|} \tag{5.6}$$

Where, Q_i is i^{th} feature component of the query image, T_i is i^{th} feature component of j^{th} database image, λ is feature vector length. Database images $\left(j=1,2,.......,DB\right)$ having small $D(Q,T_j)$ are considered to be more relevant to the query image and a list of relevant images is retrieved.

5.3 EXPERIMENTAL RESULTS AND DISCUSSIONS

Effectiveness of proposed feature is established by performing image retrieval on three standard image databases, Corel 1000 (DB1), Brodatz texture image database (DB2) and MIT VisTex texture database (DB3) namely (Appendix A). The variety in the contents of these databases enhances their utility among researchers for various scientific articles of CBIR. Retrieval performance of proposed method on these databases is compared with existing and related techniques [38, 83, 90, 91, 170, 110 and 182] with respect to various performance measures. All the experiments are performed on gray version of images.

In the implementation of CC [38] input image is converted into gray scale and intensity values are quantized into 32 levels. CC is computed in four directions with unit distance. In OQWC computation, DWT of the input image is computed upto three scales and obtained wavelet coefficients are quantized to a limited number of levels by the application of optimization algorithm [83]. Then, autocorrelogram on vertical and horizontal wavelet coefficients is computed. GWT [90] solves the problem of diagonal mixing in DWT. GWT texture features are extracted by mean and standard deviation computation on four scales and six directional subbands. GWC computes the wavelet coefficients by using Gabor wavelets at three scales and four directions [91]. The coefficients so obtained are quantized and autocorrelogram on the quantized coefficients is computed along the direction normal to Gabor wavelet orientation $\left(0°, 45°, 90°, 135°\right)$. Subrahmanyam *et al.* [170] proposed the combination of SWF and RWF to collect the directional information in four directional $\left(0°,\ +45°,\ 90°\ \text{and}\ \text{-}45°\right)$ from the image and further used it for correlogram feature calculation. To accomplish this, $0°$ and $90°$ subbands information of the image are collected from SWF and $+45°$ and $\text{-}45°$ are collected from RWF. Kokare *et al*. [110] introduced DT-CWT and DT-RCWT and its combination also to obtain more directional information and construct texture feature vector by using energy and standard deviation of respective transformed subbands. Do *et al*. [182] proposed wavelet based texture image retrieval using generalized Gaussian density and Kullback-Leibler distance (GGD &KLD) method. The results obtained are benchmarked with respect to the latest available literature on the texture feature.

The retrieval result is not a single image but it is a list of images dependent upon the relevancy to the query image. It is desired to have all retrieved images to be relevant i.e. all retrieved images should belong to query image group only. *T* gives the total number of retrieved images considered $(e.g. 10, 20, \ldots\ldots, 100)$. Value of *T* is selected by user. All images of the database are considered as query image to evaluate average performance on the image database. Performance measures like average precision (*AP*), average recall (*AR*) and average retrieval rate (*ARR*), used to approve the performance of proposed method are given as follows:

For i^{th} image precision is given by Eq. (5.7).

$$Precision\,(P_i) = \frac{No.of\ \ relevant\,images\,retrieved}{Total\,no.of\ \ retrieved\,images\,considered\,(T)} \tag{5.7}$$

$$Group\ precision\,(GP) = \frac{1}{N}\sum_{i=1}^{N} P_i \tag{5.8}$$

$$Average\ precision\,(AP) = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} GP_i \tag{5.9}$$

Similarly performance measures are computed in the case of recall by following Eq.

$$Recall\,(R_i) = \frac{No.of\ relevant\ images\ retrieved}{Total\ no.of\ relevant\ images\ in\ database(N)} \tag{5.10}$$

$$Group\ Recall\,(GR) = \frac{1}{N}\sum_{i=1}^{N} R_i \tag{5.11}$$

$$Average\ Recall\,(AR) = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} GR_i \tag{5.12}$$

$$Average\ Retrieval\ Rate\,(ARR_T) = \frac{1}{DB}\sum_{j=1}^{DB} R_j \Bigg|_{T \le 100} \tag{5.13}$$

Where, N is number of relevant images in the database, Γ is the number of groups and DB is the number of database images. T gives the number of image retrieved. In experiments for DB1, $T \ge 10$ while for DB2 and DB3, $T \ge 16$ is considered. Experimental analysis on image databases DB1, DB2 and DB3 based on above parameters are discussed in the following sections.

5.3.1 Corel 1000 Database (DB1)

Precision, recall and average retrieval rate were evaluated on DB1 for performance analysis of proposed method with respect to other existing features by using Eq. (5.7) to Eq. (5.13). Table 5.1 demonstrates average precision (*T=10*) and average recall (*T=100*) for each group and total database in the case of CHLWF, OQWC [83], GWC [91], SWF+RWF correlogram [170] and CC [38]. To examine the performance, with respect to various numbers of retrieved images, the behavior of average precision and average retrieval rate are drawn in Fig. 5.6 and Fig. 5.7 respectively on database DB1.

Table 5.1

Group precision and average precision comparison for database DB1

Category Name	CHLWF		OQWC		GWC		SWF+RWF correlogram		CC	
	P% (T=10)	R% (T=100)	P% (T=10)	R% (T=100)	P% (T=10)	R% (T=100)	P% (T=10)	R% (T=100)	P% (T=10)	R% (T=100)
Africans	61.8	37.45	57.7	31.1	52.9	33.2	64.9	34.7	81.8	42.1
Beaches	56.3	32.63	49.3	28.6	42	26.2	43.5	27.8	38	21.41
Buildings	68.7	34.74	50.9	30.5	47.8	26.5	52.4	31.1	56	32.21
Buses	97.1	77.75	87.1	64	88.3	65.1	90.3	69.5	76.4	56.5
Dinosaurs	96.4	75.6	74.6	28.8	96.2	65	81.2	45.9	99.3	89.89
Elephants	60.8	31.72	55.7	30.7	65.9	37	58.4	33.7	61.7	34.38
Flowers	86.4	64.35	84.3	65.3	75.5	50.4	86.7	58.5	94.4	77.43
Horses	81	40.52	78.9	39.9	73	39.5	80	47.2	71.6	28.45
Mountains	43.9	25.68	47.2	25.1	35.2	20.1	37.5	20	30.8	17.49
Food	79	44.89	57.1	36.4	63.2	43.1	67.8	44.1	64	32.31
Average	**73.14**	**46.53**	**64.3**	**38**	**64.1**	**40.6**	**66.5**	**41.3**	**67.40**	**43.21**

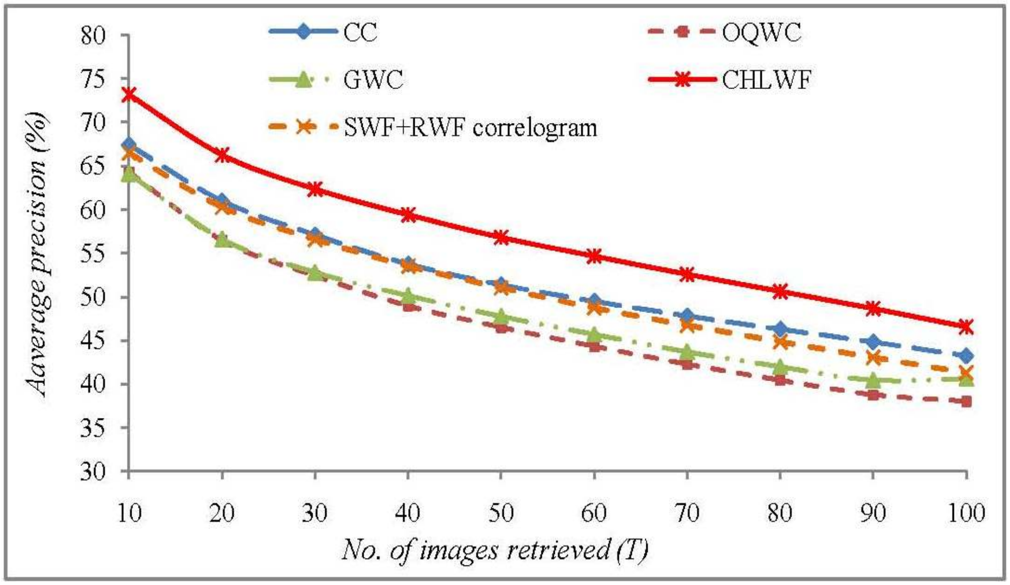


Fig. 5.6: Average precision comparison on image database DB1

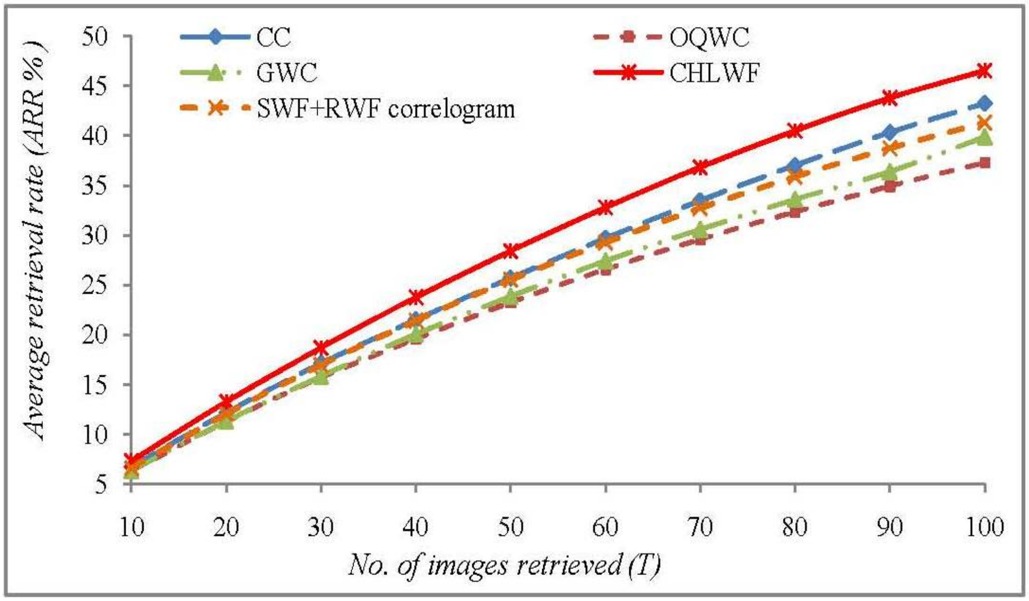


Fig. 5.7: Average retrieval rate (*ARR %*) comparison on image database DB1

Through Table 5.1, Fig. 5.6 and Fig. 5.7 following points are observed.

1. It is evident through Table 5.1 and Fig. 5.6 that CHLWF has improved average precision on the database DB1 (73.14%) in comparison to OQWC (64.3%), GWC (64.1%), SWF+RWF correlogram (66.5%) and CC (67.40%).
2. The average recall on the database DB1 is also appreciably improved by using CHLWF (46.53%) as compared to OQWC (38%), GWC (40.6%), SWF+RWF correlogram (41.3%) and CC (43.21%).
3. It is observed in Fig. 5.7 that CHLWF shows enhanced average retrieval rate as compared to OQWC, GWC, SWF+RWF correlogram and CC for all instances of retrieved images.

Fig. 5.8 explicates four queries (a) 206, (b) 683, (c) 794, (d) 927 retrieval results based on CHLWF on database DB1 for 10 retrieved images. It is apparent that all retrieved images belong to query image group only i.e. all relevant images are retrieved (Upper left most image is the query image in all cases).

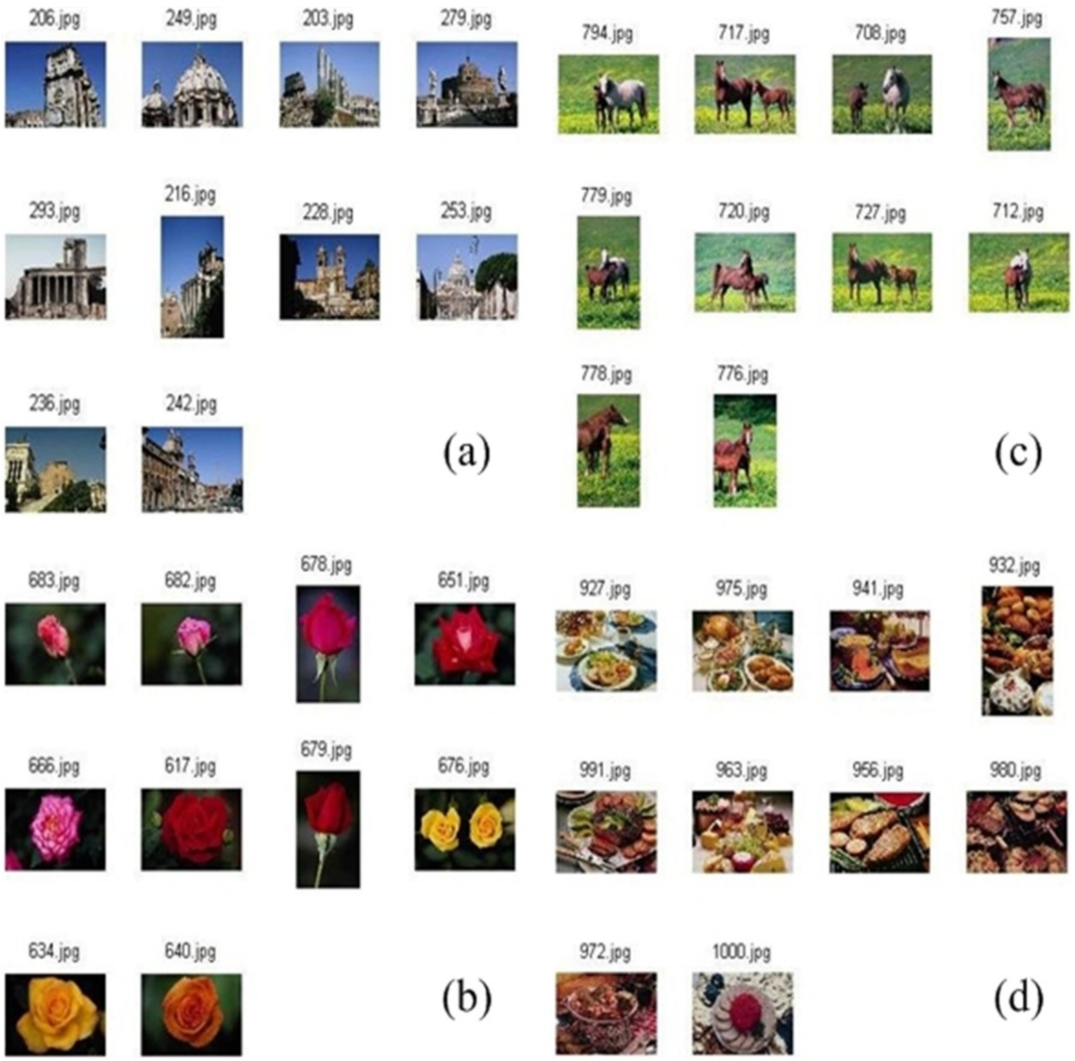


Fig. 5.8: Retrieval results of CHLWF on database DB1

5.3.2 Brodatz Database (DB2)

Next experiment is conducted on DB2; a texture image database (Appendix A). Performance of proposed feature (CHLWF) is compared with [38, 90 and 110] in terms of *ARR*%. Top-16 *ARR*% is defined as the number of relevant images retrieved out of the first 16 retrieved images and is given by Eq. (5.13). Table 5.2 illustrates top-16 *ARR*% on the image database DB2. For top-16 retrieved images, CHLWF yields 81.54% *ARR* while on same number of retrieved images CC, GWT, DT-CWT, DT-RCWT and DT-CWT+ DT-RCWT yields 72.64%, 74.32%, 74.73%, 71.17% and 77.75% respectively. Fig. 5.9 depicts the trend of *ARR%* for different number of retrieved images and it is verified from

Table 5.2

Top-16 average retrieval rate (*ARR %*) for database DB2

Features	CHLWF	CC	GWT	DT-CWT	DT-RCWT	DT-CWT+ DT-RCWT
ARR (%)	81.54	72.64	74.32	74.73	71.17	77.75

Fig. 5.9 that the proposed feature always gives significantly better performance compared to other existing methods.

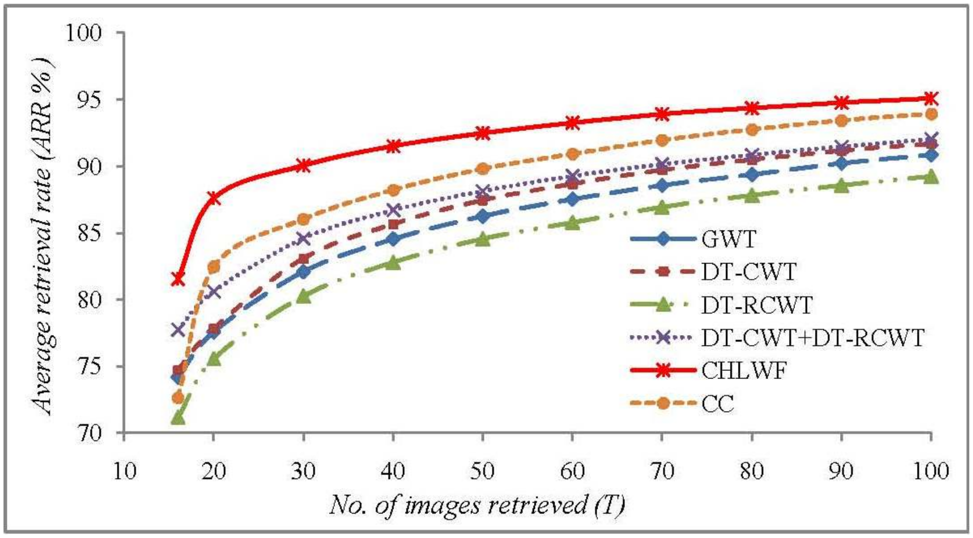


Fig. 5.9: Average retrieval rate (*ARR %*) comparison on database DB2

To demonstrate the retrieval results four query images (a) 161, (b) 650 (c) 1009, (d) 1601 are selected from the image database DB2. Fig. 5.10 manifests the retrieval result of proposed method for 10 retrieved images (Upper left most image is query image in all cases). It is observed in Fig. 5.10 that in all cases CHLWF retrieves the relevant images. Thus CHLWF proves its effectiveness by performing superior in the case of texture images as well.

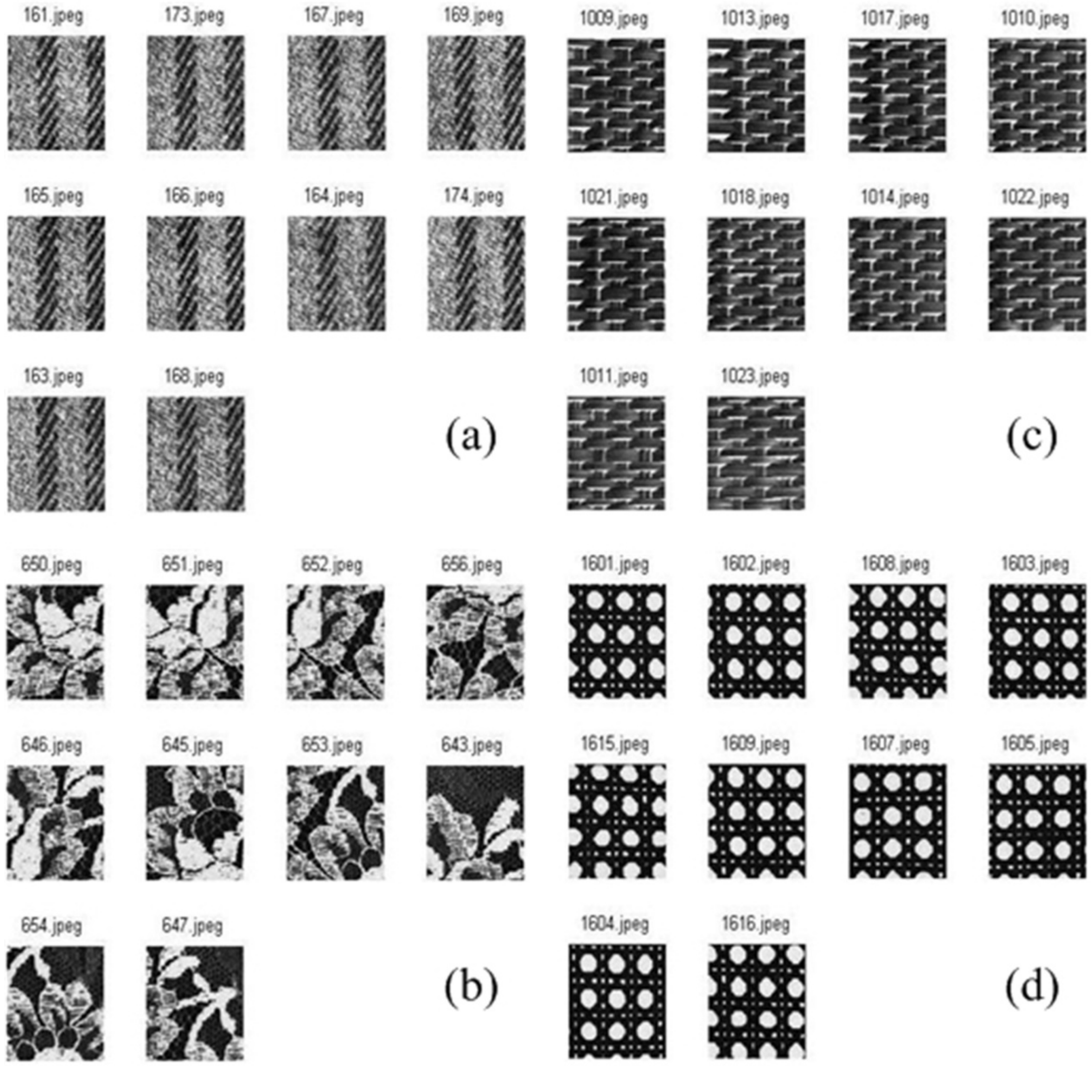


Fig. 5.10: Retrieval results of CHLWF on database DB2

5.3.3 MIT VisTex Database (DB3)

Performance of proposed method is verified on one more texture database; DB3 (Appendix A). DB3 is a color texture database. Feature calculation on DB3 is performed by converting the color images into gray scales. Retrieval performance on this database is compared in terms of *ARR%* as given by Eq. (5.13). Table 5.3 comprises the top-16 *ARR%* of proposed method as well of related existing methods [180, 38 and 110] on database DB3. It is apparent from Table 5.3 that *ARR* of CHLWF (84.08%) is magnificently improved upon GGD & KLD (76.57%), CC (79.16%), DT-CWT (80.78%), DT-RCWT (75.78%) and DT-CWT+ DT-RCWT (82.34%).

Table 5.3

Top-16 average retrieval rate (*ARR %*) for database DB3

Features	CHLWF	GGD & KLD	CC	DT-CWT	DT-RCWT	DT-CWT+ DT-RCWT
ARR (%)	84.08	76.57	79.16	80.78	75.78	82.34

Fig. 5.11 incorporates *ARR%* comparison between proposed method and existing methods for varying number of retrieved images and superior performance of CHLWF is proven over GWT, SWF+RWF correlogram, CC, DT-CWT, DT-RCWT and DT-CWT+ DT-RCWT all cases of retrieved images. Fig. 5.12 comprises image retrieval results of CHLWF on database DB3 in the case 10 retrieved images. Four query images; (a) 15, (b) 193, (c) 325 and (d) 577 are considered for the illustration of retrieval results (Upper left most image is the query image in all cases). It is desired to obtain all relevant images in the retrieval result and it is noticed from Fig. 5.12 that the proposed method retrieves all relevant images. Hence, it is experimented that proposed method, CHLWF performs superior not only in the case of natural images but also on texture images.

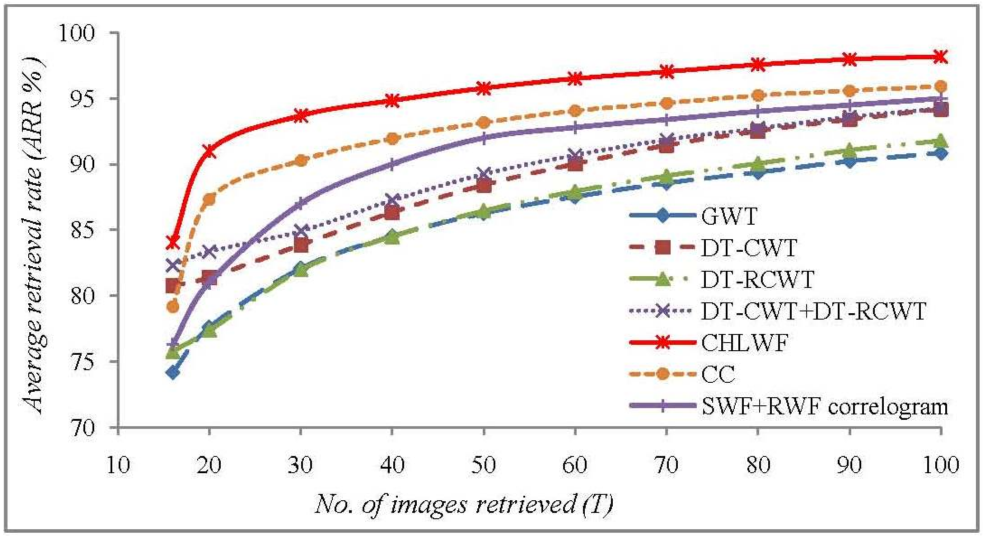


Fig. 5.11: Average retrieval rate (*ARR %*) comparison on database DB3

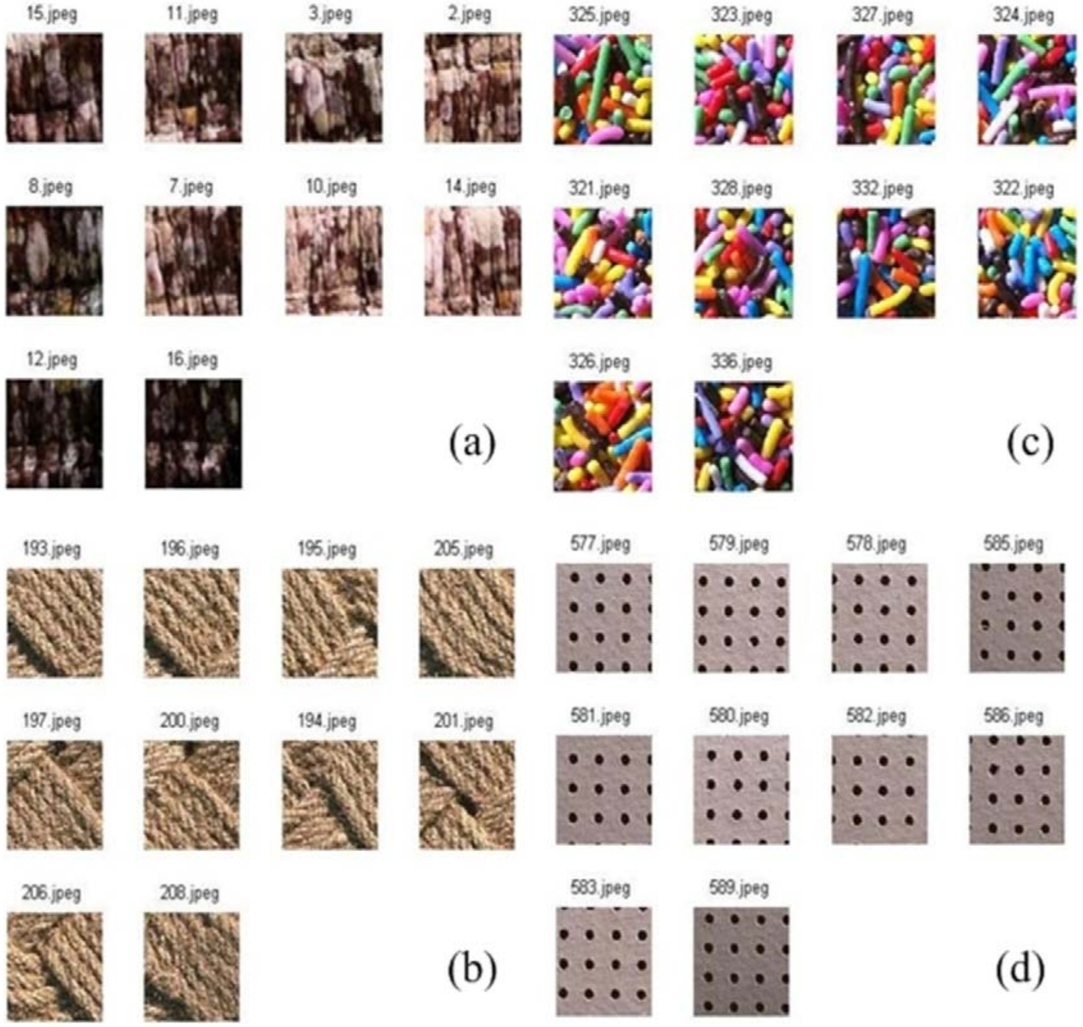


Fig. 5.12: Retrieval results of CHLWF on database DB3

5.4 SUMMARY

This chapter establishes the superiority of CHLWF in the case of image retrieval. CHLWF extracts edge responses by applying two types of Haar–like wavelet filters in various directions. The responses are compared to compute the most effective filter. Probability of most effective filter in four directions is used to compute the CHLWF. CHLWF operates on average intensities; it allows avoiding noise and illumination changes. CHLWF employs only maximal edge responses in feature computation hence, only prominent directional information which substantially represents an image is used. Complexity in quantization of coefficients is also avoided in CHLWF. All these properties assist to show the improved retrieval performance of CHLWF among related correlogram techniques. Main aim of introducing CHLWF is to improve effectiveness of retrieval system and it is affirmed by evaluating the retrieval performance on three benchmark image databases Corel 1000, Brodatz and MIT VisTex. It is evident that the results of CHLWF are remarkably superior to CC, OQWC, GWC, SWF+RWF correlogram, GGD & KLD, DT-CWT, DT-RCWT, DT-CWT+ DT-RCWT and GWT on all the image databases with respect to various performance measures. In further work, number, size and orientations of Haar–like filters can be optimized to enhance the efficacy of retrieval system.

The present chapter gives significant improvement in retrieval results. The analysis presented in Section 1.2.3 shows that the retrieval performance can be increased by integrating the color and texture features. Hence, in the next chapter an attempt has been made to propose integrated feature (color and texture) and the developed algorithm further enhances the retrieval results.

Weight Cooccurrence Based Integrated Color and Intensity Matrix

Chapter 6

The previous works reported in this thesis show improvement in retrieval performance but still there is way to further enhance the retrieval performance by integrating the features. Integration of features inhabits the properties of different features. It is done to improve image feature representation thereby enhancing the retrieval performance. Usually a feature captures one visual perception and the other perception is ignored. But to represent an image in a more meaningful manner it is required to extract both color as well as intensity information.

In this chapter a new technique; weight cooccurrence based integrated color and intensity matrix (WCICIM) is proposed for CBIR. The proposed image indexing technique introduces a feature that utilizes color as well as texture properties of the image. HSV (Hue, saturation and intensity value) color space has a good representation to extract color and intensity information simultaneously. In this chapter, this advantage of HSV space is utilized for calculating the color and intensity weights of each pixel. Weight matrixes are constructed by applying the weight function so as to decide the contribution of color as well intensity for each pixel. Final WCICI matrix is constructed as the combination of four correlation matrixes (color–color, color–intensity, intensity–color and intensity–intensity) based on neighboring pixel variations in weight matrixes with the help of quantized HSV space.

Integrated color and intensity cooccurrence matrix (ICICM) based image retrieval technique does not consider the weight correlation of color and intensity values. Thus, WCICIM and ICICM features are combined together to perform the task of image retrieval. It is experimented that the proposed feature with the combination of ICICM feature improves the retrieval performance significantly. The results of proposed method are compared with related CBIR techniques such as motif cooccurrence matrix (MCM) [55], color correlogram (CC) [38], combination of block bit plane with global color histogram (GCH+BBP) [175] and ICICM [119] on standard Corel 1000 and Corel 2450 image databases (Appendix A). Out of which MCM and CC consider color information but gray intensity values are not extracted. Hence, it is verified that by integrating gray intensity information with color, the proposed method exhibits better retrieval performance as compared to other existing methods on the basis of various performance measures.

6.1 PROPOSED METHOD

In this chapter, weight cooccurrence based integrated color and intensity matrix (WCICIM) is proposed for feature extraction. It is observed that WCICIM with the combination of ICICM performs better than various related features. To extract the color and intensity information, hue and intensity value components of the HSV color space are utilized.

ICICM captures color and intensity variations around each pixel [119]. For each pair of neighboring pixels, contribution of both color as well as gray level is considered. HSV color space has the property to represent color and intensity perception at the same time. This useful property of the HSV color space is utilized for extracting the ICICM and WCICIM features. Next subsection briefly explains relevant properties of the HSV color space. In the subsequent subsection, application of HSV space for ICICM and WCICIM construction is explained.

6.1.1 HSV color space

Various color spaces are available to represent and specify colors in a way suitable for processing of color information in images [11]. Out of these, HSV is a color space that separate out the luminance component (Intensity) of a color pixel from its chrominance components (Hue and Saturation). HSV color model is usually represented by different shapes e.g. hexagonal, triangular and circular shapes. Humans describe a color object by its hue, saturation and brightness. Hue is a color attribute that describes a pure color, whereas saturation gives a measure of the degree to which a pure color is diluted by white light and brightness is the achromatic notion of intensity. At low levels of illumination, only the rod cells of retina are excited so that only gray shades are perceived. As the illumination level increases, more and more cone cells are excited, resulting in increased color perception. There is no fixed threshold for the saturation of cone cells that leads to loss of chromatic information at higher levels of illumination caused by color dilution. Therefore, suitable weights are used such that they vary smoothly with saturation and intensity to represent both color and gray scale perception of each color pixel.

6.1.1.1 Color and intensity weight matrix

From HSV color image, color weight matrix is calculated to consider the weight of color at each pixel by using the Eq. (6.1).

$$W_C(S,I)=\begin{Bmatrix} S^{a\times\left(255/I\right)^b} & for\ I\neq 0 \\ 0 & for\ I=0 \end{Bmatrix} \tag{6.1}$$

Where, *S* and *I* are the saturation and intensity components of HSV color space respectively. *a (=0.1)* and *b (=0.85)* are the optimized constants [119]. Color weight matrix and intensity weight matrixes are related by Eq. (6.2).

$$W_I(S,I)=1-W_C(S,I) \tag{6.2}$$

6.1.1.2 Color and intensity quantization matrix

Hue and intensity components of HSV space are quantized into *HL, GL* levels and *CQ, IQ* quantized color and intensity matrixes are obtained respectively.

Following section first explains about construction of ICICM feature, further WCICIM is proposed.

6.1.2 Integrated color and intensity cooccurrence matrix (ICICM)

From quantized color and intensity matrixes, ICICM is constructed as a combination of four matrixes, represented by the following Eq. (6.3)

$$ICICM=\begin{pmatrix} ICICM_{CC} & ICICM_{CI} \\ ICICM_{IC} & ICICM_{II} \end{pmatrix} \tag{6.3}$$

Here, *ICICM*$_{CC}$ represents the color perception of pixel *p* and color perception of its neighbor pixel *N*$_p$. *ICICM*$_{CI}$ represents the color perception of pixel *p* and gray level perception of its neighbor pixel *N*$_p$. Similarly, *ICICM*$_{IC}$ and *ICICM*$_{II}$ are defined. Each component of *ICICM*$_{CC,}$ *ICICM*$_{CI}$, *ICICM*$_{IC}$ and *ICICM*$_{II}$ can be defined by Eq. (6.4).

$$\left.\begin{array}{l} icicm_{CC}\left(i,j\right)\big|_{i=0...HL-1;\, j=0,...HL-1} = \Pr\left(\left(hl_p, hl_{N_p}\right) = \left(i,j\right)\right) \\ icicm_{CI}\left(i,j\right)\big|_{i=0...HL-1;\, j=0,...GL-1} = \Pr\left(\left(hl_p, gl_{N_p}\right) = \left(i,j\right)\right) \\ icicm_{IC}\left(i,j\right)\big|_{i=0...GL-1;\, j=0,...HL-1} = \Pr\left(\left(gl_p, hl_{N_p}\right) = \left(i,j\right)\right) \\ icicm_{II}\left(i,j\right)\big|_{i=0...GL-1;\, j=0,...GL-1} = \Pr\left(\left(gl_p, gl_{N_p}\right) = \left(i,j\right)\right) \end{array}\right\} \tag{6.4}$$

Where, *HL* and *GL* are the quantization levels of hue and intensity values. Thus, $icicm_{CC}(i, j)$ represents the number of times the color perception of a pixel *p* denoted by hl_p equals *i*, and the color perception of its neighbor pixel N_p denoted by hl_{Np} equals *j*, as a fraction of the total number of pixels in the image. Similarly, $icicm_{CI}(i, j)$, $icicm_{IC}$ (i, j), $icicm_{II}(i, j)$ can be obtained.

$ICICM_{CC}$ entries are updated by the sum of color weight of *p* and color weight of N_p. $ICICM_{CI}$ entries are updated by the sum of the color weight of *p* and the intensity weight of N_p. $ICICM_{IC}$ and $ICICM_{II}$ entries are updated in the similar manner. ICICM matrix is calculated for four neighborhood orientations $\{0°, 45°, 90°, 135°\}$ and final ICICM features are obtained.

6.1.3 Weight cooccurrence based integrated color and intensity matrix (WCICIM)

This section proposes the WCICIM for image feature representation. It computes the color and texture features simultaneously. Integrated approach captures spatial variations of both color weight and intensity weight levels in the neighborhood of each pixel by using the HSV color space. While the existing schemes generally treat color and intensity separately. In contrast to the other methods, for each pixel and its neighbors, color and intensity variations between them are estimated by using a weight function. WCICIM avoids the use of weights to combine individual color and intensity features. In the subsequent subsection, construction of the WCICIM is described.

6.1.3.1 Color weight quantization matrix

From HSV color image, color weight matrix can be calculated by using the Eq. (6.1). After quantizing the $W_C(S, I)$ matrix into L_1 levels, color weight quantized matrix *WQC* is obtained.

6.1.3.2 Intensity weight quantization matrix

By using color weight matrix, intensity weight matrix can be calculated by using the Eq. (6.2). After quantizing the $W_I(S,I)$ matrix into L_2 levels, intensity weight quantized matrix *WQI* is obtained.

6.1.3.3 WCICIM construction

WCICIM is calculated as a combination of four sub-matrixes constructed to cover the color and texture properties of neighboring pixels by using Eq. (6.5).

$$WCICIM_i = \begin{pmatrix} WCICIM_{CC} & WCICIM_{CI} \\ WCICIM_{IC} & WCICIM_{II} \end{pmatrix} \tag{6.5}$$

Where, $i = \{0°, 45°, 90°, 135°\}$

$$WCICIM_{CC}\left(CQ_{(p)}, CQ_{(N_p)}\right) = P\left(WQC_{(P)} \in WQC \wedge WQC_{(N_p)} \in WQC \wedge D\left(WQC_p, WQC_{(N_p)}\right) = 1\right) \tag{6.6}$$

$$WCICIM_{CI}\left(CQ_{(p)}, IQ_{(N_p)}\right) = P\left(WQC_{(p)} \in WQC \wedge WQI_{(N_p)} \in WQI \wedge D\left(WQC_{(p)}, WQI_{(N_p)}\right) = 1\right) \tag{6.7}$$

$WCICIM_{CC}$ is updated with the correlation between weight of pixel *p* and weight of its neighboring pixel N_p in *WQC* matrix. This correlation value is placed in $WCICIM_{CC}$, indexed by the quantized color values of the corresponding pixel *p* and its neighboring pixel N_p in *CQ*.

$WCICIM_{CI}$ is updated with the correlation between weight of pixel *p* in *WQC* and weight of its neighboring pixel N_p in *WQI* matrixes. This correlation value is placed in $WCICIM_{CI}$ indexed by the quantized values of corresponding pixel *p* and its neighboring pixel N_p in *CQ* and *IQ* matrixes respectively. Likewise, $WCICIM_{IC}$, $WCICIM_{II}$ matrixes can be constructed.

6.2 DISTANCE MEASUREMENT

To perform image retrieval, a query image is taken from the image database. Query image and database images are processed to compute ICICM and WCICIM features. In experiments all images are considered as a query so as to evaluate the average retrieval performance of features on the image database. Distance metric given by Eq. (6.8) is used to compute the mismatch between query image and database image features.

$$D(Q,T) = A\sum_{i=1}^{\lambda_1}\frac{\left|Q_{1i} - T_{1i}\right|^2}{\left|1 + Q_{1i} + T_{1i}\right|^2} + B\sum_{j=1}^{\lambda_2}\frac{\left|Q_{2i} - T_{2i}\right|^2}{\left|1 + Q_{2i} + T_{2i}\right|^2} \qquad (6.8)$$

Where, Q_{1i} and Q_{2i} are WCICIM and ICICM feature vectors of the query image. T_{1i} and T_{2i} are WCICIM and ICICM feature vectors of database image. λ_1 and λ_2 are the feature vectors lengths of WCICIM and ICICM. Constants *A* and *B* are the suitable weights for WCICIM and ICICM features obtained by trial and error method. Database images having small *D (Q, T)* value is more relevant to query image and are retrieved by the system.

6.3 PROPOSED ALGORITHM FOR IMAGE RETRIEVAL

The proposed image retrieval system is based on the combination of ICICM and WCICIM features.

Algorithm for proposed image retrieval system:

1. Input RGB image
2. Transform the RGB image into HSV image
3. Calculate color and intensity weight matrix
4. Calculate color and intensity quantization matrixes
5. Obtain ICICM features
6. Obtain WCICIMfeatures
    a. Quantize color and intensity weight matrix
    b. Construct the final WCICI matrix for four neighborhood directions by using the Eq. (6.5) and arrange like Eq. (6.9).

$$WCICIM = \left[WCICIM_0, WCICIM_{45}, WCICIM_{90}, WCICIM_{135}\right] \qquad (6.9)$$

7. Based on ICICM and WCICIM features obtain the final image retrieval results.

6.4 EXPERIMENTAL RESULTS AND DISCUSSIONS

To demonstrate the performance of proposed method experiments are performed on Corel 1000 and Corel 2450 standard image databases (Appendix A) [167]. In the implementation of proposed method HL , GL , L_1 and L_2 are obtained by trial and error method and their values are considered as 10, 6, 25 and 25 respectively. For performance evaluation average precision (*AP%)*, average recall (*AR%*) and average retrieval rate (*ARR%*) are calculated by using Eq. (6.12), (6.15) and (6.16) respectively. Based on these evaluation measures, performance of proposed method is improved upon various existing methods like CC [38], MCM [55], GCH+BBP [175] and ICICM [119]. Color correlogram characterizes not only the color distributions of pixels, but also the spatial correlation between pairs of colors. It is implemented by quantizing the each color component into 32 levels and then auto correlogram is calculated in four directions with unit distance. Texture features collected from MCM work on differences in image intensity. It converts each component of color image into basic graphics called motif patterns and further finds the probability of occurrence of such seven types of patterns in the adjacent neighborhoods as the image feature. These features consider only color pixel information and intensity information is lost completely, while the proposed method considers color as well as intensity information both. In GCH+BBP, color histogram is computed as a global feature and block bit plane as a local feature. Thus the weighted combination of global and local features is used for image matching. Performance of methods is compared based on the following performance measures.

For i^{th} image precision is given by Eq. (6.10).

$$Precision\,(P_i) = \frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ retrieved\,images\,considered\,(T)} \tag{6.10}$$

$$Group\ precision\,(GP) = \frac{1}{N}\sum_{i=1}^{N} P_i \tag{6.11}$$

$$Average\ precision\,(AP) = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} GP_i \tag{6.12}$$

Similarly, performance measures are computed in case of recall by following Eq.

$$Recall\,(R_i) = \frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ relevant\,images\,in\,database\left(N\right)} \tag{6.13}$$

$$Group\,Recall\,(GR) = \frac{1}{N}\sum_{i=1}^{N} R_i \tag{6.14}$$

$$Average\,Recall\,(AR) = \frac{1}{\Gamma}\sum_{i=1}^{\Gamma} GR_i \tag{6.15}$$

$$Average\,Retrieval\,Rate\,(ARR) = \frac{1}{DB}\sum_{j=1}^{DB} \left. \frac{No.of\ relevant\,images\,retrieved}{Total\,no.of\ relevant\,images\,in\,database} \right|_{T \le 100} \tag{6.16}$$

Where, N is the number of relevant images in the database, Γ is the number of groups in the image database, DB is the number of total images in the database. The retrieval result is not a single image but it’s a list of images depending upon the relevancy. T gives the total number of retrieved images considered $(e.g. 10, 20, ...,100)$. Value of T is selected by user.

6.4.1 Corel 1000 Database (DB1)

The first experiment is performed on Corel 1000 database (DB1) (Appendix A) and performance measures are computed by Eq. 6.10 to 6.16. From Table 6.1, it It is also illustrated in Table 6.1 that average recall of proposed method (47.35%) is significantly better than MCM (38.72%), CC (40.81%), GCH+BBP (46.43%) and ICICM (45.56%).

is clear that the proposed method gives better average precision (78.20%) as compared to MCM, CC, GCH+BBP and ICICM (67.58%, 68.52%, 73.29% and 75.60% respectively). Fig. 6.1 shows a graph to represent average precision on total image databases as a function of number of retrieved images and it illustrates that the proposed method outperform other existing methods. It is clear from Fig. 6.1 that for top 10 and 100 retrieved images the average precision of proposed method (78.20% and 47.35%) is substantially better than MCM (67.58% and 38.72%), CC (68.52% and 40.81%), GCH+BBP (73.29% and 46.43%) and ICICM (75.60% and 45.56%) respectively.

ARR% is also computed for all methods to justify the retrieval performance. Fig. 6.2 gives *ARR%* on image database for various numbers of retrieved images. It is verified that the proposed method always outperforms other existing methods. For top 10 and 100 retrieved images, *ARR*% of proposed method (7.82% and 47.35%) is substantially better than MCM (6.76% and 38.72%), CC (6.85% and 40.81%), GCH+BBP (7.32% and 46.43%) and ICICM (7.56% and 45.56%) respectively. Fig. 6.3 illustrates four query image retrieval results (a) 230, (b) 425, (c) 615, (d) 775 of proposed method for ten retrieved images (In all cases top left image is the query image).

Table 6.1

Precision (%) comparison of proposed method with existing methods

Category Name	**Proposed method**		**MCM**		**CC**		**GCH+BBP**		**ICICM**	
	P% ***(T=10)***	**R%** ***(T=100)***	**P%** ***(T=10)***	**R%** ***(T=100)***	**P%** ***(T=10)***	**R%** ***(T=100)***	**P%** ***(T=10)***	**R%** ***(T=100)***	**P%** ***(T=10)***	**R%** ***(T=100)***
Africans	87.40	51.48	61.1	32.05	85.1	56.06	75.70	44.38	85.10	49.53
Beaches	54.80	26.54	52.5	29.95	46.1	23.01	49.00	29.16	53.80	26.68
Buildings	71.00	39.55	58.4	25	53.7	23.96	58.70	30.02	63.60	36.4
Buses	92.40	69.32	91.3	48.74	46.8	30.28	76.10	44.79	89.60	68.17
Dinosaurs	99.70	79.17	96.60	76.57	100	97.91	100.0	98.43	99.80	76.9
Elephants	73.80	34.26	50.5	23.19	68	35.38	72.20	35.59	72.10	36.73
Flowers	91.30	59.32	89.9	61.93	83.6	45.84	89.90	57.62	88.90	56.52
Horses	91.70	49.89	71.00	36.56	89.5	41.96	92.90	54.34	84.90	42.17
Mountains	45.10	25.34	44.2	24.35	37.1	18.34	46.80	26.72	45.80	24.88
Food	74.80	38.61	60.30	28.89	75.3	35.35	71.66	43.28	72.40	37.65
Average	**78.20**	**47.35**	**67.58**	**38.72**	**68.52**	**40.81**	**73.29**	**46.43**	**75.60**	**45.56**

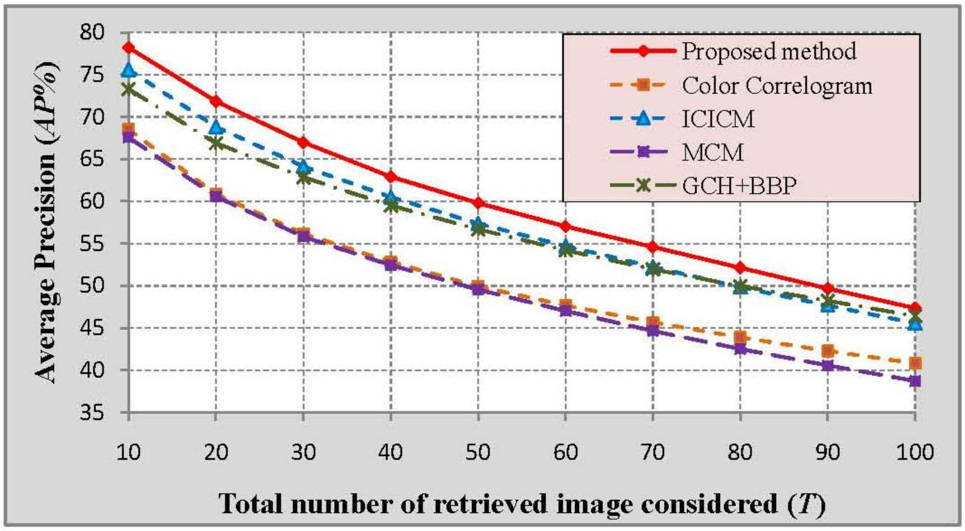


Fig. 6.1: Average precision (%) according to number of retrieved images

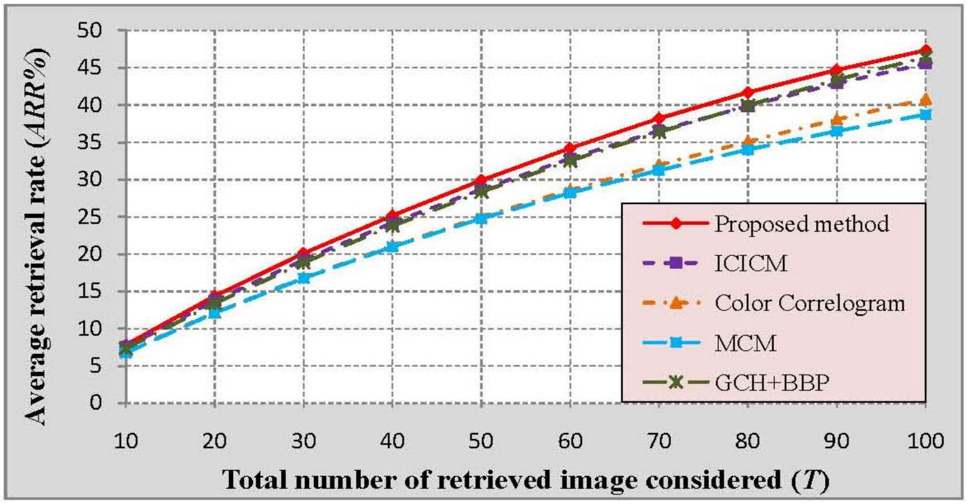


Fig. 6.2: Average retrieval rate (%) according to number of retrieved images

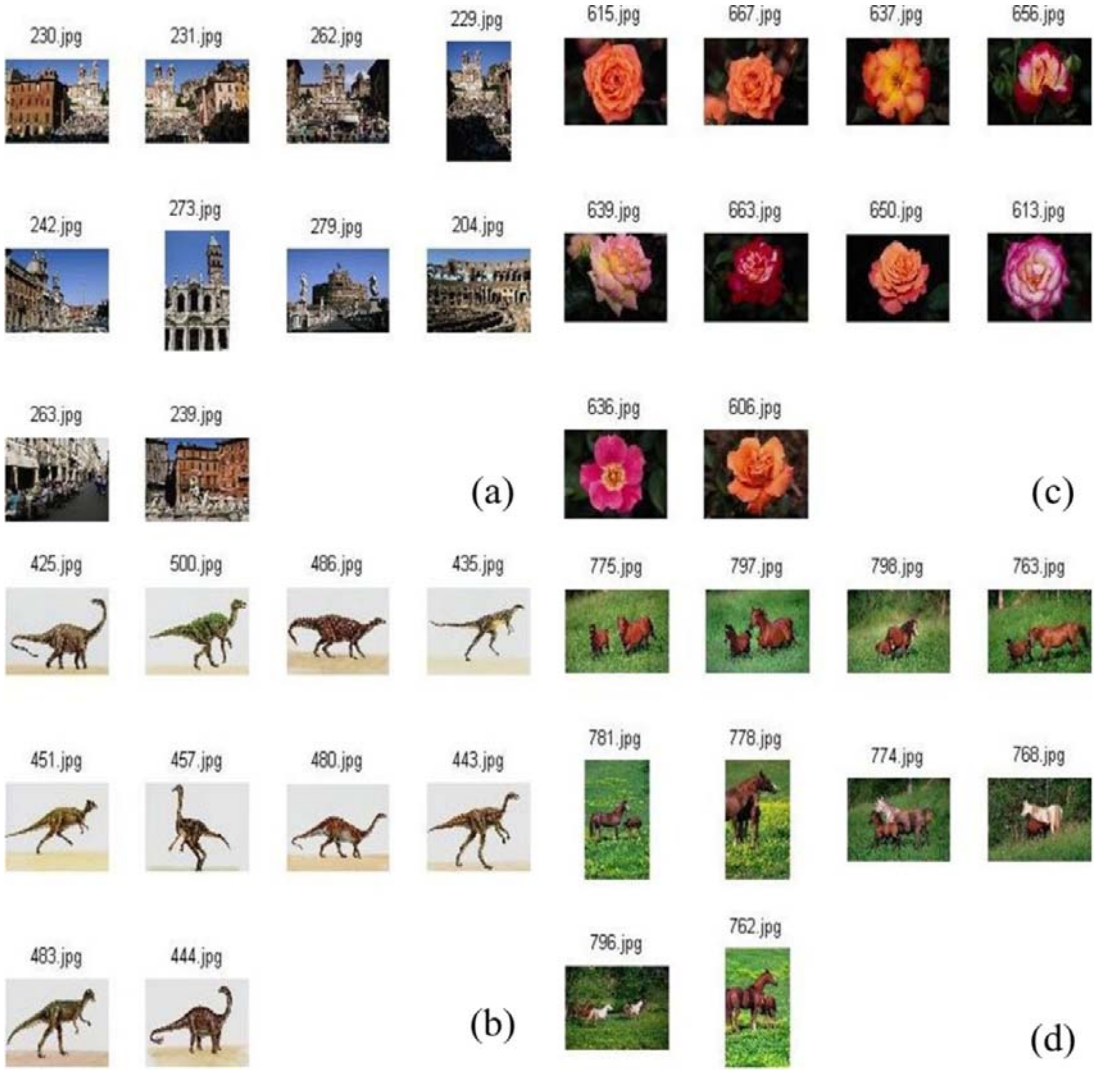


Fig. 6.3: Retrieval results of proposed method obtained for four query images

It is observed from Fig. 6.3 that all images retrieved by the proposed method are relevant images (i.e. all retrieved images belong to the query image group only). Thus, performance of proposed method is justified by image retrieval result as well.

6.4.2 Corel 2450 Database (DB2)

The next experiment is performed on Corel 2450 (DB2) (Appendix A). Average precision according to number of retrieved images is calculated on database DB2 by using Eq. 6.12. In Fig. 6.4 the trend of average precision according to number of retrieved images is shown and it is verified that the performance of proposed method, is superior in all instances. It is examined that average precision (*AP%*) of proposed method (64.23) is significantly improved in comparison to MCM (62.97), CC (59.36), GCH+BBP (55.48), ICICM (59.58) and TCM (55.75) for 10 retrieved images. Hence, better performance of proposed method is established on the image database DB2 also.

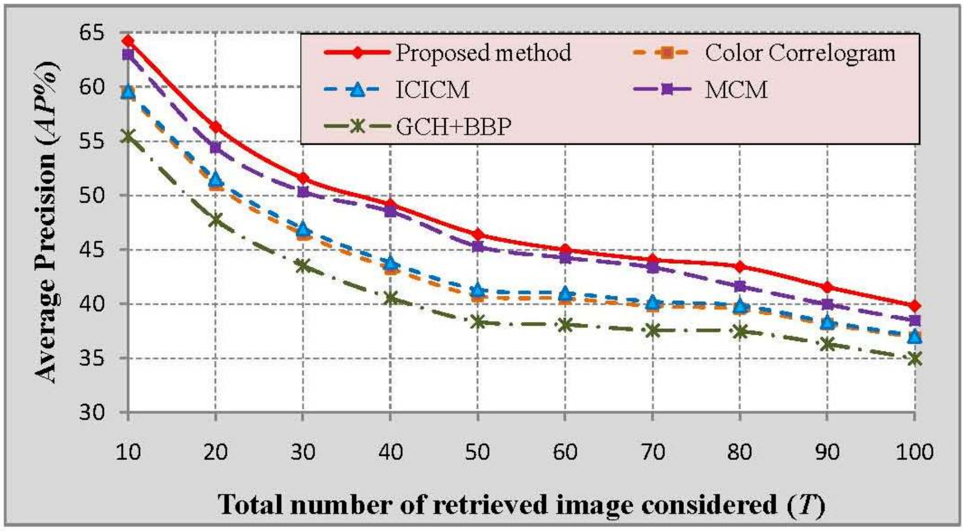


Fig. 6.4: Average precision (%) according to number of retrieved images

6.5 SUMMARY

In this chapter, a further refined approach for image retrieval is proposed by extracting the WCICIM features and its combination with ICICM. WCICIM is capable of extracting color as well as texture properties from images at the same time. It makes it suitable to capture image information for different perspectives. It is observed that WCICIM feature with the combination of ICICM feature show better performance as compared to MCM, CC, GCH+BBP and ICICM with respect to various performance evaluation measures. All experiments are performed on standard Corel 1000 and Corel 2450 image databases. Experimental results demonstrate that the proposed method has significantly improved average precision and average recall (78.20% and 47.35%) as compared to MCM (67.58% and 38.72%), CC (68.52% and 40.81%), GCH+BBP (73.29% and 46.43%) and ICICM (75.60% and 45.56%). In the same manner retrieval performance is justified by *ARR%* and image retrieval result as well. Also, in the case of Corel 2450 database the improved retrieval performance of proposed method is established.

Conclusions and Future Scope

Chapter 7

Content based image retrieval (CBIR) illustrates a method to retrieve images on the basis of automatically derived features based on visual contents present in the image like color, texture and shape in variety of domains. The need of CBIR exists so as to find desired images from a large chunk of image database, in the field of medicine, crime prevention, fashion industry, publishing and craft industry users may be interested to retrieve relevant images from a huge image database.

7.1 CONCLUSIONS AND CONTRIBUTIONS

This research work propounded enhanced qualitative approaches for CBIR systems. In the proposed research work, the goal is to enhance the efficiency and effectiveness of CBIR system by designing local, global, spatial and transform domain features as well as their integration. Experiments are conducted on five benchmark image databases (Corel 1000, Corel 2450, MIRFLICKR 25000, Brodatz, MIT VisTex) to demonstrate the effectiveness of proposed approaches. It is observed from experimental analysis that all proposed methods elucidate remarkable improvement in terms of accuracy (precision, recall, average retrieval rate etc.) with respect to related prevailing methods on respective databases.

Local features are capable of capturing fine details present in the image. In Chapter 2, a local spatial feature descriptor viz. histogram of oriented gradient (HOG) is proposed to describe the image content. HOG is computed in a local manner by dividing the gray scale image into blocks of same size. Local feature produces high dimensional feature vectors hence; to tackle with indexing of high dimensional feature vectors vocabulary tree is applied.

Since information is present in images at multiresolution thus in Chapter 3, transformed domain based multiresolution features are proposed to characterize images. Initially log Gabor filter (LGF) is proposed. It is observed that LGF represents a natural image better than Gabor wavelet transform (GWT). Another multi resolution technique, binary wavelet transform based histogram (BWTH) feature is also proposed for CBIR. BWTH is facilitated with the color as well as texture properties. Binary wavelet transform (BWT) provides image information at multiscale and multiresolution in computationally efficient manner and allows easy quantization of wavelet subband coefficients as compared to standard wavelet transform. Statistical properties of BWT decomposed subbands are extracted in terms of histogram. Further spatial relationship between subband coefficients is also integrated into BWTH and an integrated feature of binary wavelet transform (IFBWT) is proposed.

In Chapter 4, two new features viz. á trous wavelet correlogram (AWC) and á trous gradient structure descriptor (AGSD) are introduced. In AWC spatial relationship among á trous wavelet coefficients are utilized to analyze texture statistics. Further by extension AGSD is proposed. In AGSD first orientation information is computed through á trous wavelet subbands then spatial relationship among local orientation of á trous wavelet coefficients is obtained by microstructures. Microstructured image is used to map á trous wavelet image and AGSD is computed.

In Chapter 5, a spatial domain feature extraction algorithm is designed and cooccurrence of Haar–like wavelet filters (CHLWF) is introduced for CBIR. CHLWF extracts edge responses in various directions by application of two types of Haar–like filter sets. The responses are compared pixel wise to compute the most effective filter and the probability of most effective filter in four directions is used to compute CHLWF.

Integrated features also provide a way to improve effectiveness of retrieval systems on the basis of more information contained in the feature. In Chapter 6, weight cooccurrence based integrated color and intensity matrix (WCICIM) feature is proposed. It extracts both color as well as intensity information from an image. Weight matrixes are constructed by applying suitable weights function to decide the contribution of color and intensity for each pixel. WCICIM finds correlation among color–color, color–intensity, intensity–color and intensity–intensity, based on neighboring pixel variations in weight matrixes. It is observed that WCICIM feature with the combination of integrated color and intensity cooccurrence matrix (ICICM) feature illustrates better retrieval performance.

Various performance measures like average precision, average recall, average retrieval rate etc. are calculated to evaluate the retrieval performance of proposed features. In this research work total five distinct image databases are used. Following results are achieved by first to last proposed approaches on major three standard image databases.

- On Corel 1000 image database in the starting phase 55.46% average precision is achieved which is further improved to 78.20%, likewise average recall is improved from 32.03% to 47.35%. On Brodatz and MIT VisTex texture databases finally average retrieval rate is increased to 81.54% and 84.08% respectively.

7.2 SCOPE FOR FUTURE RESEARCH

- In Chapter 2, HOG is computed in local manner by dividing the gray scale image into blocks of same size. This work can be further extended to obtain better retrieval performance by segmenting the image first and then evaluate local features on each region. It may reduce the number of mismatch occurred by working on very small regions (cells) of image.
- BWT and AWT coefficients are quantized by trial and error method in BWTH and AGSD construction. In future, the performance of these two proposed methods can be improved by optimization of quantization thresholds through any optimization algorithms like genetic algorithm, particle swarm optimization, etc.
- In further research, number, size and orientations of Haar–like filters can also be optimized to enhance the efficacy of CHLWF based retrieval system.
- In AGSD and CHLWF, cooccurrence is implemented with unit distance neighborhood; large size of neighborhoods can also be included in future to obtain coarse details of intensity variations.
- WCICIM integrates only color and texture information. In future some other features can be developed by adding more image information in the feature for better image retrieval performance. Like, any shape feature can also be considered to construct the final feature vector.
- Relevance feedback provides a tool to bridge the gap between low level visual features and high level semantics by online updating of query feature according to user’s perception. Thus, relevance feedback can be integrated into the retrieval system to result in perceptually and semantically more meaningful retrieval results.

Bibliography

[1] Y. Liu, D. Zhang, G. Lu, A Survey of Content-Based Image Retrieval with High-Level Semantics, *Pattern Recognition*, 40 (1) (2007) 262–282.

[2] D. Zhang, Md. M. Islam, G. Lu, A Review on Automatic Image Annotation Techniques, *Pattern Recognition*, 45 (2012) 346–362.

[3] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain, Content-Based Image Retrieval at the End of the Early Years, *IEEE Pattern Analysis and Machine Intelligence*, 22 (12) (2000) 1349–1380.

[4] R. Datta, D. Joshi, J. Li, J. Z. Wang, Image Retrieval: Ideas, Influences and Trends of the New Age, *ACM Computing Surveys*, 40 (2) (2008) 1– 60.

[5] M. S. Lew, N. Sebe, C. Djeraba, R. Jain, Content-Based Multimedia Information Retrieval: State of the Art and Challenges, *ACM Transactions on Multimedia Computing, Communications, and Applications,* 2(1) (2006) 1–19.

[6] F. Long, H. J. Zhang, D. D. Feng, Fundamentals of Content-Based Image Retrieval, in: D.D. Feng, W.C. Siuandg, H.J. Zhan (Eds.), *Springer Multimedia Information Retrieval and Management*, (2003).

[7] Y. Rui, T. S. Huang, S. F. Chang, Image Retrieval: Current Techniques, Promising Directions and Open Issues, *Journal of Visual Communication and Image Representation*, 10 (1999) 39–62.

[8] O. A. B. Penatti, E. Valle, R. S. Torres, Comparative Study of Global Color and Texture Descriptors for Web Image Retrieval, *Journal of Visual Communication and Image Representation*, 23 (2012) 359–380.

[9] T. Deselaers, D. Keysers, H. Ney, Features for Image Retrieval: An Experimental Comparison, *Information Retrieval*, 11 (2008) 77–107.

[10] E. Mathias and A. Conci, Comparing the Influence of Color Spaces and Metrics in Content-Based Image Retrieval, in Proc. *IEEE International Symposium Computer Graphics, Image Processing, and Vision*, Rio de Janeiro, (1998) 371–378.

[11] A. Ford and A. Roberts, Color Space Conversions, [online] Available 2013: www.poynton.com/PDFs/coloureq.pdf.

[12] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice Hall, 3, (2008), ISBN: 9780131687288.

[13] H. Stokman and T. Gevers, Selection and Fusion of Color Models for Image Feature Detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29 (3) (2007) 371–381.

[14] T. Randen and J. Håkon Husøy, Filtering for Texture Classification: A Comparative Study, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21 (4) (1999) 291– 310.

[15] R. M. Haralick, K. Shanmugam, Its'Hak Dinstein, Texture Feature for Image Classification, *IEEE Transactions on Systems, Man and Cybernetics*, 3(6) (1973) 610–621.

[16] B. Julesz, Texton Gradients: The Texton Theory Revisited, *Biological Cybernetics*, 54 (1986) 245–251.

[17] M. Pietikäinen, A. Hadid, G. Zhao, T. Ahonen, Local Binary Patterns for Still Images, *Springer Computational Imaging and Vision*, Chapter 2, 40 (2011) 13–47 ISSN: 1381-6446.

[18] D. Zhang, G. Lu, Review of Shape Representation and Description Techniques, *Pattern Recognition*, 37 (2004) 1–19.

[19] S. Antani, R. Kasturi, R. Jain, A Survey on the Use of Pattern Recognition Methods for Abstraction, Indexing and Retrieval of Images and Video, *Pattern Recognition*, 35 (4) (2002) 945–965.

[20] Y. Rui, T. S. Huang, M. Ortega, And S. Mehrotra, Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval, *IEEE Transactions on Circuits and Systems for Video Technology*, 8 (5) (1998) 644–655.

[21] A. Grigorova, F. G. B. De Natale, C. Dagli, and T. S. Huang, Content-Based Image Retrieval by Feature Adaptation and Relevance Feedback, *IEEE Transactions on Multimedia*, 9 (6) (2007) 1183–1192.

[22] E. Cheng, F. Jing, And L. Zhang, A Unified Relevance Feedback Framework for Web Image Retrieval, *IEEE Transactions on Image Processing*, 18 (6) (2009) 1350–1357.

[23] V. N. Gudivada, V. V. Raghavan, Content Based Image Retrieval Systems, *IEEE Journal Computer*, 28 (9) (1995) 18–22.

[24] R. Veltkamp and M. Tanase, Content-Based Image Retrieval Systems: A Survey, *Technical Report UU-CS-2000-34*, Department of Computing Science, Utrecht University, (2002) 1–62.

[25] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, W. Equitz. Efficient and Effective Querying by Image Content. *Journal of Intelligent Information Systems*, 3 (3) (1994) 231–262.

[26] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein and J. Malik. Blobworld: A System for Region-Based Image Indexing and Retrieval, in Proc. *Springer International Conference Visual Information Systems*, The Netherlands, (1999) 509–516.

[27] A. Oerlemans, M. S. Lew, Retrievallab – A Programming Tool for Content Based Retrieval, in Proc. *1st ACM International Conference on Multimedia Retrieval,* Trento, Italy, (2011) 17–20.

[28] P. Zezula, Multi Feature Indexing Network MUFIN for Similarity Search Applications, *Springer Lecture Notes in Computer Science*, 7147 (2012) 77–87.

[29] Google Images: [online] Available 2013: http://images.google.com/

[30] T. Deselaers, D. Keysers and H. Ney, Features for Image Retrieval: A Quantitative Comparison, in Proc. *26th DAGM Symposium on Pattern Recognition*, Springer Lecture Notes in Computer Science, Tubingen Germany, 3175 (2004) 228–236.

[31] M. J. Swain, D. H. Ballard, Indexing via Color Histograms, in Proc. *IEEE 3rd International Conference Computer Vision*, Osaka, Japan, (1990) 390–393.

[32] A. Vadivel, A. K. Majumdar and S. Sural, Perceptually Smooth Histogram Generation from the HSV Color Space for Content Based Image Retrieval, in Proc. *International Conference on Advances in Pattern Recognition*, Calcutta, India, (2003) 248–251.

[33] Z.-H. Zhang, Y. Quan, W.-H. Li, W. Guo, A New Content-Based Image Retrieval, in Proc. *IEEE 5th International Conference on Machine Learning and Cybernetics*, Dalian, (2006) 4013–4018.

[34] H. N. Pour, E. Kabir, Image Retrieval using Histograms of Uni-Color and Bi-Color Blocks and Directional Changes in Intensity Gradient, *Pattern Recognition Letters*, 25 (2004) 1547–1557.

[35] M. Stricker and M. Orengo, Similarity of Color Images, in Proc. *SPIE Storage and Retrieval for Image and Video Databases*, San Jose, 2185 (1995) 381–392.

[36] M. Redi, B. Merialdo, Saliency-Aware Color Moments Features for Image Categorization and Retrieval, in Proc. *IEEE International Workshop on Content-Based Multimedia Indexing*, Madrid, (2011) 199–204.

[37] G. Pass, R. Zabih and J. Miller, Comparing Images using Color Coherence Vectors, in Proc. *4th ACM Multimedia Conference,* Boston, Massachusetts, US, (1997) 65–73.

[38] J. Huang, S. R. Kumar, M. Mitra, W. J. Zhu, R. Zabih, Image Indexing using Color Correlograms, in Proc. *IEEE Conference Computer Vision and Pattern Recognition*, San Juan, (1997) 762–768.

[39] P. Kinnaree, S. Pattanasethanon, S. Thanaputtiwirota, S. Boontho, RGB Color Correlation Index for Image Retrieval, *Procedia Engineering*, 8 (2011) 36–41.

[40] T. Ojala, M. Rautiainen, E. Matinmikko and M. Aittola, Semantic Image Retrieval with HSV Correlograms, in Proc. *The Scandinavian Conference on Image Analysis*, Norway, (2001) 621–627.

[41] S. O. Shim, T. S. Choi, Image Indexing by Modified Color Co–occurrence Matrix, in Proc. *IEEE International Conference on Image Processing*, Spain, 3 (2003) 493–496.

[42] J. Luo, and D. Crandall, Color Object Detection using Spatial-Color Joint Probability Functions, *IEEE Transactions on Image Processing*, 15 (6) (2006) 1443–1453.

[43] D. Crandall, J. Luo, Robust Color Object Detection using Spatial-Color Joint Probability Functions, in Proc. *IEEE Conference on Computer Vision and Pattern Recognition,* Washington, DC, 1 (2004) 379–385.

[44] R. Phan, D. Androutsos, Content-Based Retrieval of Logo and Trademarks in Unconstrained Color Image Databases using Edge Gradient Co-Occurrence Histogram, *J. Computer Vision and Image Understanding*, 114 (2010) 66–84.

[45] C. Lin, D. Huang, Y. Chan, K. Chen, Y. Chang, Fast Color-Spatial Feature Based Image Retrieval Methods, *Expert Systems with Applications*, 38 (2011) 11412–11420.

[46] S. Kiranyaz, M. Birinci, M. Gabbouj, Perceptual Color Descriptor Based on Spatial Distribution: A Top-Down Approach, *Image and Vision Computing*, 28 (2010) 1309–1326.

[47] T. Lu, C. Chang, Color Image Retrieval Technique Based on Color Features and Image Bitmap, *Information Processing and Management*, 43 (2007) 461–472.

[48] C. Urdiales, M. Dominguez, C. De Trazegnies, F. Sandoval, A New Pyramid-Based Color Image Representation for Visual Localization, *Image and Vision Computing,* 28 (2010) 78–91.

[49] M. H. Kolekar and S. Sengupta, A Hierarchical Framework for Generic Sports Video Classification, *Lecture Notes in Computer Science*, 3852 (2006) 633–642.

[50] M. Tuceryan and A. K. Jain, Texture Analysis, *Handbook of Pattern Recognition and Computer Vision*, World Scientific, 2 (1998).

[51] H. Tamura, S. Mori, T. Yamawaki, Textural Features Corresponding to Visual Perception, *IEEE Transactions on Systems, Man and Cybernetics*, 8 (6) (1978) 460–473.

[52] F. Malik, B. Baharudin, Efficient Image Retrieval Based on Texture Features, in Proc. *National Postgraduate Conference–Energy and Sustainability: Exploring the Innovative Minds,* Malaysia, (2011) 1– 6.

[53] F. Malik, B. Baharudin, Median and Laplacian Filters Based Feature Analysis for Content Based Image Retrieval using Color Histogram Refinement Method, *Journal of Applied Sciences*, 12 (5) (2012) 416–427.

[54] P. Shah, S. N. Merchant, U. B. Desai, An Efficient Spatial Domain Fusion Scheme for Multifocus Images using Statistical Properties of Neighborhood, in Proc. *IEEE International Conference Multimedia and Expo*, Barcelona, Spain, (2011) 1–6.

[55] N. Jhanwar, S. Chaudhuri, G. Seetharaman, B. Zavidovique, Content Based Image Retrieval using Motif Cooccurrence Matrix, *Image and Vision Computing*, 22 (2004) 1211–1220.

[56] G. Liu, J. Yang, Image Retrieval Based on the Texton Co-occurrence Matrix, *Pattern Recognition*, 41 (2008) 3521–3527.

[57] G. Liu, L. Zhang, Y. Hou, Z. Li, J. Yang, Image Retrieval Based on Multi-Texton Histogram, *Pattern Recognition*, 43 (2010) 2380–2389.

[58] G. Liu, Z. Li, L. Zhang, Y. Xu, Image Retrieval Based on Micro-Structure Descriptor, *Pattern Recognition*, 44 (2011) 2123–2133.

[59] T. Xu, I. Gondra, A Simple and Effective Texture Characterization for Image Segmentation, *Signal Image and Video Processing*, 6 (2) (2012) 231–245.

[60] T. Mäenpää, M. Pietikäinen, Texture Analysis with Local Binary Patterns, in: C. *Chen, P. Wang (Eds.) Handbook of Pattern Recognition and Computer Vision*, World Scientific, Singapore, Chapter 1, (2005) 197–216.

[61] T. Ojala, M. Pietikäinen, T. Mäenpää, Multi-Resolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24 (7) (2002) 971–987.

[62] M. Li, and R. C. Staunton, Optimum Gabor Filter Design and Local Binary Patterns for Texture Segmentation, *Pattern Recognition Letters*, 29 (5) (2008) 664–672.

[63] X. Qian, X. Hua, P. Chen, L. Ke, PLBP: An Effective Local Binary Patterns Texture Descriptor with Pyramid Representation, *Pattern Recognition*, 44 (10) (2011) 2502–2515.

[64] Z. Guo, L. Zhang, D. Zhang, Rotation Invariant Texture Classification using LBP Variance (LBPV) with Global Matching, *Pattern Recognition*, 43 (2010) 706–719.

[65] S. Murala, R. P. Maheshwari, R. Balasubramanian, Directional Binary Wavelet Patterns for Biomedical Image Indexing and Retrieval, *Medical System*, 36 (5) (2012) 2865–2879.

[66] N. Armanfard, M. Komeili, E. Kabir, TED: A Texture-Edge Descriptor for Pedestrian Detection in Video Sequences, *Pattern Recognition*, 45 (3) (2012) 983–992.

[67] S. Prakash, P. Gupta, A Rotation and Scale Invariant Technique for Ear Detection in 3D, *Pattern Recognition Letters*, 33 (14) (2012) 1924–1931.

[68] M. Masotti, R. Campanini, Texture Classification using Invariant Ranklet Features, *Pattern Recognition Letters*, 29 (2008) 1980–1986.

[69] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna and T. Poggio, Pedestrian Detection using Wavelet Templates, in Proc. *IEEE Conference Computer Vision and Pattern Recognition,* San Juan, (1997) 193–99.

[70] P. Viola, M. J. Jones, Rapid Object Detection using a Boosted Cascade of Simple Features, in Proc. *IEEE Conference Computer Vision and Pattern Recognition*, Los Alamitos, (2001) 511–518.

[71] P. Viola, M. J. Jones, Robust Real-Time Face Detection, *International Journal of Computer Vision*, 57 (2) (2004) 137–154.

[72] R. Lienhart, J. Maydt, An Extended Set of Haar-Like Features for Rapid Object Detection, in *Proc. IEEE International Conference Image Processing*, NY, USA, 1 (2002) 900–903.

[73] M. Pham, T. Cham, Fast Training and Selection of Haar Features using Statistics in Boosting-Based Face Detection, in Proc. *IEEE International Conference Computer Vision,* Brazil, (2007) 1–7.

[74] S. Z. Li, L. Zhu, Z. Zhang, A. Blake, H. Zhang, H. Shum, Statistical Learning of Multi- View Face Detection, in Proc. *European Conference Computer Vision*, *Lecture Notes in Computer Science*, London, UK, 2353 (2002) 67–81.

[75] S. K. Pavani, D. Delgado, A. F. Frangi, Haar-like Features with Optimally Weighted Rectangles for Rapid Object Detection, *Pattern Recognition*, 43 (2010) 160–172.

[76] S. Mallat, A Theory for Multiresolution Signal Decomposition: The Wavelet Representation, *IEEE Transaction Pattern Analysis and Machine Intelligence*, 11 (1989) 674–693.

[77] J. R. Smith, S. Chang, Automated Binary Texture Feature Sets for Image Retrieval, in Proc. *IEEE International Conference Acoustics, Speech and Signal Processing*, Atlanta, 4 (1996) 2239–2242.

[78] M. M. Mushrif, S. Sengupta and A. K. Ray, Texture Classification using a Novel, Soft-Set Theory Based Classification Algorithm, *Lecture Notes in Computer Science*, 3851 (2006) 246–254.

[79] G. R. Joubert and O. Kao, Efficient Dynamic Image Retrieval using the Á trous Wavelet Transformation, *Lecture Notes in Computer Science,* 2195 (2001) 343–350.

[80] M. G. Audicana, X. Otazu, O. Fors and A. Seco, Comparison Between Mallat's and the 'Á trous' Discrete Wavelet Transform Based Algorithms for the Fusion of Multispectral and Panchromatic Images, *Taylor Francis International Journal Remote Sensing*, 26 (2005) 1–19.

[81] M. Kokare, P. K. Biswas, B. N. Chatterji, Texture Image Retrieval using Rotated Wavelet Filters, *Pattern Recognition Letters*, 28 (2007) 1240–1249.

[82] H. A. Moghaddam, T. T. Khajoie and A. H. Rouhi, A New Algorithm for Image Indexing and Retrieval using Wavelet Correlogram, in Proc. *IEEE International Conference Image Processing*, Barcelona, 2 (2003) 497–500.

[83] M. Saadatmand–T, H. A. Moghaddam, A Novel Evolutionary Approach for Optimizing Content Based Image Retrieval, *IEEE Transactions Systems, Man, and Cybernetics*, 37 (1) (2007) 139–153.

[84] P. S. Hiremath, S. Shivashankar, Wavelet Based Co-Occurrence Histogram Features for Texture Classification with an Application to Script Identification in a Document Image, *Pattern Recognition Letters*, 29 (2008) 1182–1189.

[85] Y. Li, X. Chen, X. Fu and S. Belkasim, Multi-Level Discrete Cosine Transform for Content-Based Image Retrieval by Support Vector Machines, in Proc. *IEEE International Conference Image Processing*, Texas, U.S.A., 6 (2007) 21–24.

[86] M. D. Swanson, A. H. Tewfik, A Binary Wavelet Decomposition of Binary Images, *IEEE Transactions Image Processing*, 5 (12) (1996) 1637–1650.

[87] H. Pan, L. Jin, X. Yuan, S. Xia, L. Xia, Context Based Embedded Image Compression using Binary Wavelet Transform, *Image and Vision Computing*, 28 (2010) 991–1002.

[88] H. Pan, W. C. Siu, N. F. Law, Lossless Image Compression using Binary Wavelet Transform, *IET Image Processing*, 1 (4) (2007) 353– 362.

[89] N. F. Law, W. C. Siu, A Filter Design Strategy for Binary Field Wavelet Transform using the Perpendicular Constraint, *Signal Processing*, 87 (2007) 2850–2858.

[90] B. S. Manjunath, W. Y. Ma, Texture Features for Browsing and Retrieval of Image Data, *IEEE Transactions Pattern Analysis and Machine Intelligence*, 18 (8) (1996) 837–842.

[91] H. A. Moghaddam, M. S. Tarzjan, Gabor Wavelet Correlogram Algorithm for Image Indexing and Retrieval, in Proc. *18th IEEE International Conference Pattern Recognition*, Hong Kong, (2006) 925–928.

[92] S. Murala, R. P. Maheshwari, R. Balasubramanian, Optimization of Gabor Wavelet Correlogram Quantization Thresholds Using Genetic Algorithm, *TECHNIA – International Journal of Computing Science and Communication Technologies*, 4 (1) (2011) 662–667.

[93] S. Arivazhagan, L. Ganesan, S. Padam Priyal, Texture Classification using Gabor Wavelets Based Rotation Invariant Features, *Pattern Recognition Letters*, 27 (2006) 1976–1982.

[94] J. Han, K. Ma, Rotation-Invariant and Scale-Invariant Gabor Features for Texture Image Retrieval, *Image and Vision Computing*, 25 (2007) 1474–1481.

[95] D. J. Field, Relations Between the Statistics of Natural Images and the Response Properties of Cortical Cells, *Journal Optical Society of America A*, 4 (12) (1987) 2379–2394.

[96] X. Gao, F. Sattar, R. Venkateswarlu, Multiscale Corner Detection of Gray Level Images Based on Log-Gabor Wavelet Transform, *IEEE Transactions Circuits and Systems for Video Technology*, 17 (7) (2007) 868–875.

[97] W. R. Schwartz, R. D. Da Silva, Larry S. Davis, H. Pedrini, A Novel Feature Descriptor Based on the Shearlet Transform, in Proc. *IEEE International Conference on Image Processing,* Brussels, Belgium, (2011) 1053–1056.

[98] E. J. Candes and D. L. Donoho, Ridgelets: A Key to Higher-Dimensional Intermittency, *Philosophical Transactions of the Royal Society A*, (1999) 2495–2509.

[99] K. Huang and S. Aviyente, Rotation Invariant Texture Classification with Ridgelet Transform and Fourier Transform, in Proc. *IEEE International Conference on Image Processing*, Atlanta, USA (2006) 2141–2144.

[100] Y. Qiao, C. Song, C. Zhao, M-Band Ridgelet Transform Based Texture Classification, *Pattern Recognition Letters*, 31 (2010) 244–249.

[101] S. B. Nikam, S. Agarwal, Ridgelet-Based Fake Fingerprint Detection, *Neurocomputing*, 72 (2009) 2491–2506.

[102] A. B. Gonde, R. P. Maheshwari, R. Balasubramanian, Multiscale Ridgelet Transform for Content Based Image Retrieval, *IEEE 2nd International Advance Computing Conference*, Patiala, (2010) 139–144.

[103] J. Starck, E. J. Candès and D. L. Donoho, The Curvelet Transform for Image Denoising, *IEEE Transactions on Image Processing*, 11 (6) (2002) 670–684.

[104] I. J. Sumana, Md. M. Islam, D. Zhang and G. Lu, Content Based Image Retrieval using Curvelet Transform, in Proc. *10th IEEE Workshop Multimedia Signal Processing*, Cairns, Qld, (2008) 11–16.

[105] A. Mosleh, F. Zargari, R. Azizi, Texture Image Retrieval using Contourlet Transform, in Proc. *IEEE International Symposium Signals, Circuits and Systems*, Iasi, (2009) 1–4.

[106] P. Shah, S N. Merchant, U. B. Desai, Multifocus and Multispectral Image Fusion Based on Pixel Significance using Multiresolution Decomposition, *Signal, Image and Video Processing*, 7 (1) (2013) 95–109.

[107] N. G. Kingsbury, Image Processing With Complex Wavelet, *Philosophical Transactions of the Royal Society A*, 137 (1999) 2543–2560.

[108] A. B. Gonde, R. P. Maheshwari, R. Balasubramanian, Complex Wavelet Transform with Vocabulary Tree for Content Based Image Retrieval, in Proc. *7th IEEE Indian Conference on Computer Vision, Graphics and Image Processing,* New York (2010) 359–366.

[109] H. Wang, X. He, W. Zai, Texture Image Retrieval using Dual-Tree Complex Wavelet Transform, in Proc. *IEEE International Conference Wavelet Analysis and Pattern Recognition*, Beijing, 1 (2007) 230–234.

[110] M. Kokare, P. K. Biswas and B.N. Chatterji, Texture Image Retrieval using New Rotated Complex Wavelet Filters, *IEEE Transactions Systems, Man and Cybernetics—Part B: Cybernetics*, 35 (6) (2005) 1168–1178.

[111] M. Kokare, P. K. Biswas and B. N. Chatterji, Rotation-Invariant Texture Image Retrieval using Rotated Complex Wavelet Filters, *IEEE Transactions Systems, Man, and Cybernetics—Part B: Cybernetics*, 36 (6) (2006) 1273–1282.

[112] T. Torra and Y. Narukawa, Information Fusion and Aggregation Operators, *Springer Modeling Decision,* 14 (2007), 284, ISBN: 978-3-540-68789-4

[113] H. Tahani and J.M. Keller, Information Fusion in Computer Vision using Fuzzy Integral, *IEEE Transaction on System, Man and Cybernetics*, 20 (3) (1990) 733–741.

[114] T. Gevers and A. W. M. Smeulders, Pictoseek: Combining Color and Shape Invariant Features for Image Retrieval, *IEEE Transaction on Image Processing*, 9 (1) (2000) 102–119.

[115] B. S. Manjunath, J. –R. Ohm, V. V. Vasudevan and Akio Yamada, Color and Texture Descriptors, *IEEE Transactions on Circuits and Systems for Video Technology*, 11 (6) (2001) 703–715.

[116] C. Tanase and B. Merialdo, Efficient Spatio-Temporal Edge Descriptor, *Lecture Notes in Computer Science*, 7131 (2012) 210–221.

[117] H. Shao, J. W. Zhang, W. C. Cui, H. Zhao, Automatic Feature Weight Assignment Based on Genetic Algorithm for Image Retrieval, in Proc. *IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, Changsha, China, 2 (2003) 731–735.

[118] P. Liu, K. Jia, Z. Wang, Z. Lv, A New and Effective Image Retrieval Method Based on Combined Features, in Proc. *IEEE 4th International Conference on Image and Graphics,* Sichuan, (2007) 786–790.

[119] A. Vadivel, S. Sural, A. Majumdar, An Integrated Color and Intensity Co–occurrence Matrix, *Pattern Recognition Letters*, 28 (2007) 974–983.

[120] J. Ha, G. Kim, H. Choi, The Content-Based Image Retrieval Method using Multiple Features, in Proc. *IEEE Fourth International Conference on Networked Computing and Advanced Information Management*, Gyeongju, (2008) 652–657.

[121] C. Lin, R. Chen, Y. Chan, A Smart Content-Based Image Retrieval System Based on Color and Texture Feature, *Image and Vision Computing*, 27 (6) (2009) 658–665.

[122] A. Abdullah, R. C. Veltkamp, M. A. Wiering, Fixed Partitioning and Salient Points with MPEG-7 Cluster Correlograms for Image Categorization, *Pattern Recognition*, 43 (2010) 650–662.

[123] X. Wang, Y. Yu, H. Yang, An Effective Image Retrieval Scheme using Color, Texture and Shape Features, *Computer Standards & Interfaces*, 33 (2011) 59–68.

[124] M. Hanmandlu and A. Das, Content-Based Image Retrieval by Information Theoretic Measure, *Defence Science Journal*, 61 (5) (2011) 415–430.

[125] Y. D. Chun, N. C. Kim and I. H. Jang, Content-Based Image Retrieval using Multiresolution Color and Texture Features, *IEEE Transactions on Multimedia*, 10 (6) (2008) 1073–1084.

[126] S. Selvarajah and S. R. Kodithuwakku, Combined Feature Descriptor for Content Based Image Retrieval, in Proc. *IEEE 6th International Conference on Industrial and Information Systems,* Sri Lanka, (2011) 164–168.

[127] S. Selvarajah and S. R. Kodituwakku, Analysis and Comparison of Texture Features for Content Based Image Retrieval, *International Journal of Latest Trends in Computing*, 2 (1) (2011) 108–113.

[128] H. A. Jalab, Image Retrieval System Based on Color Layout Descriptor and Gabor Filters, in Proc. *IEEE International Conference Open System*, Malaysia, (2011) 32–36.

[129] C. Palm, Color Texture Classification By Integrative Co-occurrence Matrices, *Pattern Recognition*, 37 (2004) 965–976.

[130] F. Bianconi, A. Fernández, E. González, D. Caride, A. Calviño, Rotation-Invariant Colour Texture Classification Through Multilayer CCR, *Pattern Recognition Letters*, 30 (2009) 765–773.

[131] D. S. Zhang, G. Lu, A Comparative Study of Fourier Descriptors for Shape Representation and Retrieval, in Proc. *The 5th Asian Conference on Computer Vision*, Melbourne, Australia, (2002) 646–651.

[132] M. Hu, Visual Pattern Recognition by Moment Invariants, *IRE Transactions on Information Theory*, 8 (2) (1962) 179–187.

[133] A. Çapar, B. Kurt, M. Gökmen, Gradient-Based Shape Descriptors, *Machine Vision And Applications,* 20 (2009) 365–378.

[134] S. Abbasi, F. Mokhtarian, J. Kittler, Enhancing CSS-Based Shape Retrieval for Objects with Shallow Concavities, *Image and Vision Computing*, 18 (2000) 199–211.

[135] A. El-Ghazal, O. Basir, S. Belkasim, Farthest Point Distance: A New Shape Signature for Fourier Descriptors, *Signal Processing: Image Communication*, 24 (2009) 572–586.

[136] R. Mukundan, K. R. Ramakrishnan, Fast Computation of Legendre and Zernike Moments, *Pattern Recognition*, 28 (9) (1995) 1433–1442.

[137] J. Revaud, G. Lavoue and A. Baskurt, Improving Zernike Moments Comparison for Optimal Similarity and Rotation Angle Retrieval, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31 (4) (2009) 627–636.

[138] H. Zhang, H. Z. Shu, P. Haigron, B. S. Li, L. M. Luo, Construction of a Complete Set of Orthogonal Fourier–Mellin Moment Invariants for Pattern Recognition Applications, *Image and Vision Computing*, 28 (2010) 38–44.

[139] A. Rodtook, S. S. Makhanov, Selection of Multiresolution Rotationally Invariant Moments for Image Recognition, *Mathematics and Computers in Simulation*, 79 (2009) 2458–2475.

[140] X. Chen and I. S. Ahmad, Shape-Based Image Retrieval Using *K*-Means Clustering and Neural Networks, *Lecture Notes in Computer Science,* 4872 (2007) 893–904.

[141] S. P. Priyal and P. K. Bora, A Study on Static Hand Gesture Recognition using Moments, in Proc. *IEEE International Conference Signal Processing and Communications*, Bangalore, (2010) 1–5.

[142] M. K. Bhuyan, D. Ghosh and P. K. Bora, Feature Extraction from 2D Gesture Trajectory in Dynamic Hand Gesture Recognition, in Proc. *IEEE Conference Cybernetics and Intelligent Systems*, Bangkok, (2006) 1–6.

[143] J. Li, N. M. Allinson, A Comprehensive Review of Current Local Features for Computer Vision, *Neurocomputing*, 71 (2008) 1771–1787.

[144] D. G. Lowe, Distinctive Image Features from Scale–Invariant Keypoints, *Int. J. Computer Vision*, 2 (60) (2004) 91−110.

[145] S. Murala, R. P. Maheshwari, R. Balasubramanian, A relevance feedback–based learner for image retrieval using SIFT descriptors, *International Journal of Computational Vision and Robotics*, 2 (2) (2011) 99−114.

[146] N. Dalal and B. Triggs, Histograms of Oriented Gradients for Human Detection, in Proc. *IEEE Conference Computer Vision and Pattern Recognition*, San Diego, CA, USA, 1 (2005) 886−893.

[147] N. Dalal, B. Triggs and C. Schmid, Human Detection using Oriented Histograms of Flow and Appearance, *European Conference Computer Vision*, Graz, Austria, 2 (2006) 428−441.

[148] Q. Zhu, S. Avidan, M. C. Yeh, and K. T. Cheng, Fast Human Detection using a Cascade of Histograms of Oriented Gradients, in Proc. *IEEE Conference Computer Vision and Pattern Recognition*, NY, 2 (2006) 1491−1498.

[149] A. Albiol, D. Monzo, A. Martin, J. Sastre and A. Albiol, Face Recognition using HOG–EBGM, *Pattern Recognition Letters*, 29 (10) (2008) 1537−1543.

[150] N. He, J. Cao and L. Song Scale Space Histogram of Oriented Gradients for Human Detection, in Proc. *IEEE Symposium on Information Science and Engineering*, Shanghai, (2008) 167−170.

[151] L. Nanni and A. Lumini, Descriptors for Image−Based Fingerprint Matchers, *Expert Systems with Applications*, 36 (10) (2009) 12414−12422.

[152] U. Jayaraman, S. Prakash, P. Gupta, An Efficient Color And Texture Based Iris Image Retrieval Technique, *Expert Systems with Applications*, 39 (2012) 4915–4926.

[153] C. R. Huang, C. S. Chen, P. C. Chung, Contrast Context Histogram−An Efficient Discriminating Local Descriptor for Object Recognition and Image Matching, *Pattern Recognition*, 41 (2008) 3071−3077.

[154] Z. Z. Yong, L. S. Gang, A Novel Region-Based Image Retrieval Algorithm using Hybrid Feature, in Proc. *IEEE World Congress on Computer Science and Information Engineering*, Los Angeles, CA, (2009) 416−420.

[155] M. Hanmandlu, R. Vijay, N. Mittal, A Study of Some New Features for the Palmprint Based Authentication, in Proc. *World Congress on Engineering*, London, U.K, 2 (2011) 1623−1628.

[156] A. Mohabey, A. K. Ray, Fusion of Rough Set Theoretic Approximations and FCM for Color Image Segmentation, in Proc. *IEEE Int. Conf. System, Man and Cybernetics*, Nashville, 2 (2000) 1529−1534.

[157] M. M. Mushrif and A. K. Ray, Color Image Segmentation: Rough-Set Theoretic Approach, *Pattern Recognition Letters*, 29 (4) (2008) 483−493.

[158] J. Stottinger, A. Hanbury, N. Sebe and T. Gevers, Sparse Color Interest Points for Image Retrieval and Object Categorization, *IEEE Transactions on Image Processing*, 21 (5) (2012) 2681−2692.

[159] N. M. Khan, I. S. Ahmad, An Efficient Signature Representation for Retrieval of Spatially Similar Images, *Signal, Image and Video Processing*, 6 (1) (2012) 55−70.

[160] S. B. Nikam, P. Goel, R. Tapadar, S. Agarwal, Combining Gabor Local Texture Pattern and Wavelet Global Features for Fingerprint Matching, in Proc. *IEEE International Conference on Computational Intelligence and Multimedia Applications*, Sivakasi, Tamil Nadu, 2 (2007) 412−417.

[161] X. Pan, Q. Ruan, Palmprint Recognition using Gabor-Based Local Invariant Features, *Neurocomputing*, 72 (2009) 2040–2045.

[162] E. Choi, C. Lee, Feature Extraction Based on the Bhattacharyya Distance, *Pattern Recognition*, 36 (2003) 1703–1709.

[163] T. Chaira, A. K. Ray, Fuzzy Measures for Color Image Retrieval, *Fuzzy Sets and Systems*, 150 (2005) 545–560.

[164] I. Gondra, D. R. Heisterkamp, Content-Based Image Retrieval with the Normalized Information Distance, *Computer Vision and Image Understanding*, 111 (2008) 219–228.

[165] B. Zou and M. P. Umugwaneza, Shape-Based Trademark Retrieval using Cosine Distance Method, in Proc. *8th International Conference Intelligent Systems Design and Applications*, Kaohsiung, 2 (2008) 498–504.

[166] D. Nister, and H. Stewenius Scalable Recognition with a Vocabulary Tree, in Proc. *IEEE Conference Computer Vision and Pattern Recognition*, NY, 2 (2006) 2161−2168.

[167] Corel 1000 And Corel 10000 Image Database. [Online]. Available 2013: http://wang.ist.psu.edu/docs/related/

[168] S. Murala, A. B. Gonde and R. P. Maheshwari, Color and Texture Features for Image Indexing and Retrieval, in Proc. *IEEE International Advance Computing Conference,* Patiala, (2009) 1411−1416.

[169] M. J. Huiskes, M. S. Lew, The MIR Flickr Retrieval Evaluation, in Proc. *ACM International Conference on Multimedia Information Retrieval,* Vancouver, (2008) 39–43.

[170] M. Subrahmanyam, R. P. Maheshwari, R. Balasubramanian, A Correlogram Algorithm for Image Indexing and Retrieval using Wavelet and Rotated Wavelet Filters, *International Journal of Signal Imaging System Engineering*, 4 (2011) 27–34.

[171] A. B. Gonde, R. P. Maheshwari, R. Balasubramanian, Texton Co-occurrence Matrix: A New Feature for Image Retrieval, *IEEE India Conference*, Chennai, (2010) 1–5.

[172] P. Brodatz, *Textures: A Photographic Album For Artists And Designers*. New York: Dover, (1996).

[173] University of Southern California, Los Angeles, Signal and Image Processing Institute, [Online]. Available 2013: http://sipi.usc.edu/database/

[174] MIT Vision and Modeling Group, Cambridge, Vision Texture, [Online]. Available 2013: http://vismod.media.mit.edu/pub/VisTex

[175] A. B. Gonde, R. P. Maheshwari, R. Balasubramanian, Content Based Image Retrieval using Color Feature and Color Bit Planes, *International Journal of Signal and Imaging Systems Engineering*, 3 (2) (2010) 105–115.

[176] N. Rose, Facial Expression Classification using Gabor and Log-Gabor Filters, in Proc. *IEEE 7th International Conference Automatic Face and Gesture Recognition*, Southampton, (2006) 346–350.

[177] X. Zhitao, G. Chengming, Y. Ming, L. Qiang, Research on Log Gabor Wavelet and its Application in Image Edge Detection, in Proc. *IEEE 6th International Conference Signal Processing*, Beijing, China, 1 (2002) 592–595.

[178] P. D. Kovesi. What Are Log-Gabor Filters and Why Are They Good?, School of Computer Science & Software Engineering, The University of Western Australia. [Online]. Available 2013: http://www.csse.uwa.edu.au/~pk/Research/MatlabFns/PhaseCongruency/Docs/convexpl.html

[179] S. Fischer, R. Redondo and G. Cristóbal, How to construct Log-Gabor filters. *Open Access Digital CSIC Document*, (2009).

[180] M. Saadatmand-T. and H. A. Moghaddam, Enhanced Wavelet Correlogram Methods for Image Indexing and Retrieval, in Proc. *IEEE International Conference Image Processing*, Genoa, Italy, 1 (2005) 541–544.

[181] V. Jain, A. Mukherjee, The Indian Face Database, [Online]. Available 2013: http://vis-www.cs.umass.edu/~vidit/IndianFaceDatabase/

[182] M. N. Do, M. Vetterli, Wavelet-based texture retrieval using generalized Gaussian density and Kullback–Leibler distance, *IEEE Transactions on Image Processing*, 11 (2) (2002) 146–158.

Publications from the Research Work

Papers in International Journals:

[1] Megha Agarwal, R. P. Maheshwari, HOG Feature and Vocabulary Tree for Content Based Image Retrieval, *International Journal of Signal and Imaging Systems Engineering*, 3 (4) (2010) 246–254.

[2] Megha Agarwal, R. P. Maheshwari, Content Based Image Retrieval Based on Log Gabor Wavelet Transform, *Journal of Advanced Materials Research*, 403 (2012) 871–878.

[3] Megha Agarwal, R. P. Maheshwari, Binary Wavelet Transform Based Histogram Feature for Content Based Image Retrieval, *International Journal of Electronic Signals and Systems*, 1 (1) (2011) 68–75.

[4] Megha Agarwal, R. P. Maheshwari, Á trous Gradient Structure Descriptor for Content Based Image Retrieval, *International Journal of Multimedia Information Retrieval*, 1 (2) (2012) 129–138.

[5] Megha Agarwal, R. P. Maheshwari, Co-occurrence of Haar-Like Wavelet Filters for Content Based Image Retrieval, *International Journal of Signal and Imaging Systems Engineering.*

[6] Megha Agarwal, R. P. Maheshwari, Weight co-occurrence Based Integrated Color and Intensity Matrix for CBIR, *TECHNIA: International Journal of Computing Science and Communication Technologies*, 4 (1) (2011) 651–656.

Papers in International Conference:

[1] Megha Agarwal, R. P. Maheshwari, Image Retrieval using Integrated Features of Binary Wavelet Transform, *AIP Conference Proceeding*, Jaipur, India, 1414 (2011) 188–192.

Appendix A

To compare the performance of various feature descriptors it is required to demonstrate their performance on same standard image databases. Due to variety in contents following databases are being used by researchers in various scientific articles of CBIR. In this research work experiments are performed on following databases.

A.1 COREL 1000

Corel database [167] is widely used in the field of CBIR due to large amount of images and variety in contents ranging from animals and outdoor sports to natural images. In the experiments 1,000 images are collected to form Corel 1000 database. These images are pre-classified into 10 categories by experts and each category contains 100 images of similar type. Ten diverse categories used in this database are Africans, beaches, buildings, buses, dinosaurs, elephants, flowers, horses, mountains and food. Each category of this database has images of different sizes (either 256×384 or 384×256) in *jpg* format. Fig. A.1 comprises few example images of Corel 1000 image database (Each row has images from same category).

A.2 COREL 2450

Corel 2450 [167] image database contains a total of 2,450 images. It consists of images with two different sizes (either 256×384 or 384×256*)*. All images are in *jpg* format. These images are pre-categorized into 19 groups. Each category contains 50–600 images. Corel 2450 contains all categories of Corel 1000 image database with additional categories of airplane, panther, bird, deer, fox, car, ship, tiger and train. Thus, this database covers more diversified categories to explore effectiveness of retrieval system. Fig. A.2 comprises few example images from Corel 2450 image database (additional to Corel 1000).

A.3 MIRFLICKR 25000

MIRFLICKR 25000 database [169] consists of 25,000 images with tags. All images are in *jpg* format and are publically available on internet dedicated to research community [169]. This collection is prepared from the Flickr website. Various images in this collection comprise more than one tag. The most useful tags are assigned to image that can clearly describe the visual contents of the image. Likewise, few images are being repeated in groups, it creates a complex image database. Total images are categorized into 19 groups; each group contains approximately 260–2,100 numbers of images. Diverse groups of MIRFLICKR 25000 are animal, baby, bird, building, car, cloud, dog, female, flower, food, male, nature, night, people, sea, sky, sunset, transport and water. This database comprises images with various sizes, so to maintain homogeneity in feature calculation all images are resized to 384×256. Fig. A.3 comprises few example images from each category of MIRFLICKR 25000 image database.

A.4 BRODATZ

Brodatz image database consists of 116 different textures. Out of 116, 109 textures have been collected from Brodatz texture photographic album [172] and 7 textures from USC database [173]. The size of each texture image is 512×512. Each image is divided into sixteen 128×128 non-overlapping subimages. Thus, each category contains 16 texture images of similar type. It constructs a database of 1856 texture images with 112 different categories. All images are in *jpeg* format. Few example images of Brodatz texture image database are comprised in Fig. A.4.

A.5 MIT VISTEX

MIT color texture database consists of 40 different real world natural textures [174]. The size of each texture is 512×512. Each texture is divided into sixteen 128×128 non-overlapping subimages. Thus, each category contains 16 texture images of similar type. It constructs a database of 640 texture images with 40 different categories. Since, similar texture images are defined from a single original one so, texture images whose visual properties do not change too much over the image are selected. All images are in *jpeg* format. Only gray version of images has been used in the experiments. Example categories of this database are bark, brick, building, fabric, flower, food, grass, leaf, metal, sand, stone, terrain, tile, water, wood etc. Fig. A.5 comprises few example images from MIT color texture database.

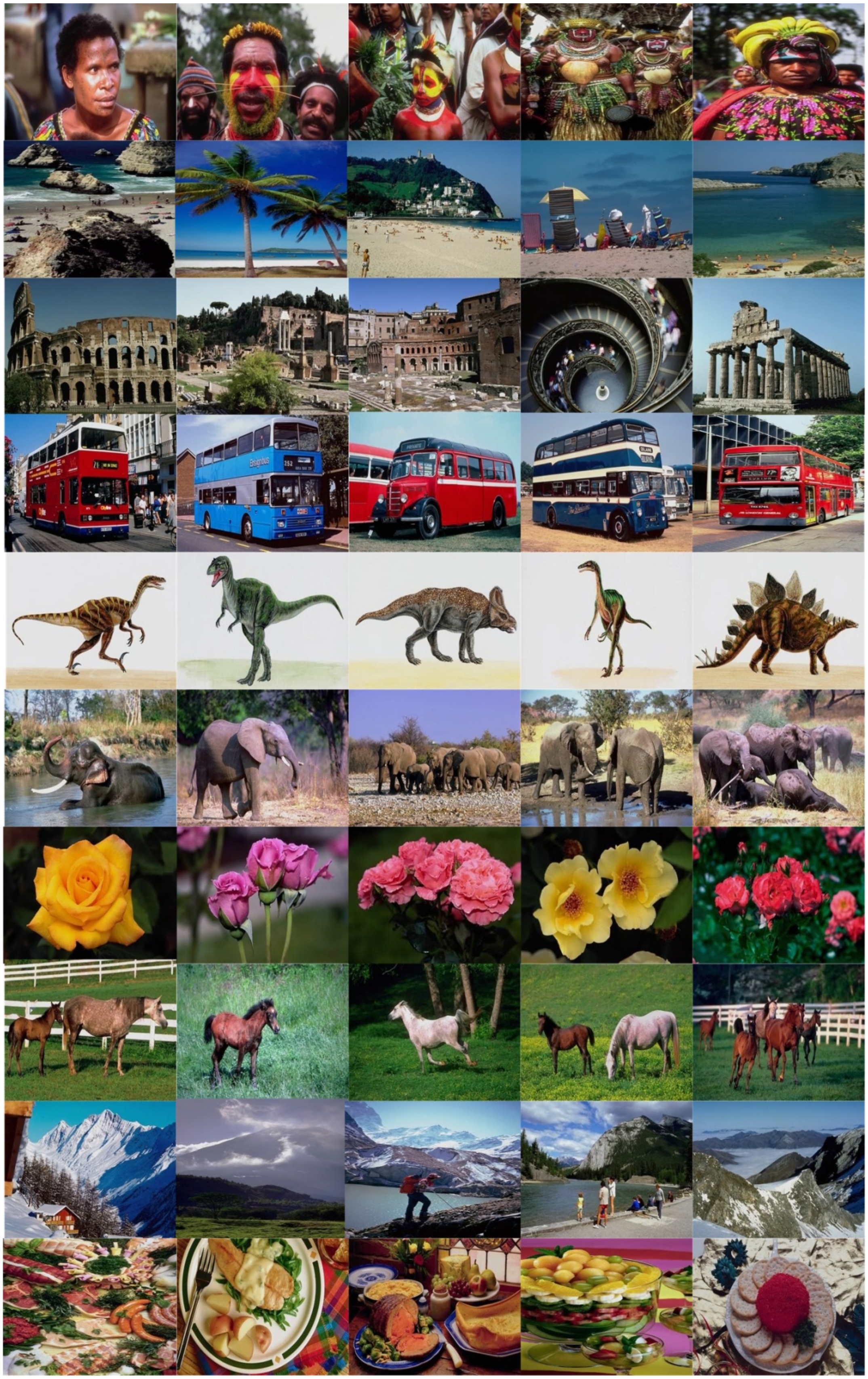


Fig. A.1: Example images from Corel 1000 database

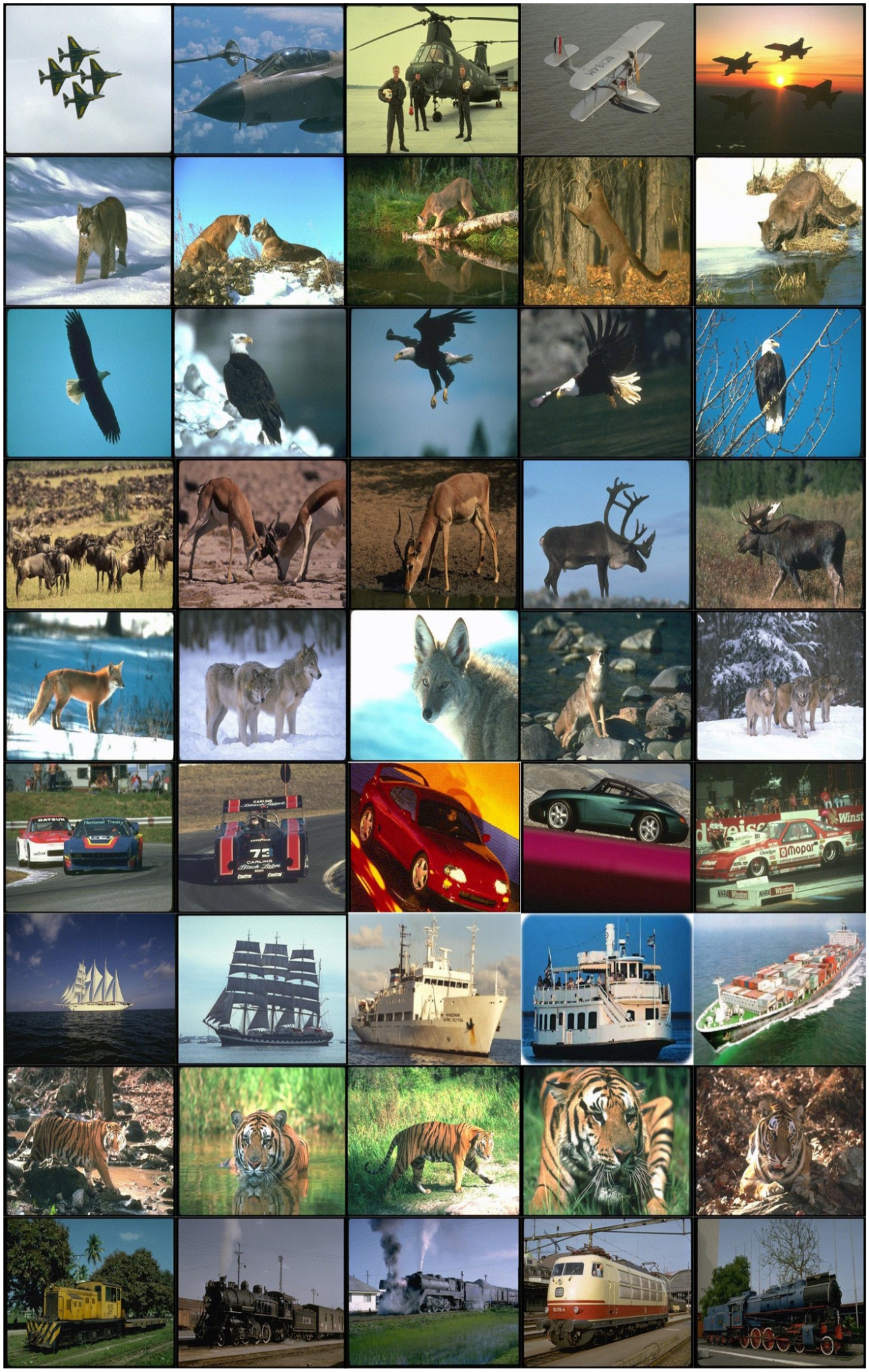


Fig. A.2: Example images from Corel 2450 database

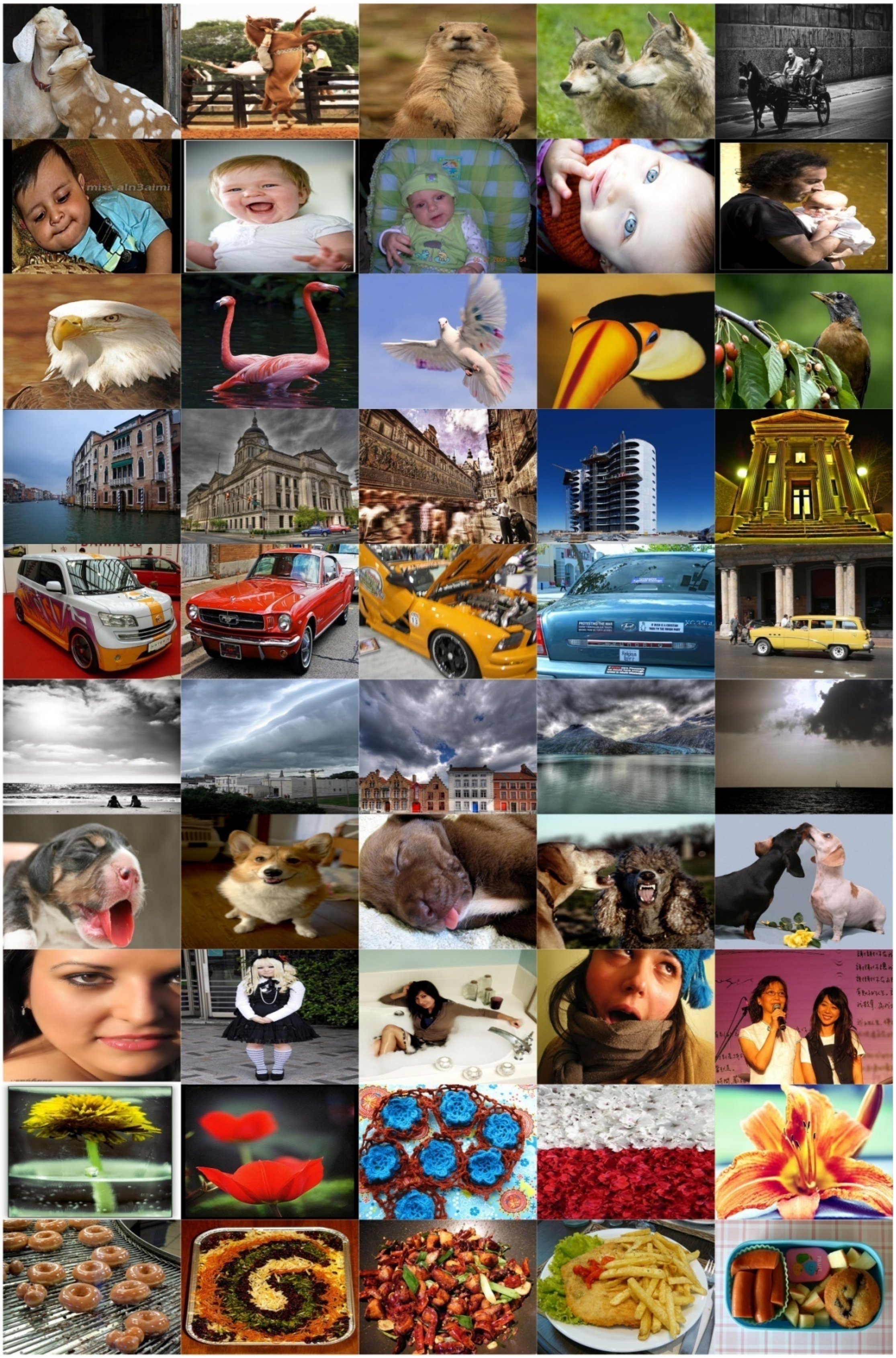


Fig. A.3: Example images from MIRFLICKR 25000 database (contd.)

Fig. A.3: Example images from MIRFLICKR 25000 database

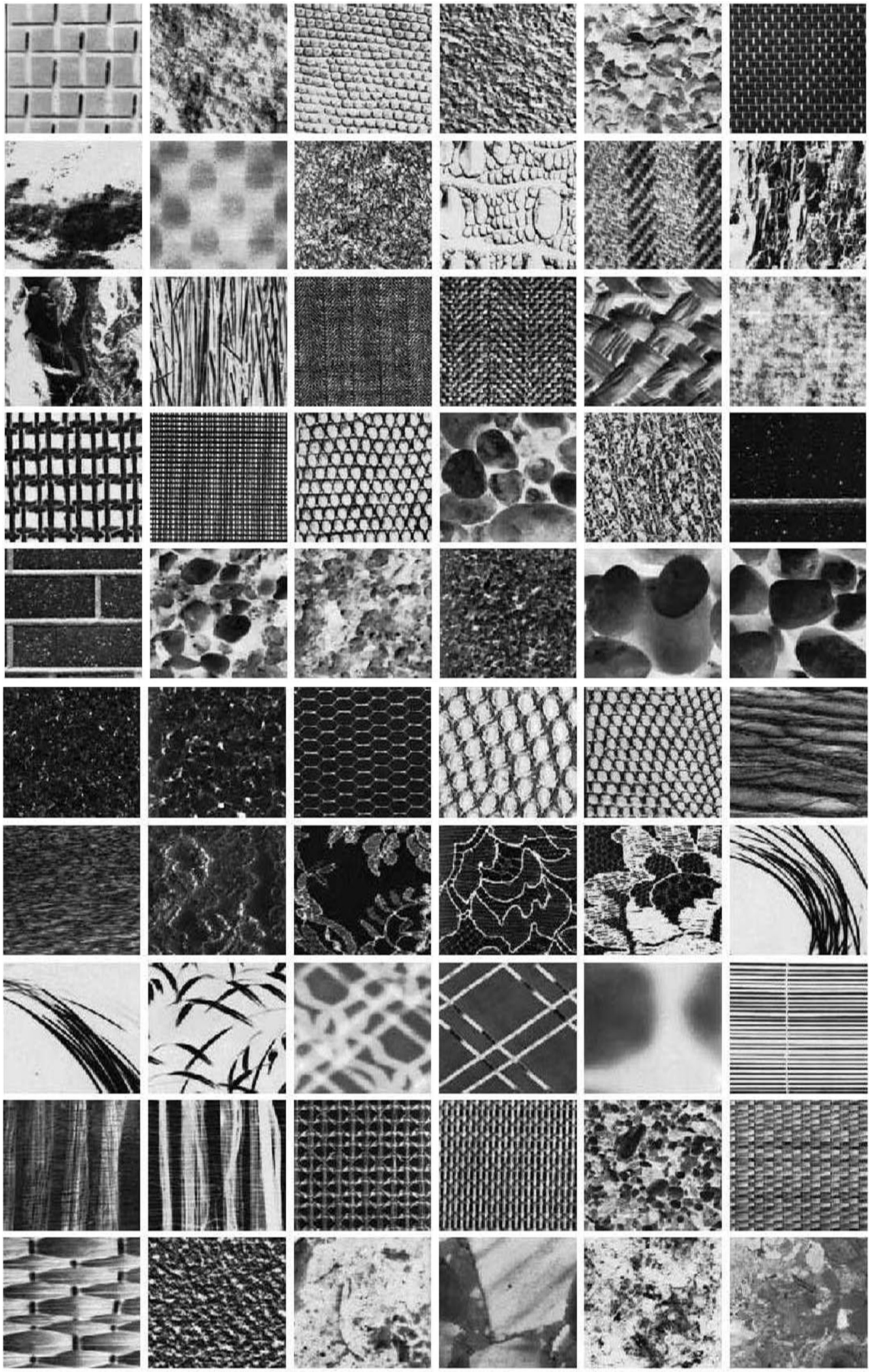


Fig. A.4: Example images from Brodatz texture database (contd.)

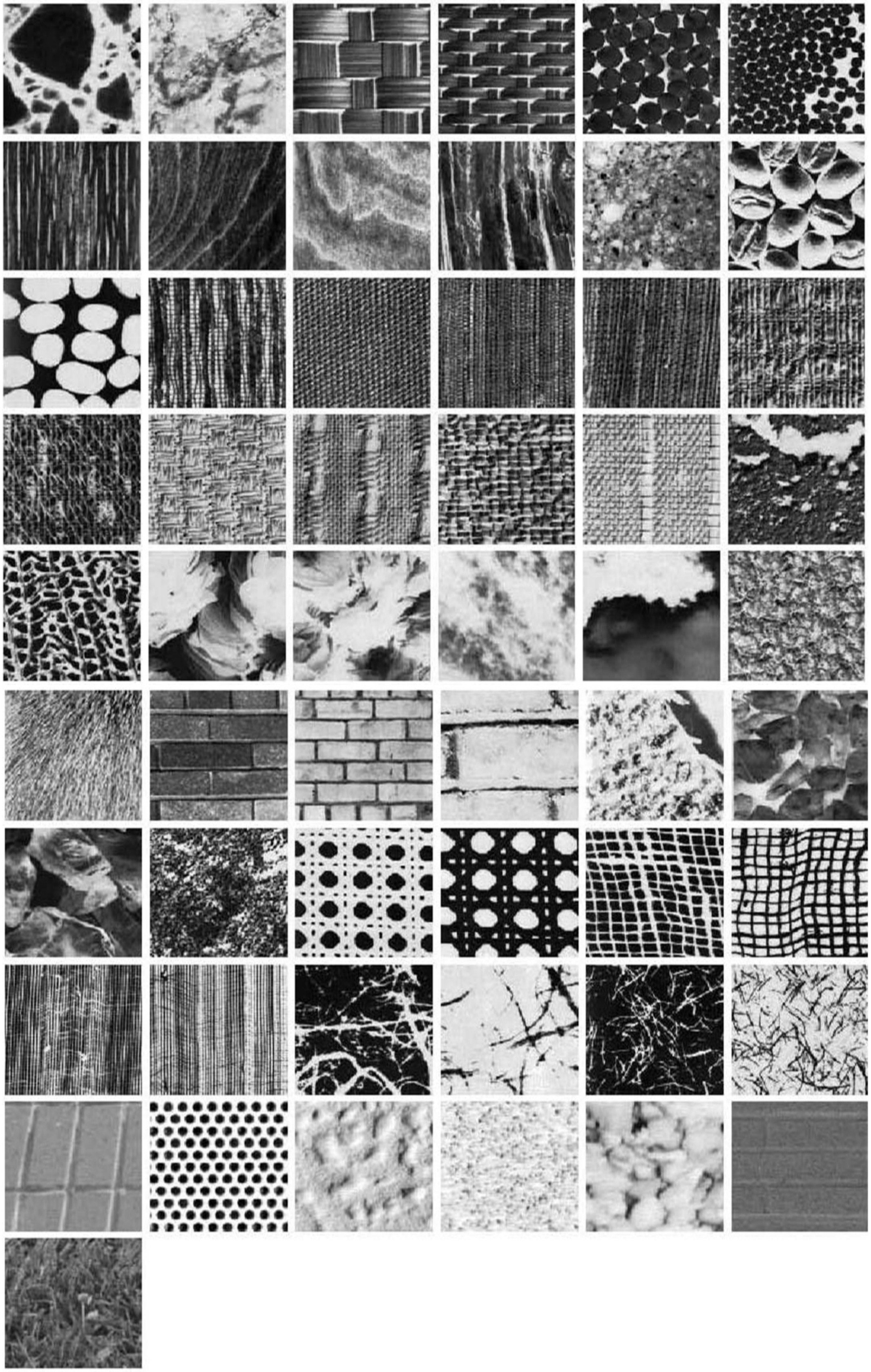


Fig. A.4: Example images from Brodatz texture database

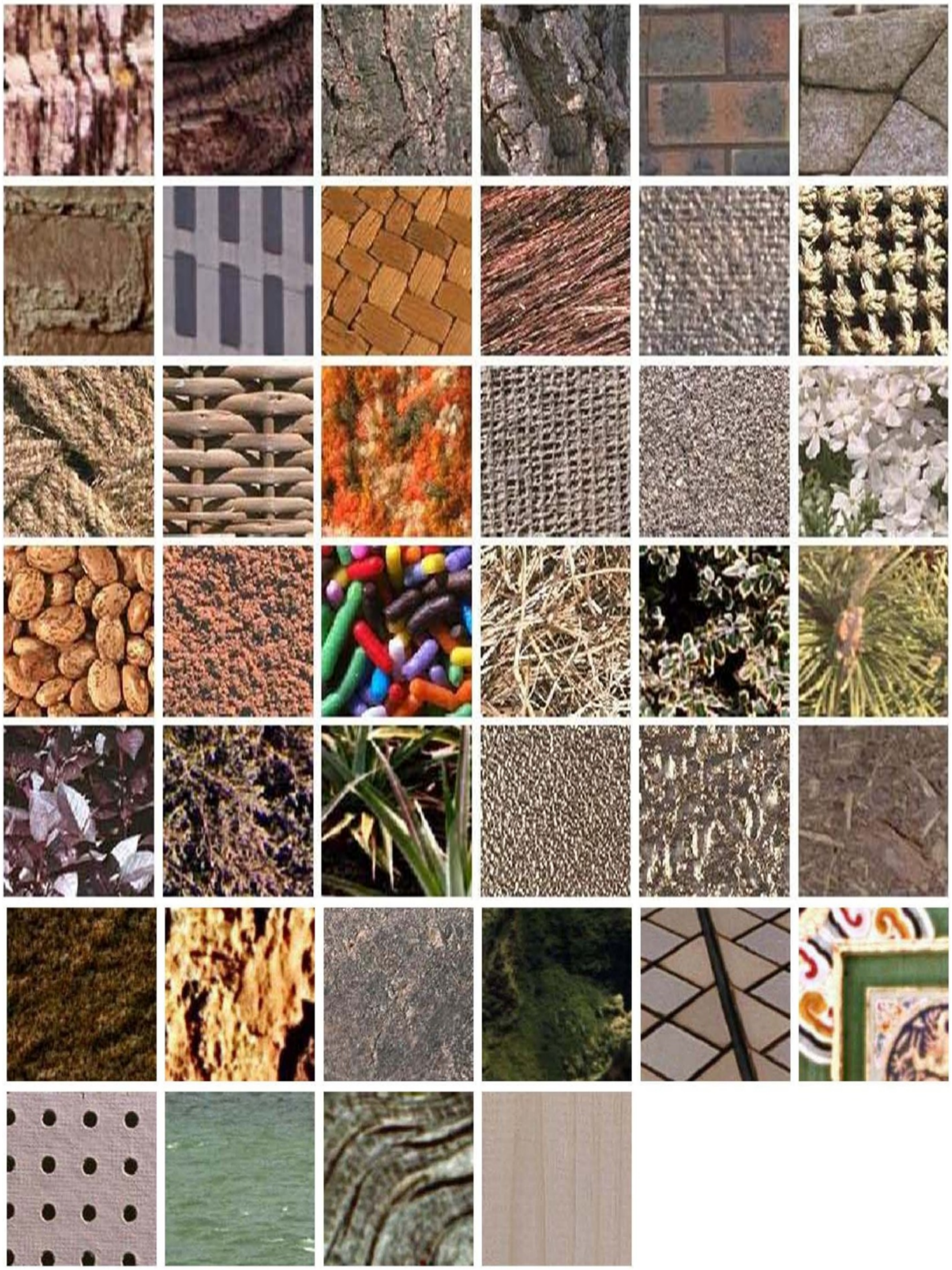


Fig. A.5: Example images from MIT color texture database