Content Based Image Retrieval

Srinivasa Kumar Devireddy
Dept. of Computer Science & Engineering, Nalanda Institute of Engineering & Technology,
Siddharth Nagar, Kantepudi(V), Sattenapalli(M), Guntur (Dt.), A.P.,India,
E-Mail: srinivaskumar_d@yahoo.com

Abstract

The importance of an effective technique in searching and retrieving images from the huge collection cannot be overemphasized. One approach for indexing and retrieving image data is using manual text annotations. The annotations can then be used to search images indirectly. But there are several problems with this approach. First, it is very difficult to describe the contents of an image using only a few keywords. Second, the manual annotation process is very subjective, ambiguous, and incomplete. Those problems have created great demands for automatic and effective techniques for content-based image retrieval (CBIR) systems. Most CBIR systems use low-level image features such as color, texture, shape, edge, etc., for image indexing and retrieval. It's because the low-level features can be computed automatically. Content Based Image Retrieval (CBIR) has emerged during the last several years as a powerful tool to efficiently retrieve images visually similar to a query image. The main idea is to represent each image as a feature vector and to measure the similarity between images with distance between their corresponding feature vectors according to some metric. Finding the correct features to represent images with, as well as the similarity metric that groups visually similar images together, are important steps in the construction of any CBIR system.

1 INTRODUCTION

The field of image retrieval has been an active research area for several decades and has been paid more and more attention in recent years as a result of the dramatic and fast increase in the volume of digital images. The development of Internet not only cause an explosively growing volume of digital images, but also give people more ways to get those images.

There were two approaches to content-based image retrieval initially. The first one is based on attribute representation proposed by database researchers where image contents are defined as a set of attributes which are extracted manually and are maintained within the framework of conventional database management systems. Queries are specified using these attributes. This obviously involves high-level of image abstraction. The second approach which was presented by image interpretation researchers depends on an integrated feature-extraction / object-recognition subsystem to overcome the limitations of attribute-based retrieval. This system automates the feature-extraction and object- recognition tasks that occur when an image is inserted into the database. This automated approaches to object recognition are computationally expensive, difficult and tend to be domain specific. There are two major categories of features. One is basic which is concerned with extracting boundaries of the image and the other one is logical which defines the image at various levels of details. Regardless of which approach is used, the retrieval in content-based image retrieval is done by color, texture, sketch, shape, volume, spatial constraints, browsing, objective attributes, subjective attributes, motion, text and domain concepts.

1.1 The growth of digital imaging: The use of images in human communication is hardly new – our cave-dwelling ancestors painted pictures on the walls of their caves, and the use of maps and building plans to convey information almost certainly dates back to pre-Roman times. But the

twentieth century has witnessed unparalleled growth in the number, availability and importance of images in all walks of life. Images now play a crucial role in fields as diverse as medicine, journalism, advertising, design, education and entertainment. Technology, in the form of inventions such as photography and television, has played a major role in facilitating the capture and communication of image data. But the real engine of the imaging revolution has been the computer, bringing with it a range of techniques for digital image capture, processing, storage and transmission which would surely have started even pioneers like John Logie Baird. The involvement of computers in imaging can be dated back to 1965, with Ivan Sutherland's Sketchpad project, which demonstrated the feasibility of computerised creation, manipulation and storage of images, though the high cost of hardware limited their use until the mid-1980s. Once computerised imaging became affordable, it soon penetrated into areas traditionally depending heavily on images for communication, such as engineering, architecture and medicine. Photograph libraries, art galleries and museums, too, began to see the advantages of making their collections available in electronic form. The creation of the World-Wide Web in the early 1990s, enabling users to access data in a variety of media from anywhere on the planet, has provided a further massive stimulus to the exploitation of digital images. The number of images available on the Web was recently estimated to be between 10 and 30 million a figure which some observers consider to be a significant underestimate.

1.2 CBIR: Content Based Image Retrieval is the retrieval of images based on visual features such as colour and texture. Reasons for its development are that in many large image databases, traditional methods of image indexing have proven to be insufficient, laborious, and extremely time consuming. These old methods of image indexing, ranging from storing an image in the database and associating it with a keyword or number, to associating it with a categorized description, have become obsolete. This is not in *CBIR*. In CBIR, each image that is stored in the database has its features extracted and compared to the features of the query image. It involves two steps:

- **Feature Extraction:** The first step in the process is extracting image features to a distinguishable extent.
- **Matching:** The second step involves matching these features to yield a result that is visually similar.

1.3 Existing System

Early techniques were not generally based on visual features but on the textual annotation of images. In other words, images were first annotated with text and then searched using a text-based approach from traditional database management systems. Text-based image retrieval uses traditional database techniques to manage images. Through text descriptions, images can be organized by topical or semantic hierarchies to facilitate easy navigation and browsing based on standard Boolean queries. However, since automatically generating descriptive texts for a wide spectrum of images is not feasible, most text-based image retrieval systems require manual annotation of images. Obviously, annotating images manually is a cumbersome and expensive task for large image databases, and is often subjective, context-sensitive and incomplete. As a result, it is difficult for the traditional text-based methods to support a variety of task-dependent queries.

1.4 Proposed System

The solution proposed is to extract the primitive features of a query image and compare them to those of database images. The image features under consideration were colour, texture and shape. Thus, using matching and comparison algorithms, the colour, texture and shape features of one image are compared and matched to the corresponding features of another image. This comparison is performed using colour, texture and shape distance metrics. In the end, these metrics are performed one after another, so as to retrieve database images that are similar to the query. The similarity between features is to be calculated using Distance algorithm.

2 Applications of CBIR

A wide range of possible applications for CBIR technology has been identified. Potentially fruitful areas include: Crime prevention, The military ,Fashion and interior design, Journalism and advertising, Medical diagnosis , Geographical information and remote sensing systems , Web searching.

Crime prevention: Law enforcement agencies typically maintain large archives of visual evidence, including past suspects' facial photographs (generally known as mug shots), fingerprints, tyre treads and shoeprints. Whenever a serious crime is committed, they can compare evidence from the scene of the crime for its similarity to records in their archives. Strictly speaking, this is an example of identity rather than similarity matching, though since all such images vary naturally over time, the distinction is of little practical significance. Of more relevance is the distinction between systems designed for verifying the identity of a known individual (requiring matching against only a single stored record), and those capable of searching an entire database to find the closest matching records.

The military: Military applications of imaging technology are probably the best-developed, though least publicized. Recognition of enemy aircraft from radar screens, identification of targets from satellite photographs, and provision of guidance systems for cruise missiles are known examples – though these almost certainly represent only the tip of the iceberg. Many of the surveillance techniques used in crime prevention could also be relevant to the military field.

Fashion and interior design: Similarities can also be observed in the design process in other fields, including fashion and interior design. Here the designer has to work within externally-imposed constraints, such as choice of materials. The ability to search a collection of fabrics to find a particular combination of colour or texture is increasingly being recognized as a useful aid to the design process.

Journalism and advertising : Both newspapers and stock shot agencies maintain archives of still photographs to illustrate articles or advertising copy. These archives can often be extremely large (running into millions of images), and dauntingly expensive to maintain if detailed keyword indexing is provided. Broadcasting corporations are faced with an even bigger problem, having to deal with millions of hours of archive video footage, which are almost impossible to annotate without some degree of automatic assistance.

Medical diagnosis : The increasing reliance of modern medicine on diagnostic techniques such as radiology, histopathology, and computerised tomography has resulted in an explosion in the number and importance of medical images now stored by most hospitals. While the prime requirement for medical imaging systems is to be able to display images relating to a named patient, there is increasing interest in the use of CBIR techniques to aid diagnosis by identifying similar past cases.

Geographical information systems (GIS) and remote sensing : Although not strictly a case of image retrieval, managers responsible for planning marketing and distribution in large corporations need to be able to search by spatial attribute (e.g. to find the 10 retail outlets closest to a given warehouse). And the military are not the only group interested in analysing satellite images. Agriculturalists and physical geographers use such images extensively, both in research and for more practical purposes, such as identifying areas where crops are diseased or lacking in nutrients – or alerting governments to farmers growing crops on land they have been paid to leave lying fallow.

Web searching: Cutting across many of the above application areas is the need for effective location of both text and images on the Web, which has developed over the last five years into an

indispensable source of both information and entertainment. Text-based search engines have grown rapidly in usage as the Web has expanded; the well-publicized difficulty of locating images on the Web indicates that there is a clear need for image search tools of similar power. Paradoxically, there is also a need for software to prevent access to images which are deemed pornographic.

3 Literature Survey

3.1 Image and Digital Imaging

It is said that one image is worth a thousand words. Visual information accounts for about 90% of the total information content that a person acquires from the environment through his sensory systems. This reflects the fact that human being relies heavily on his highly developed visual system compared with other sensory pathways. The external optical signal is perceived by eyes, and then converted into neural signal; the corresponding neural subsystem specialized for visual system is specially organized to detect subtle image features and perform high-level processing, which is further processed to generate object entities and concepts. The anatomy of the visual system explains from the structure aspect why visual information is so important to human cognition. The cognitive functions that such a system must support include the capability to distinguish among objects, their positions in space, motion, sizes, shapes, and surface texture. Some of these primitives can be used as descriptors of image content in machine vision research.

3.1.1 Taxonomy of Images

There are two formats, in which visual information can be recorded and presented – static image, and motion picture, or video. Image is the major focus of research interest in digital image processing and image understanding. Although a relatively recent development, computerized digital image processing has attracted much attention and shed lights to a broad range of existing and potential applications. This is directly caused by rapid accumulation of image data, a consequence of exponential increases of digital storage capacity and computer processing power. There are several major types of digital images depending on the elemental constituents that convey the image content. Images can take the form of:

1. Printed text and manuscript. Some examples of the kind are micro-films of old text documents, photograph of handwriting.
2. Line sketch, including diagrams, simple line graphs
3. Halftones. Images are represented by a grid of dots of variable sizes and shapes.
4. Continuous tone. Photographic images that use smooth and subtle tones.
5. Mixture of above.

Among all above, continuous tone or photographic images are most common in the digital imaging practice and of major concern of content-based image retrieval. They used to be generated by converting from other media using a scanner. The process is laboured intensive and costly. It was estimated that the image capturing and the subsequent manual indexing may account for 90 percent of the total cost of building an image database. This was the situation a few years ago. Now, digital cameras are becoming very popular and images are also being converted from analog electronic format to digital format.

At the conceptual level, an image is a representation of its target object(s). According to the Webster's 3rd New International Dictionary, an image is “the optical counterpart of an object ... a mental picture, a mental conception ...” The definition reflects the fact that an image has its content, which captures the optical or mental properties of an object; with its format varying across different kinds of media. It is determined by how the optical properties are quantized, or the degree of mental abstractions that is required. Here are some examples of image according to the above broader definition:

1. An image can be an array of pixel values stored in uncompressed bitmap digital image format. In this format, each value represents the color intensity at discrete points or pixels. A well-known example of this is Microsoft's BMP format. Although BMP format allows for pixel packing

and run-length encoding to achieve certain level of compression, its uncompressed version is more popular.

2. Popular Internet standard image formats see more extensive image transformation and compression, such as GIF and JPEG. The GIF standard defines a color degeneration process, which maps the colors in an image into no more than 256 new colors.

3. Specially defined signature file stores specifically extract images features in numeric or Boolean format, indicating the presence (or non-presence) and strength of the features. This is a very compact representation of image that is targeted for fast retrieval instead of display, archival, etc. Only the essential image features are conserved in the signature file and they are algorithm dependent. This allows for easy indexing and fast search for matching features of the query. An example of this can be found in image coding method using vector quantization (VQ), in which image blocks are coded according to a carefully chosen codebook. If the image blocks are similar to each other or the images in a set bear significant similarity, higher compression ratio can usually be achieved than the general- purpose compression algorithms such as GIF, JPEG.

4. Textual annotation can also be thought of as an instantiation of mental image, and sometimes, the descriptors can be coded by a predefined convention, or a thesaurus. The fact that two visually different images can convey the same concept and different concepts may present in images that share many similar optimal properties brings about a gap between image retrieval by content, and retrieval by concept.

3.2 Visual Features

Feature extraction plays an important role in content-based image retrieval to support for efficient and fast retrieval of similar images from image databases. Significant features must first be extracted from image data. Retrieving images by their content, as opposed to external features, has become an important operation. A fundamental ingredient for content based image retrieval is the technique used for comparing images. There are two general methods for image Comparison: intensity based (color and texture) and geometry based (shape). So we will concentrate on these features

3.2.1 Colour

One of the most important features that make possible the recognition of images by humans is colour. Colour is a property that depends on the reflection of light to the eye and the processing of that information in the brain. We use colour everyday to tell the difference between objects, places, and the time of day. Usually colours are defined in three dimensional colour spaces. These could either be *RGB* (Red, Green, and Blue), *HSV* (Hue, Saturation, and Value) or *HSB* (Hue, Saturation, and Brightness). The last two are dependent on the human perception of hue, saturation, and brightness. Most image formats such as *JPEG*, *BMP*, *GIF*, use the RGB colour space to store information. The RGB colour space is defined as a unit cube with red, green, and blue axes. Thus, a vector with three co-ordinates represents the colour in this space. When all three coordinates are set to zero the colour perceived is black. When all three coordinates are set to 1 the colour perceived is white. The other colour spaces operate in a similar fashion but with a different perception.

3.2.2 Transformation and Quantization

The color regions are perceptually distinguishable to some extent. The human eye cannot detect small color different and may perceive these very similar colors as the same color. This leads to the *quantization* of color, which means that some pre-specified colors will be present on the image and each color is mapped to some of these pre-specified colors. One obvious consequence of this is that each color space may require different levels of quantized colors, which is nothing but a different quantization scheme. In below, the effect of color quantization is illustrated. Figure (a) is the original image with *RGB* color space and (b) is the image produced after transformation into *HSV* color space and quantization. A detailed explanation of color space transformations (from *RGB* into *HSV*) and quantization can be found in

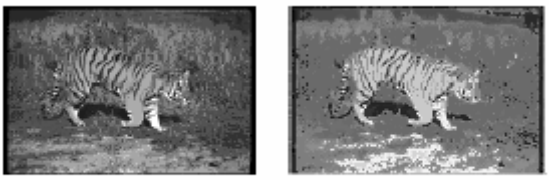


Figure 1) Transformation, Quantization of *tiger* Image. (a) Original image. (b) Image produced by applying *RGB* to *HSV* color transformation and quantization.

3.2.3 Methods of Representation

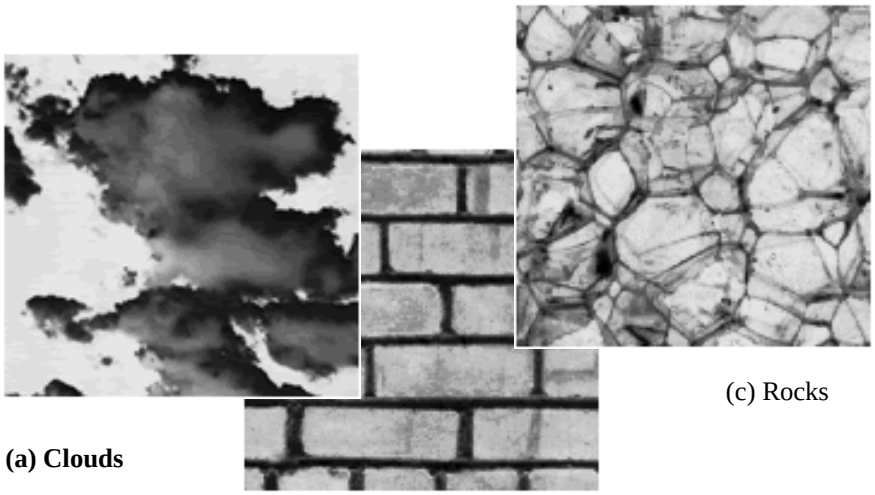
The main method of representing colour information of images in CBIR systems is through colour histograms. A colour histogram is a type of bar graph, where each bar represents a particular colour of the colour space being used. We can get a colour histogram of an image in the RGB or HSV colour space. The bars in a colour histogram are referred to as bins and they represent the x-axis. The number of bins depends on the number of colours there are in an image. The y-axis denotes the number of pixels there are in each bin. In other words how many pixels in an image are of a particular colour.

3.2.4 Color Content Extraction

One of the widely used methods for querying and retrieval by color content is color histograms. The color histograms are used to represent the color distribution in an image mainly, the color histogram approach counts the number of occurrences of each unique color on a sample image. Since an image is composed of pixels and each pixel has a color, the color histogram of an image can be computed easily by visiting every pixel once. Smith and Chang proposed colorsets as an opponent to color histograms. The colorsets are binary masks on color histograms and they store the presence of colors as 1 without considering their amounts. For the absent colors, the colorsets store 0 in the corresponding bins. The colorsets reduce the computational complexity of the distance between two images. Besides, by employing colorsets region-based color queries are possible to some extent. On the other hand, processing regions with more than two or three colors is quite complex. Another image content storage and indexing mechanism is color correlograms. It involves an easy-to-compute method and includes not only the spatial correlation of color regions but also the global distribution of local spatial correlation of colors. In fact, a color correlogram is a table each row of which is for a specific color pair of an image. The *k*-th entry in a row for color pair (*i; j*) is the probability of finding a pixel of color *j* at a distance *k* from a pixel of color *i*. The method resolves the drawbacks of the pure local and pure global color indexing methods since it includes local spatial color information as well as the global distribution of color information.

3.2.5 Texture

Texture is that innate property of all surfaces that describes visual patterns, each having properties of homogeneity. It contains important information about the structural arrangement of the surface, such as; clouds, leaves, bricks, fabric, etc. It also describes the relationship of the surface to the surrounding environment .In short, it is a feature that describes the distinctive physical composition of a surface. Texture properties include: (i) Coarseness, ii)Contrast, iii)Directionality, iv)Line-likeness, v)Regularity & v)Roughness.



(a) Clouds

(b) Bricks

(c) Rocks

Figure 2) Examples of Textures

Texture is one of the most important defining features of an image. It is characterized by the spatial distribution of gray levels in a neighbourhood. In order to capture the spatial dependence of gray-level values, which contribute to the perception of texture, a two-dimensional dependence texture analysis matrix is taken into consideration. This two-dimensional matrix is obtained by decoding the image file; jpeg, bmp, etc.

3.3 MPEG-7 Texture Descriptors

The MPEG-7 multimedia content description interface involves three texture descriptors for representing texture regions in images, namely

- **Texture browsing descriptor** to characterize perceptual directionality, regularity, and coarseness of a texture,
- **Homogeneous texture descriptor** (HTD) to quantitatively characterize homogeneous texture regions for similarity retrieval using local spatial statistics of the texture obtained by scale- and orientation-selective Gabor filtering, and
- **Edge histogram descriptor** to characterize non-homogeneous texture regions.

3.4 Distance Measure Techniques

3.4.1 Histogram Intersection Method

In the Histogram Intersection technique, two normalized histograms are intersected as a whole, as the name of the technique implies. The similarity between the histograms is a floating point number between 0 and 1. Equivalence is designated with similarity value 1 and the similarity between two histograms decreases when the similarity value approaches to 0. Both of the histograms must be of the same size to have a valid similarity value. Let $H1[1::n]$ and $H2[1::n]$ denote two histograms of size n, and $SH1;H2$ denote the similarity value between $H1$ and $H2$. Then, $SH1;H2$ can be expressed by the distance between the histograms $H1$ and $H2$ as:

$$S_{H_1,H_2} = \frac{\sum_i^n \min(H_1[i], H_2[i])}{\min(|H_1|, |H_2|)}.$$

In the system, this technique is employed for similarity calculations as a result of texture vector and color histogram comparisons between database images and query image.

4 General schema of Content Based image Retrieval

The block diagram consists of following main blocks - digitizer, feature extraction, image database, feature database, and matching and multidimensional indexing. Function of each block is as follows.

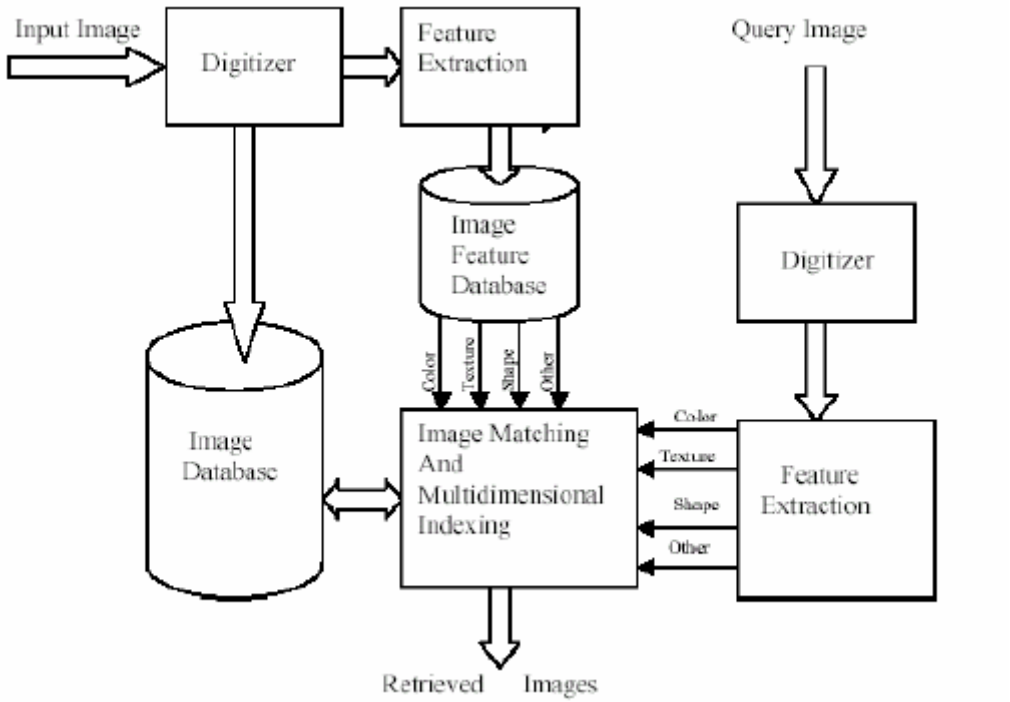


Fig 3) General Scheme of Content Based image Retrieval

Digitizer: To add new images in image database or query images which are acquired from CCD Camera, X-ray imaging system, microdensitometers, image dissectors, vision cameras etc. are needed to be digitized, so that computer can process those images.

Image Database: The Comparison between Query image and images from image database can be done directly pixel by pixel which will give precise match but on the other hand, recognizing objects entirely at query time will limit the retrieval speed of the system, due to the high expense of such computing. Generally this crude method of comparison is not used, but image database, which contains raw images, is required for visual display purpose.

Feature Extraction: To avoid above problem of pixel-by-pixel comparison next abstraction level for representing images is the feature level. Every image is characterized by a set of features such as Texture, Color, Shape and others. Extract these features at the time of injecting new image in image database. Then summarize these features in a reduced set of k indexes and store it in Image feature database. The query image is processed in the same way as images in the database. Matching is carried out on the feature database.

Image matching and Multidimensional Indexing: Extracted features of query image are compared with features, which are stored in image feature database. To achieve fast retrieval speed and make the retrieval system truly scalable to large size image collections an effective multidimensional indexing is indispensable part of the whole system. The system selects the N images having the greatest overall similarities to the query image.

5 Theoretical Analysis

The basic idea behind content- based image retrieval is that, when building an image database, or retrieving an image from the database, we first extract feature vectors from images (the

features can be color, shape, and texture), then store the vectors in another database for future use. When given a query image, we similarly extract its feature vectors, and match these vectors with those already in the database, if the distance between two images feature vectors is small enough; we consider the corresponding image in the database match the query. The search is usually based on similarity rather than on exact match, and the retrieval results are then ranked according to a similarity index. Usually, a group of similar target images are presented to users.

Recent efforts in image retrieval applications have focused on indexing a few specific visual dimensions of the images, such as color, texture, shape, motion and spatial information. However without integrating these visual dimensions, the current content-based techniques have limited capacity to satisfactorily retrieve images. Many difficult problems in image retrieval systems remain to be investigated. The explosive proliferation of "unconstrained" digital imagery in the form of images, graphics, and videos, due to the improved accessibility of computer technology and the main-stream acceptance of the world-Wide Web as a variable medium for publishing, advertising and communicating has created an immediate need for efficient and effective tools for cataloguing, indexing, managing, compressing and searching for the "unconstrained" visual information. A significant gap exists between the ability of computers to analyze images and videos at the feature level (colors, textures, shapes) compared to the inability at the semantic-level (objects, scenes, people, moods, artistic value). Closing this gap by improving technologies for image understanding would certainly improve the image search and retrieval systems. However, the solution of this "large" problem remains distant while much of the current technology consists of solutions to "small" problems (for example, image segmentation, color constancy, shape from texture, video shot detection, and so forth). However there are new opportunities to climb the semantic ladder in today's divers media enjoinments. The final difficulty limiting progress in image retrieval concerns system evaluation. Unless there are reliable and widely accepted ways of measuring effectiveness of new technique, it will be impossible to judge whether they represent any advancement on existing methods. This will inventible limit the progress. For an image retrieval system to be successful for geographical images, we need to develop approaches robust in spatial pattern characterization, rotation invariant, and taking into account of the scale effect. This project can be thought of having two phases of operations. The first one consists of different types of image storage techniques, the second one is having the retrieving the image from existing database.

What kinds of query are users likely to put to an image database? To answer this question in depth requires a detailed knowledge of user needs – why users seek images, what use they make of them, and how they judge the utility of the images they retrieve. As we show in below, not enough research has yet been reported to answer these questions with any certainty. Common sense evidence suggests that still images are required for a variety of reasons, including:

- illustration of text articles, conveying information or emotions difficult to describe in words,
- display of detailed data (such as radiology images) for analysis,
- formal recording of design data (such as architectural plans) for later use.

Access to a desired image from a repository might thus involve a search for images depicting specific types of object or scene, evoking a particular mood, or simply containing a specific texture or pattern. Potentially, images have many types of attribute which could be used for retrieval, including:

- the presence of a particular combination of colour, texture or shape features (e.g. green stars);
- the presence or arrangement of specific types of object (e.g. chairs around a table);
- the depiction of a particular type of event (e.g. a football match);
- the presence of named individuals, locations, or events (e.g. the Queen greeting a crowd);
- subjective emotions one might associate with the image (e.g. happiness);

- metadata such as who created the image, where and when.

Level 1 comprises retrieval by primitive features such as colour, texture, shape or the spatial location of image elements. Examples of such queries might include “find pictures with long thin dark objects in the top left-hand corner”, “find images containing yellow stars arranged in a ring” – or most commonly “find me more pictures that look like this”. This level of retrieval uses features which are both objective, and directly derivable from the images themselves, without the need to refer to any external knowledge base. Its use is largely limited to specialist applications such as trademark registration, identification of drawings in a design archive, or colour matching of fashion accessories.

Level 2 comprises retrieval by derived (sometimes known as logical) features, involving some degree of logical inference about the identity of the objects depicted in the image. It can usefully be divided further into:

1. retrieval of objects of a given type (e.g. “find pictures of a double-decker bus”);
2. retrieval of individual objects or persons (“find a picture of the Eiffel tower”).

To answer queries at this level, reference to some outside store of knowledge is normally required – particularly for the more specific queries at level 2(b). In the first example above, some prior understanding is necessary to identify an object as a bus rather than a lorry; in the second example, one needs the knowledge that a given individual structure has been given the name “the Eiffel tower”. Search criteria at this level, particularly at level, are usually still reasonably objective.

Level 3 comprises retrieval by abstract attributes, involving a significant amount of high-level reasoning about the meaning and purpose of the objects or scenes depicted. Again, this level of retrieval can usefully be subdivided into:

1. retrieval of named events or types of activity (e.g. “find pictures of Scottish folk dancing”);
2. retrieval of pictures with emotional or religious significance (“find a picture depicting suffering”).

Success in answering queries at this level can require some sophistication on the part of the searcher. Complex reasoning, and often subjective judgement, can be required to make the link between image content and the abstract concepts it is required to illustrate. Queries at this level, though perhaps less common than level 2, are often encountered in both newspaper and art libraries.

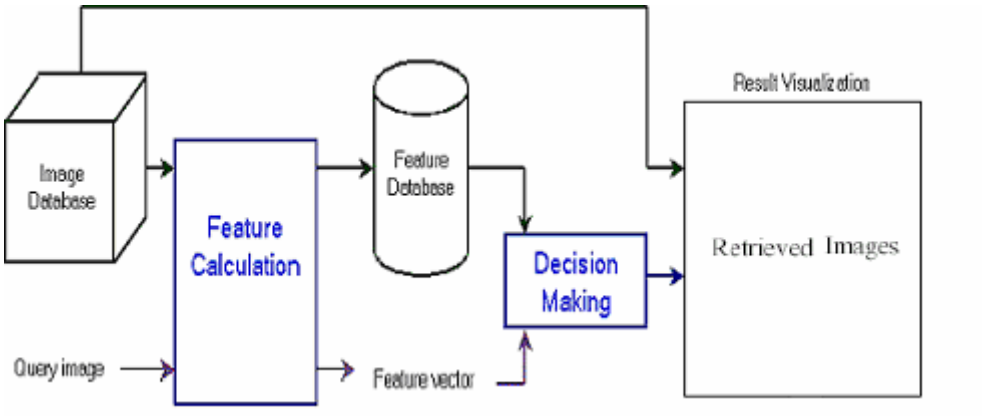


Fig 4) Architecture of CBIR

6 IMPLEMENTATION DETAILS

6.1 Color Histogram calculation

We define color histograms as a set of bins where each bin denotes the probability of pixels in the image being of a particular color. A color histogram H for a given image is defined as a vector:

H = {H[0],H[1],H[2],……..,H[i],…H[n]}

Where 'i' represents color in the color histogram[i] is the Number of pixels in color 'i' in that image and 'n' is the number of bins in the color histogram.Typically, each pixel in an image will be assigned to a bin of a color histogram of that image, so for the color histogram of an image, the value of each bin is the number of pixels that has the same corresponding color. In order to compare images of different sizes, color histograms should be normalized. The normalized color histogram 'H' is defined as:

H' = {H'[0],H'[1],H'[2],……..,H'[i],…H'[n]} Where H'[i]=H'[i]/p, p is the total number of pixels in an image. An ideal color space quantization presumes that distinct colors should not be located in the same sub-cube and similar colors should be assigned to the same sub-cube. Using few colors will decrease the possibility that similar colors are assigned to different bins, but it increases the possibility that distinct colors are assigned to the same bins, and that the information content of the images will decrease by a greater degree as well. On the other hand, color histograms with a large number of bins will contain more information about the content of images, thus decreasing the possibility of distinct colors will be assigned to the same bins. However, they increase the possibility that similar colors will be assigned to different bins, the storage space of metadata, and the time for calculating the distance between color histograms. Therefore, there is a trade-off in determining how many bins should be used in color histograms.

6.2. Converting RGB to HSV

The value is given by $V = \frac{R+G+B}{3}$.

Where the quantities R, G and B are the amounts of the red, green and blue Components normalized to the range [0; 1]. The value is therefore just the average of the red, green and blue components.

The saturation is given by $S = 1 - \frac{\min(R,G,B)}{I} = 1 - \frac{3}{R+G+B}\min(R,G,B)$.

Where the min(R; G;B) term is really just indicating the amount of white Present. If any of R, G or B are zero, there is no white and we have a pure colour. The hue is given by

$$H = \cos^{-1}\left(\frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\left[(R-G)^2+(R-B)(G-B)\right]^{\frac{1}{2}}} \right).$$

6.3. Edge Histogram calculation

Edges in images constitute an important feature to represent their content. Also, human eyes are sensitive to edge features for image perception. One way of representing such an important edge feature is to use a histogram. An edge histogram in the image space represents the frequency and the directionality of the brightness changes in the image. It is a unique feature for images. To represent this unique feature, in MPEG-7, there is a descriptor for edge distribution in the image. This Edge Histogram Descriptor (EHD) proposed for MPEG-7 expresses only the local edge distribution in the image. That is, since it is important to keep the size of the descriptor as compact as possible for efficient storage of the metadata, the MPEG-7 edge histogram is designed to contain only 80 bins describing the local edge distribution. These 80 histogram bins are the only standardized semantics for the MPEG-7 EHD. However, using the local histogram bins only may not be sufficient to represent global features of the edge distribution. Thus, to improve the retrieval performance, we need global edge distribution as well.

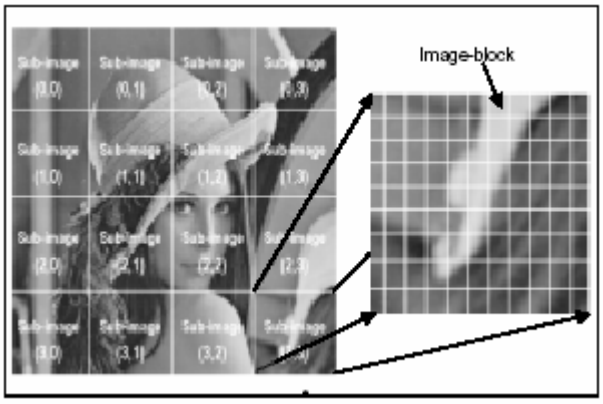


Figure 5) Definition of sub-image and image-block

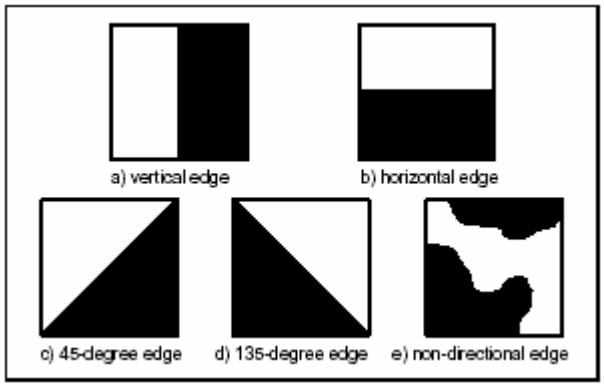


Figure 6) Five Edge Types

The EHD basically represents the distribution of 5 types of edges in each local area called a sub-image. As shown in Fig. 5, the sub-image is defined by dividing the image space into 4×4 no overlapping blocks. Thus, the image partition always yields 16 equal-sized sub-images regardless of the size of the original image. To characterize the sub-image, we then generate a histogram of edge distribution for each sub-image. Edges in the sub-images are categorized into 5 types: vertical, horizontal, 45-degree diagonal, 135-degree diagonal and non-directional edges (Fig. 6). Thus, the histograms for each sub-image represent the relative frequency of occurrence of the 5 types of edges in the corresponding sub-image. As a result, as shown in Fig. 1, each local histogram contains 5 bins. Each bin corresponds to one of 5 edge types. Since there are 16 sub-images in the image, a total of 5×16=80 histogram bins is required. Note that each of the 80-histogram bins has its own semantics in terms of location and edge type. The semantics of the histogram bins form the normative part of the MPEG-7 standard descriptor. Specifically, starting from the sub-image at (0,0) and ending at (3,3), 16 sub-images are visited in the raster scan order and corresponding local histogram bins are arranged accordingly. Within each sub image, the edge types are arranged in the following order: vertical, horizontal, 45-degree diagonal, 135-degree diagonal, and non directional. Table 1 summarizes the complete semantics for the EHD with 80 histogram bins. Of course, each histogram bin value should be normalized and quantized. For normalization, the number of edge occurrences for each bin is divided by the total number of image-blocks in the sub-image.

Histogram bins	**Semantics**
BinCounts[0]	Vertical edge of sub-image at (0,0)
BinCounts[1]	Horizontal edge of sub-image at (0,0)
BinCounts[2]	45 degree edge of sub-image at (0,0)
BinCounts[3]	135 degree edge of sub-image at (0,0)
BinCounts[4]	Non-directional edge of sub-image at (0,0)
BinCounts[5]	Vertical edge of sub-image at (0,1)
:	:
BinCounts[74]	Non-directional edge of sub-image at (3,2)
BinCounts[75]	Vertical edge of sub-image at (3,3)
BinCounts[76]	Horizontal edge of sub-image at (3,3)
BinCounts[77]	45 degree edge of sub-image at (3,3)
BinCounts[78]	135 degree edge of sub-image at (3,3)
BinCounts[79]	Non-directional edge of sub-image at (3,3)

Table-1) Semantics of Local Edge bins

Image block is a basic unit for extracting the edge information. That is, for each image-block, we determine whether there is at least an edge and which edge is predominant. When an edge exists, the predominant edge type among the 5 edge categories is also determined. Then, the

histogram value of the corresponding edge bin increases by one. Otherwise, for the monotone region in the image, the image-block contains no edge. In this case, that particular image-block does not contribute to any of the 5 edge bins. Consequently, each image-block is classified into one of the 5 types of edge blocks or a non edge block. Although the non edge blocks do not contribute to any histogram bins, each histogram bin value is normalized by the total number of image-blocks including the non edge blocks. This implies that the summation of all histogram bin values for each sub-image is less than or equal to 1. This in turn, implies that the information regarding non edge distribution in the sub-image (smoothness) is also indirectly considered in the EHD. Now, the normalized bin values are quantized for binary representation. Since most of the values are concentrated within a small range (say, from 0 to 0.3), they are nonlinearly quantized to minimize the overall number of bits Since the EHD describes the distribution of non-directional edges and non edge cases as well as four directional edges, the edge extraction scheme should be based on the image-block as a basic unit for edge extraction rather than on the pixel. That is, to extract directional edge features, we need to define small square image-blocks in each sub-image as shown in Fig. 5. Specifically, we divide the image space into non overlapping square image-blocks and then extract the edge information from them. Note that, regardless of the image size, we divide the image space into a fixed number of image-blocks . The purpose of fixing the number of image-blocks is to cope with the different sizes (resolutions) of the images. That is, by fixing the number-of-image blocks, the size of the image block becomes variable and is proportional to the size of the whole image. The size of the image-block is assumed to be a multiple of 2. Thus, it is sometimes necessary to ignore the outmost pixels in the image to satisfy that condition. Simple method to extract an edge feature in the image block is to apply digital filters in the spatial domain. To this end, we first divide the image-block into four sub-blocks. Then, by assigning labels for four sub-blocks from 0 to 3, we can represent the average gray levels for four sub-blocks at (i,j)th image-block as $a0(i,j)$, $a1(i,j)$, $a2(i,j)$, and $a3(i,j)$, respectively. Also, we can rep-resent the filter coefficients for vertical, horizontal, 45-degree diagonal, 135-degree diagonal, and non-directional edges as $fv(k)$, $fh(k)$, $fd\text{-}45(k)$, $fd\text{-}135(k)$, and $fnd(k)$, respectively, where $k=0,12,,3$ represents the location of the sub-blocks. Now, the respective edge magnitudes $m_v(i,j)$, $m_h(i,j)$, $m_{d\text{-}45}(i,j)$, $m_{d\text{-}135}(i,j)$, and $m_{nd}(i,j)$ for the (i,j)th image-block can be obtained as follows:

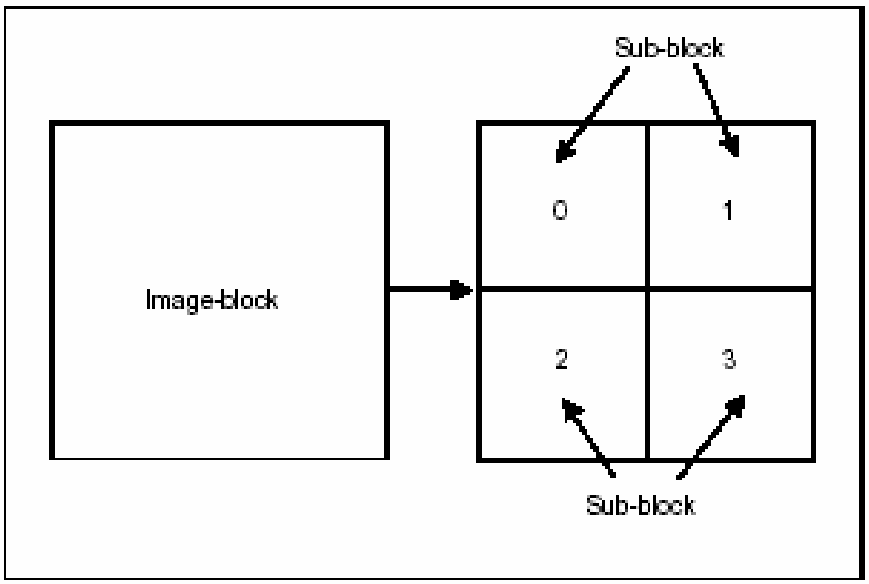


Figure 7) Sub-blocks and their labeling

$$m_v(i,j)=\left|\sum_{k=0}^{3} a_k(i,j)\times f_v(k)\right| \quad (1)$$

$$m_h(i,j)=\left|\sum_{k=0}^{3} a_k(i,j)\times f_h(k)\right| \quad (2)$$

$$m_{d-45}(i,j)=\left|\sum_{k=0}^{3} a_k(i,j)\times f_{d-45}(k)\right| \quad (3)$$

$$m_{d-135}(i,j)=\left|\sum_{k=0}^{3} a_k(i,j)\times f_{d-135}(k)\right| \quad (4)$$

$$m_{nd}(i,j)=\left|\sum_{k=0}^{3} a_k(i,j)\times f_{nd}(k)\right| \quad (5)$$

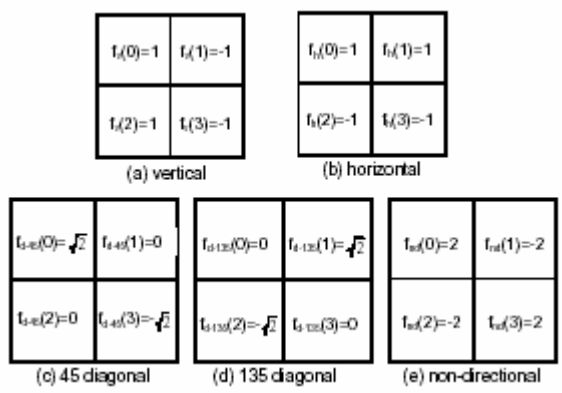


Figure 8) Filter Coefficients for edge detection

The maximum value among 5 edge strengths obtained from (1) to (5) in the image-block is considered to have the corresponding edge in it. Otherwise, the image-block contains no edge.

$$\max\{m_v(i,j), m_h(i,j), m_{d-45}(i,j), m_{nd}(i,j)\} \quad .....(6)$$

6.4 Distance Measures

We tried 3 kinds of histogram distance measures for a histogram H(i), i=1,2,.., N

6.4.1 L-2 Distance Defined as:

$$d(q,t)=\left[\sum_{m=1}^{N}(h_q(m)-h_t(m))^2\right]^{\frac{1}{2}}$$

this Metric is uniform in terms of the Euclidian distance between vectors in feature space, but the vectors are not normalized to unit length (infact, they are on a hyperlane if the histogram is normalized).

6.4.2 Cosine Distance

If we normalize all vectors to unit length, and look at the angle between them, we have cosine distance, defined as:

$$d(q,t)=\frac{2}{\pi}\cos^{-1}\left(\frac{\sum_{m=1}^{N} h_q(m)h_t(m)}{\min(\|h_q\|,\|h_t\|)}\right)$$

6.4.3 Histogram Intersection

Defined as:

$$d'_{q,t}=1-\frac{\sum_{m=0}^{M-1}\min(h_q[m],h_t[m])}{\min(|\mathbf{h}_q|,|\mathbf{h}_t|)}.$$

The denominator term is needed for non-normalized histogram features

6.5 Combining Features and Making Decisions

This section formulates the problem of combining different features as a problem of finding a set of weights of different feature distances, and then presents a ***Mini-Max*** algorithm in finding the best-matching image,

Mini-Max Combination

Let's first look at a more general case, where we have: query image ***q***, images in the database ***i, K*** features and thus ***K*** kinds of distances (they are constants)

$$d_k(q,i), k = 1,..,K, i = 1,..,N$$

Assume we are going to combine them as a weighted sum of all the distances, i.e. the distance for a image in the database is written as:

$$D(q,i) = \sum_{k=1}^{K} w_k d_k(q,i) \quad .....(1) \quad \text{and} \quad \sum_{k=1}^{K} w_k = 1, w_k \geq 0, \forall k = 1,2,..,K \quad .....(2)$$

Now we want to search for a vector ***w*** that satisfies Eq.(2) and the resulting distance measure is "*most close*" to our subjective criteria. There are two candidate approaches:

i) assign a set of weights based on the perceptually judgment of the designer on some image set (training). But the problem here is that this set of weights may perform poorly on new dataset.

ii) Or, having no assumption about the subjective judgment of a user, we choose the image that minimizes the maximum distance over all valid set of weights as the best match(denoted as ***Mini-Max*** hereafter). For every image ***i***, searching for the maximum distance over the weight space turns our to be a linear program, an thus have fast solution:

Maximize: (1), Subject to (2).Where all ***d***s are the constants and $w_k, k = 1..,K$ are unknown. The image with the miminum "max-distance" is declared as the best match to the query image. For our 2 features case the max distance

$$D(q,i) = wd_c(q,i) + (1-w)d_e(q,i), 0 \leq w \leq 1$$

of every image i, is a linear function of ***w*** over [0,1]. Thus the maximum either lies at ***w=0*** or ***w=1***, and comparing $d_c(q,i)$ and $d_e(q,i)$ is sufficient. Then we rank the maximum of $d_c(q,i)$ and $d_e(q,i)$ for all ***i***, and take ***n*** images with the least distance as our return result.

7 TESTING

7.1 Recall and Precision Evaluation:

Testing the effectiveness of the content based image Retrieval about testing how well the CBIR can retrieve similar images to the query image and how well the system prevents the return results that are not relevant to the source at all in the user point of view. A sample query image must be selected from one of the image category in the database. When the system is run and the result images are returned, the user needs to count how many images are returned and how many of the returned images are similar to the query image. Determining whether or not two images are similar is purely up to the user's perception. Human perceptions can easily recognise the similarity between two images although in some cases, different users can give different opinions. After images are retrieved, the system's effectiveness needs to be determined. To achieve this, two evaluation measures are used. The first measure is called **Recall**. It is a measure of the ability of a system to present all relevant items. The equation for calculating recall is given below:

$$\text{Re}call = \frac{Number\ of\ relevant\ items\ \text{Re}trieved}{Number\ of\ relevant\ items\ in\ Collection}$$

The second measure is called **Precision**. It is a measure of the ability of a system to present only relevant items. The equation for calculating precision is given below.

$$Precision = \frac{Number\ of\ relevant\ items\ retrieved}{Total\ number\ of\ items\ retrieved}$$

The number of relevant items retrieved is the number of the returned images that are similar to the query image in this case. The number of relevant items in collection is the number of images that are in the same particular category with the query image. The total number of items retrieved is the number of images that are returned by the system.

7.2 Test cases

The CBIR implementation consists of various sub-systems for color and texture that are built out of modules, which are composed of procedures and functions. The testing process hence consists of different stages, where testing is carried out incrementally in conjunction with system implementation. We have conducted several color and texture experiments on the test database of 250 images. The sample results of these experiments are given as follows:

7.2.1 Color test

Test	Total Number of Relevant Images	Number of relevant images Retrieved	Total Number of Retrieved Images	Recall	Precision
1	46	29	45	63.04	64.44
2	38	30	44	78.94	68.00
3	18	12	20	66.66	80.00
4	30	25	36	83.33	69.44
5	32	23	35	71.80	65.70

Table 2) Color Test cases

7.2.2 Texture Test

Test	Total Number of Relevant Images	Number of relevant images Retrieved	Total Number of Retrieved Images	Recall	Precision
1	46	19	45	41.30	42.22
2	38	28	44	73.68	63.64
3	18	11	20	61.11	55.00
4	30	22	36	73.33	61.11
5	32	19	35	59.37	54.28

Table 3) Texture Test cases

7.2.3 Color and Texture Test

Test	Total Number of Relevant Images	Number of relevant images Retrieved	Total Number of Retrieved Images	Recall	Precision
1	46	35	45	76.08	77.77
2	38	33	44	86.84	75.00
3	18	15	20	83.33	75.00
4	30	26	36	86.67	72.22
5	32	27	35	84.38	77.14

Table 4) Color and Texture Test cases

8 CONCLUSION

The dramatic rise in the sizes of images databases has stirred the development of effective and efficient retrieval systems. The development of these systems started with retrieving images using textual annotations but later introduced image retrieval based on content. This came to be known as

CBIR or Content Based Image Retrieval. Systems, using CBIR we can retrieve images based on visual features such as colour, texture and shape, as opposed to depending on image descriptions or textual indexing. In this paper we proposed an image retrieval system that evaluates the similarity of each image in its data store to a query image in terms of color and textural characteristics, and returns the images within a desired range of similarity. From among the existing approaches to color and texture analysis within the domain of image processing, we have adopted the MPEG-7 edge histogram and color histogram to extract texture and color features from both the query images and the images of the data store. For distance measuring between histogram vectors of two images, we have tried three distance measures, they are l2 distance measure, cosine distance measure and histogram intersection distance measure. The experiment results showing that, the average relevant image retrieval rate is increased by 10% by combined features of color and texture than considering features individually.

9 REFERENCES

[1] Efficient use of MPEG-7 edge histogram descriptor- chee sun won, Dong kown park and soo-jun park ETRI Journal, volume 24, number 1, February 2002.

[2] Ak jain and a vailaya " Image retrieval using color and shape, "Pttern recogn vol 29 no 8, 1966 pp1233-1244

[3] Fundamentals of content based image retrieval Dr. fuhui long, Dr. Hongjiang and Prof. David Dagan Feng

[4] Lars Jacob hove, Extending image retrivel systems with a thesaurus for shapes, Institute for information and media sciences, University of Bergen

[5] Lu, G (1999) multimedia Database Management Systems. Norwood, Artech House INC

[6] Manjunath, B.S. and W.-y.Ma(2002) Texture Features for Image Retrieval. Image Databases:Search and Retrieval of Digital Imagery V. castelli and R.Baeza-Yates, John Wiley&sons, Inc:313-344

[7] K. Arbter, W. E. Snyder, H. Burkhardt, and G. Hirzinger, "Application of affine-invariant Fourier descriptors to recognition of 3D objects," *IEEE Trans. Pattern Analysis and MachineIntelligence*, vol. 12, pp. 640-647, 1990.

[8] E. M. Arkin, L.P. Chew, D..P. Huttenlocher, K. Kedem, and J.S.B. Mitchell, "An efficiently computable metric for comparing polygonal shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 209-226, 1991.

[9] J. Assfalg, A. D. Bimbo, and P. Pala, "Using multiple examples for content-based retrieval," *Proc.Int'l Conf. Multimedia and Expo*, 2000.

[10] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C. F. Shu, "The virage image search engine: An open framework for image management," *In Proc. SPIC Storage and Retrieval for Image and Video Database*, Feb. 1996.

[11] N. Beckmann, *et al*, "The R*-tree: An efficient robust access method for points and rectangles," *ACM SIGMOD Int. Conf. on Management of Data*, Atlantic City, May 1990.

[12] A. Blaser, Database Techniques for Pictorial Applications, *Lecture Notes in Computer Science*,Vol.81, Springer Verlag GmbH, 1979.

[13] P. Brodatz, "Textures: A photographic album for artists & designers," Dover, N Y, 1966.

[14] H. Burkhardt, and S. Siggelkow, "Invariant features for discriminating between equivalence classes," *Nonlinear Model-based Image Video Processing and Analysis*, John Wiley and Sons, 2000.

[15] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik, "Blobworld: A system for region-based image indexing and retrieval," In D. P. Huijsmans and A. W. M. Smeulders, ed. *Vi s u al Information and Information System, Proceedings of the Third International Conference VISUAL'99*, Amsterdam, The Netherlands, June 1999, Lecture Notes in Computer Science 1614.Springer, 1999.

[16] J.A. Catalan, and J.S. Jin, "Dimension reduction of texture features for image retrieval using hybrid associative neural networks," *IEEE International Conference on Multimedia and Expo*, Vol.2, pp. 1211 -1214, 2000.

[I7]A. E. Cawkill, "The British Library's Picture Research Projects: Image, Word, and Retrieval,"*Advanced Imaging*,Vol.8, No.10, pp.38-40, October 1993.

[18] N. S. Chang, and K. S. Fu, "A relational database system for images," *Technical Report TR-EE 79-82*, Purdue University, May 1979.

Article received: 2008-12-18