Retrieval of Digital Images Based On Multi-Feature Similarity Using Genetic Algorithm

K. APARNA
Department of Electronics and Communication Engineering, JNTUH, Hyderabad-500085.

ABSTRACT

Conventional relevance feedback schemes may not be suitable to all practical applications of content based image retrieval (CBIR), since most ordinary users like to complete their search in a single interaction, especially on web search. In this paper, we explore a new approach based on multi-feature similarity score fusion using genetic algorithm. Single feature describes image content only from one point of view, which has a certain one-sided. Fusing multifeature similarity score is expected to improve the system's retrieval performance. In this paper, the retrieval results from color feature and texture feature are analyzed, and the method of fusing multi-feature similarity score is described. For the purpose of assigning the fusion weights of multi-feature similarity scores reasonably, the genetic algorithm is applied. For comparison, other three methods are implemented. They are image retrieval based on color feature, texture feature and fusion of color-texture feature similarity score with equal weights. The experimental results demonstrate the image retrieval performance of the proposed method is superior to other methods.

Keywords – Image Retrieval, CBIR, Multi-Feature

I. INTRODUCTION

Image retrieval techniques are useful in many image-processing applications. Content-based image retrieval systems work with whole images and searching is based on comparison of the query. General techniques for image retrieval are color, texture and shape. These techniques are applied to get an image from the image database. They are not concerned with the various resolutions of the images, size and spatial color distribution. Hence all these methods are not appropriate to the art image retrieval. Moreover shape based retrievals are useful only in the limited domain. The content and metadata based system gives images using an effective image retrieval technique. Many Image retrieval techniques are useful in many image-processing applications. Content-based image retrieval systems work with whole images and searching is based on comparison of the query. General techniques for image retrieval are color, texture and shape. These techniques are applied to get an image from the image database. They are not concerned with the various resolutions of the images, size and spatial color distribution. Hence all these methods are not appropriate to the art image retrieval. Moreover shape based retrievals are useful only in the limited domain. The content and metadata other image retrieval systems use global features like color, shape and texture. But the prior results say there are too many false positives while using those global features to search for similar images. Hence we give the new view of image retrieval system using both content and metadata.

1.1 BACKGROUND

1.1.1 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 startled 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 computerized creation, manipulation and storage of images, though the high cost of hardware limited their use until the mid-1980s.

Once computerized imaging became affordable (thanks largely to the development of a mass market for computer games), 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 [Sclaroff et al, 1997] – a figure in which some observers consider to be a significant underestimate.

1.1.2 THE NEED FOR IMAGE DATA MANAGEMENT

The process of digitization does not in itself make image collections easier to manage. Some form of cataloguing and indexing is still necessary – the only difference being that much of the required information can now potentially be derived automatically from the images themselves. The extent to which this potential is currently being realized is discussed below. The need for efficient storage and retrieval of images – recognized by managers of large image collections such as picture libraries and design archives for many years – was reinforced by a workshop sponsored by the USA's National Science Foundation in 1992 [Jain, 1993]. After examining the issues involved in managing visual information in some depth, the participants concluded that images were indeed likely to play an increasingly important role in electronically-mediated communication. However, significant research advances, involving collaboration between a numbers of disciplines, would be needed before image providers could take full advantage of the opportunities offered. They identified a number of critical areas where research was needed, including data representation, feature extractions and indexing, image query matching and user interfacing. One of the main problems they highlighted was the difficulty of locating a desired image in a large and varied collection. While it is perfectly feasible to identify a desired image from a small collection simply by browsing, more effective techniques are needed with collections containing thousands of items. Journalists requesting photographs of a particular type of event, designers looking for materials with a particular color or texture, and engineers looking for drawings of a particular type of part, all need some form of access by image content. The existence and continuing use of detailed classification schemes such as ICONCLASS [Gordon, 1990] for art images, and the Opitz code [Opitz et al, 1969] for machined parts, reinforces this message.

1.2 PROBLEM STATEMENT

The goals for this thesis have been the following.

The primary goal our project is to reduce the computation time and user interaction. The conventional Content Based Image Retrieval (CBIR) systems also display the large amount of results at the end of the process this will drove the user to spend more time to analyze the output images. In our proposed system we compute texture feature and color feature for compute the similarity between query and database images. This integrated approach will reduce the output results to a certain levels based on the user threshold value. The secondary goal is to reduce semantic gap between high level concepts and low level features. Generally the content based image retrieval systems compute the similarity between the query image and the database images. Hence there might be chances for unexpected results at the end the retrieval process. The novel clustering technique cluster the output images and select one representative image from each clusters. A third goal is to evaluate their performance with regard to speed and accuracy. These properties were chosen because they have the greatest impact on the implementation effort. A final goal has been to design and implement an algorithm. This should be done in high-level language or Matlab. The source code should be easy to understand so that it can serve as a reference on the standard for designers that need to implement real-time motion detection.

LITERATURE SURVEY

Several reviews of the literature on image retrieval have been published, from a variety of different viewpoints. Enser [1995] reviews methods for providing subject access to pictorial data, developing a four-category framework to classify different approaches. He discusses the strengths and limitations both of conventional methods based on linguistic cues for both indexing and search, and experimental systems using visual cues for one or both of these. His conclusions are that, while there are serious limitations in current text-based techniques for subject access to image data, significant research advances will be needed before visually-based methods are adequate for this task. He also notes, as does Cawkell [1993] in an earlier study, that more dialogue between researchers into image analysis and information retrieval is needed. Aigrain et al [1996] discuss the main principles of automatic image similarity matching for database retrieval, emphasizing the difficulty of expressing this in terms of automatically generated features. They review a selection of current techniques for both still image retrieval and video data management, including video parsing, shot detection, key frame extraction and video skimming. They conclude that the field is expanding rapidly, but that many major research challenges remain, including the difficulty of expressing semantic information in terms of primitive image features, and the need for significantly improved user interfaces. CBIR techniques are likely to be of most use in restricted subject domains, and where synergies with other types of data (particularly text and speech) can be exploited. Eakins [1996] proposes a framework for image retrieval, classifying image queries into a series of levels, and discussing the extent to which advances in technology are likely to meet users' needs at each level. His conclusion is that automatic CBIR techniques can already address many of users' requirements at level 1, and will be capable of making a significant contribution at level 2 if current research ideas can be successfully exploited. They are however most unlikely to make any impact at level 3 in the foreseeable future. Idris and Panchanathan [1997a] provide an in-depth review of CBIR technology, explaining the principles behind techniques for color, texture, shape and spatial indexing and retrieval in some detail. They also discuss the issues involved in video segmentation, motion detection and retrieval techniques for compressed images. They identify a number of key unanswered research questions, including the development of more robust and compact image content features, more accurate modelling of human perceptions of image similarity, the identification of more efficient physical storage and indexing techniques, and the development of methods of recognizing objects within images. De Marsicoi et al [1997] also review current CBIR technology, providing a useful feature-by-feature comparison of 20 experimental and commercial systems.

As a key issue in CBIR, similarity measure quantifies the resemblance in contents between a pair of images [28]. Depending on the type of features, the formulation of the similarity measure varies greatly. The Mahalanobis distance [12] and intersection distance [35] are commonly used to compute the difference between two histograms with the same number of bins. When the number of bins are different, the Earth mover's distance (EMD) [26] applies. The EMD is computed by solving a linear programming problem. Moments [18], the Hausdorff metric [14], elastic matching [2], and decision trees [16] have been proposed for shape comparison. In addition to these reviews of the literature, a survey of "non-text information retrieval" was carried out in 1995 on behalf of the European Commission by staff from GMD (Gesellschaft für Mathematik und Datenverarbeitung), Darmstadt and Université Joseph Fourier de Grenoble [Berrut et al, 1995]. This reviewed current indexing practice in a number of European image, video and sound archives, surveyed the current research literature, and assessed the likely future impact of recent research and development on electronic publishing. The survey found that all current operational image archives used text-based indexing methods, which were perceived to have a number of shortcomings. In particular, indexing vocabularies were not felt to be adequate for non-text material. Despite this, users seemed generally satisfied with existing systems. The report concluded that standard information retrieval techniques were appropriate for managing collections of non-text data, though the adoption of intelligent text retrieval techniques such as the inference-based methods developed in the INQUERY project [Turtle and Croft, 1991] could be beneficial.

II. HEADINGS

III. CONTENT BASED IMAGE RETRIEVAL

2.1 CONTENT-BASED IMAGE RETRIEVAL (CBIR)

In contrast to the text-based approach of the systems described in section above, CBIR operates on a totally different principle, retrieving stored images from a collection by comparing features automatically extracted from the images themselves. The commonest features used are mathematical measures of color, texture or shape; hence virtually all-current CBIR systems, whether commercial or experimental, operate at level 1. A typical system that allows users to formulate queries by submitting an example of the type of image being sought, though some offer alternatives such as selection from a palette or sketch input. The system then identifies those stored images whose feature values match those of the query most closely, and displays thumbnails of these images on the screen. Some of the more commonly used types of feature used for image retrieval are described below.

1) *2.1.1 Color Feature Based Retrieval*

Several methods for retrieving images on the basis of color similarity have been described in the literature, but most are variations on the same basic idea. Each image added to the collection is analyzed to compute a *color histogram*, which shows the proportion of pixels of each color within the image. The color histogram for each image is then stored in the database. At search time, the user can either specify the desired proportion of each color (75% olive green and 25% red, for example), or submit an example image from which a color histogram is calculated. Either way, the matching process then retrieves those images whose color histograms match those of the query most closely. The matching technique most commonly used, histogram intersection, was first developed by Swain and Ballard [1991]. Variants of this technique are now used in a high proportion of current CBIR systems. Methods of improving on Swain and Ballard's original technique include the use of cumulative color histograms [Stricker and Orengo, 1995], combining histogram intersection with some element of spatial matching [Stricker and Dimai, 1996], and the use of region-based color querying [Carson et al, 1997]. The results from some of these systems can look quite impressive.

a) RGB color model

The RGB color model is composed of the primary colors Red, Green, and Blue. This system defines the color model that is used in most color CRT monitors and color raster graphics. They are considered the "additive primaries" since the colors are added together to produce the desired color. The RGB model uses the cartesian coordinate system as shown in Figure 1. (a). Notice the diagonal from (0,0,0) black to (1,1,1) white which represents the grey-scale. Figure 1. (b) is a view of the RGB color model looking down from "White" to origin.

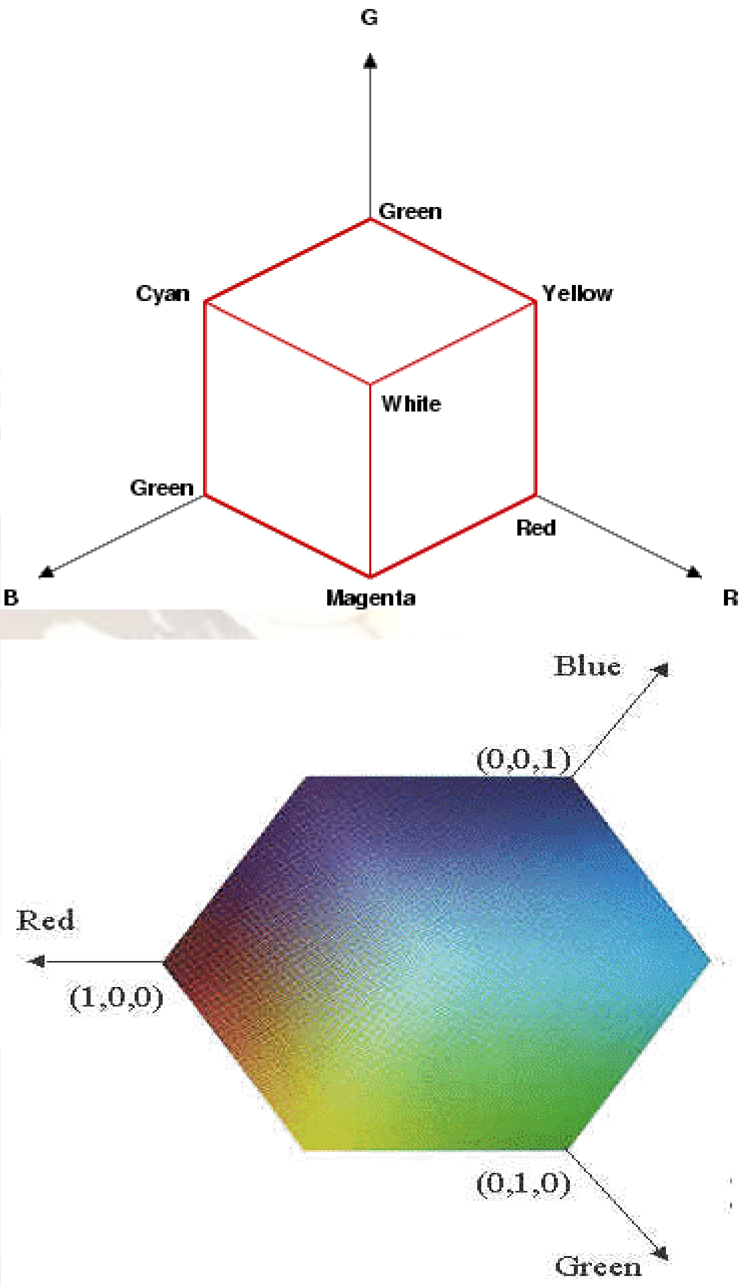


Figure1. (a) RGB coordinates system (b) RGB color model

b) HSV Color Model: The HSV stands for the Hue, Saturation, and Value based on the artists (Tint, Shade, and Tone). The coordinate system in a hexacone in Figure 2. (a). And Figure 2.(b) a view of the HSV color model. The Value represents intensity of a color, which is decoupled from the color information in the represented image. The hue and saturation components are intimately related to the way human eye perceives color resulting in image processing algorithms with physiological basis. As hue varies from 0 to 1.0, the corresponding colors vary from red, through yellow, green, cyan, blue, and magenta, back to red, so that there are actually red values both at 0 and 1.0. As saturation varies from 0 to 1.0, the corresponding colors (hues) vary from unsaturated (shades of gray) to fully saturated (no white component). As value, or brightness, varies from 0 to 1.0, the corresponding colors become increasingly brighter.

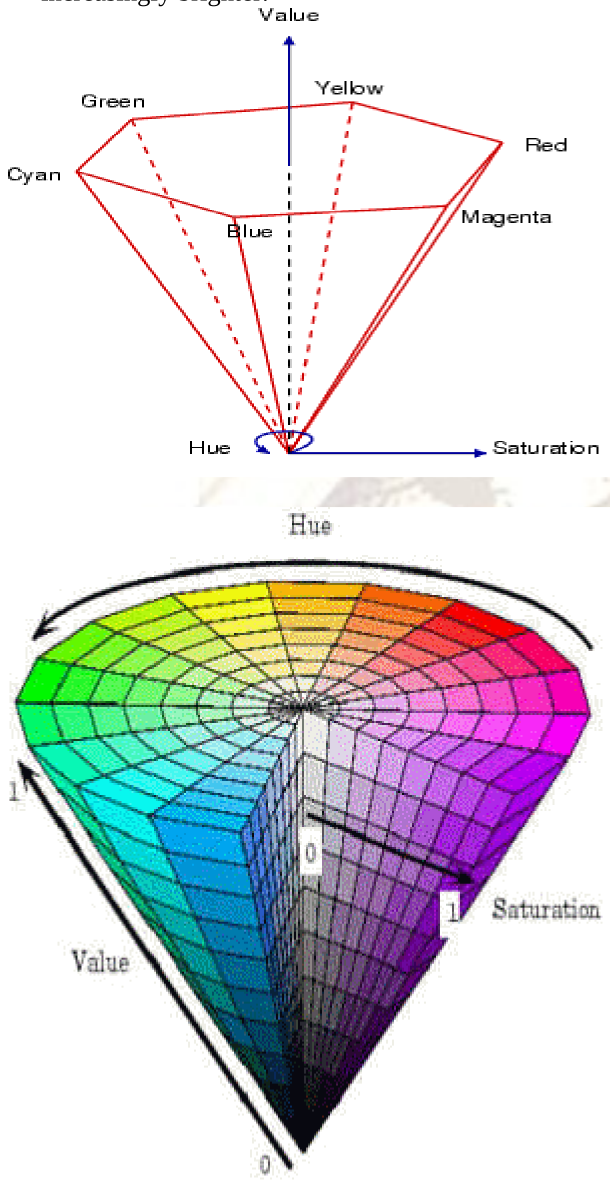


Figure 2. (a) HSV coordinates system (b) HSV color model

Color conversion

In order to use a good color space for a specific application, color conversion is needed between color spaces. The good color space for image retrieval system should preserve the perceived color differences. In other words, the numerical Euclidean difference should approximate the human perceived difference.

RGB to HSV conversion

In Figure 3., the obtainable HSV colors lie within a triangle whose vertices are defined by the three primary colors in RGB space:

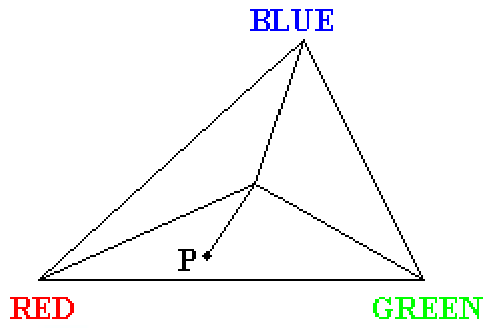


Figure 3. Obtainable HSV color from RGB color space

The hue of the point **P** is the measured angle between the line connecting **P** to the triangle center and line connecting RED point to the triangle center. The saturation of the point **P** is the distance between **P** and triangle center.

The value (intensity) of the point **P** is represented as height on a line perpendicular to the triangle and passing through its center. The grayscale points are situated onto the same line. And the conversion formula is as follows :

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

$$, S = 1 - \frac{3}{R+G+B}[\min(R,G,B)] ,$$

$$V = \frac{1}{3}(R+G+B) \qquad (1)$$

HSV to RGB conversion

Conversion from HSV space to RGB space is more complex. And, given to the nature of the hue information, we will have a different formula for each sector of the color triangle.

Red-Green Sector

for $0° < H \leq 120°$

$$b = \frac{1}{3}(1-S), \quad r = \frac{1}{3}\left[1 + \frac{S\cos H}{\cos(60° - H)}\right], \quad g = 1-(r+b)$$

Green-Blue Sector

for $120° < H \leq 240°$

$$r = \frac{1}{3}(1-S), \quad g = \frac{1}{3}\left[1 + \frac{S\cos H}{\cos(60° - H)}\right], \quad b = 1-(r+b)$$

Blue-Red Sector

$for \quad 240° < H \leq 360°$

$$g = \frac{1}{3}(1-S),\quad b = \frac{1}{3}\left[1 + \frac{S\cos H}{\cos(60° - H)}\right],\quad r = 1-(r+b) \quad \ldots (2)$$

2.1.2 Histogram-Based Image Search

The color histogram for an image is constructed by counting the number of pixels of each color. Retrieval from image databases using color histograms has been investigated in [tools, fully, automated]. In these studies the developments of the extraction algorithms follow a similar progression: (1) selection of a color space, (2) quantization of the color space, (3) computation of histograms, (4) derivation of the histogram distance function, (5) identification of indexing shortcuts. Each of these steps may be crucial towards developing a successful algorithm.

There are several difficulties with histogram based retrieval. The first of these is the high dimensionality of the color histograms. Even with drastic quantization of the color space, the image histogram feature spaces can occupy over 100 dimensions in real valued space. This high dimensionality ensures that methods of feature reduction, pre-filtering and hierarchical indexing must be implemented. The large dimensionality also increases the complexity and computation of the distance function. It particularly complicates `cross' distance functions that include the perceptual distance between histogram bins [2].

2.1.3 Color histogram definition

An image histogram refers to the probability mass function of the image intensities. This is extended for color images to capture the joint probabilities of the intensities of the three color channels. More formally, the color histogram is defined by,

$$h_{A,B,C}(a,b,c) = N \cdot \Pr ob(A = a, B = b, C = c) \quad \ldots (3)$$

where A , B and C represent the three color channels (R,G,B or H,S,V) and N is the number of pixels in the image. Computationally, the color histogram is formed by discretizing the colors within an image and counting the number of pixels of each color. Since the typical computer represents color images with up to 224 colors, this process generally requires substantial quantization of the color space. The main issues regarding the use of color histograms for indexing involve the choice of color space and quantization of the color space. When a perceptually uniform color space is chosen uniform quantization may be appropriate. If a non-uniform color space is chosen, then non-uniform quantization may be needed. Often practical considerations, such as to be compatible with the workstation display, encourage the selections of uniform quantization and RGB color space. The color histogram can be thought of as a set of vectors. For gray-scale images these are two dimensional vectors. One dimension gives the value of the gray-level and the other the count of pixels at the gray-level. For color images the color histograms are composed of 4-D vectors. This makes color histograms very difficult to visualize. There are several lossy approaches for viewing color histograms, one of the easiest is to view separately the histograms of the color channels. This type of visualization does illustrate some of the salient features of the color histogram [2].

2.1.4 Color uniformity

The RGB color space is far from being perceptually uniform. To obtain a good color representation of the image by uniformly sampling the RGB space it is necessary to select the quantization step sizes to be fine enough such that distinct colors are not assigned to the same bin. The drawback is that oversampling at the same time produces a larger set of colors than may be needed. The increase in the number of bins in the histogram impacts performance of database retrieval. Large sized histograms become computationally unwieldy, especially when distance functions are computed for many items in the database. Furthermore, as we shall see in the next section, to have finer but not perceptually uniform sampling of colors negatively impacts retrieval effectiveness.

However, the HSV color space mentioned earlier offers improved perceptual uniformity. It represents with equal emphasis the three color variants that characterize color: Hue, Saturation and Value (Intensity). This separation is attractive because color image processing performed independently on the color channels does not introduce false colors. Furthermore, it is easier to compensate for many artifacts and color distortions. For example, lighting and shading artifacts are typically be isolated to the lightness channel. But this color space is often inconvenient due to the non-linearity in forward and reverse transformation with RGB space [2].

2.1.5 Color Histogram Discrimination

There are several distance formulas for measuring the similarity of color histograms. In general, the techniques for comparing probability distributions, such as the kolmogoroff-smirnov test are not appropriate for color histograms. This is because visual perception determines similarity rather than closeness of the probability distributions. Essentially, the color distance formulas arrive at a measure of similarity between images based on the perception of color content. Three distance formulas that have been used for image retrieval including histogram euclidean distance, histogram intersection and histogram quadratic (cross) distance [2, 3].

2.1.6 Histogram Quadratic Distance

Let ***h*** and ***g*** represent two color histograms. The euclidean distance between the color histograms ***h*** and ***g*** can be computed as:

$$d^2(h,g) = \sum_A \sum_B \sum_C (h(a,b,c) - g(a,b,c))^2 \quad \ldots (4)$$

In this distance formula, there is only comparison between the identical bins in the respective histograms. Two different bins may represent perceptually similar colors but are not compared cross-wise. All bins contribute equally to the distance.

2.1.7 Histogram intersection distance

The color histogram intersection was proposed for color image retrieval in [4]. The intersection of histograms ***h*** and ***g*** is given by:

$$d(h,g) = \frac{\sum_A \sum_B \sum_C \min(h(a,b,c), g(a,b,c))}{\min(|h|,|g|)} \quad (5)$$

where |***h***| and |***g***| gives the magnitude of each histogram, which is equal to the number of samples. Colors not present in the user's query image do not contribute to the intersection distance. This reduces the contribution of background colors. The sum is normalized by the histogram with fewest samples. The color histogram quadratic distance was used by the QBIC system introduced in [1, 6]. The cross distance formula is given by:

$$d(h,g) = (h-g)^t A (h-g) \quad \ldots (6)$$

The cross distance formula considers the cross-correlation between histogram bins based on the perceptual similarity of the colors represented by the bins. And the set of all cross-correlation values are represented by a matrix ***A***, which is called a similarity matrix. And a (*i*,*j*)th element in the similarity matrix ***A*** is given by : for RGB space,

$$a_{ij} = 1 - d_{ij} / \max(d_{ij}) \quad \ldots (7)$$

where d_{ij} is the L_2 distance between the color *i* and *j* in the RGB space. In the case that quantization of the color space is not perceptually uniform the cross term contributes to the perceptual distance between color bins.

For HSV space it is given in [5] by:

$$a_{ij} = 1 - \frac{1}{\sqrt{5}} \left[(v_i - v_j)^2 + (s_i \cos h_i - s_j \cos h_j)^2 + (s_i \cos h_i - s_j \cos h_j)^2 \right]^{\frac{1}{2}} \quad (8)$$

Which corresponds to the proximity in the HSV color space.

2) *2.3 Texture Based Image Retrieval*

The ability to retrieve images on the basis of texture similarity may not seem very useful. But the ability to match on texture similarity can often be useful in distinguishing between areas of images with similar color (such as sky and sea, or leaves and grass). A variety of techniques have been used for measuring texture similarity; the best established rely on comparing values of what are known as *second-order statistics* calculated from query and stored images. Essentially, these calculate the relative brightness of selected *pairs* of pixels from each image. From these it is possible to calculate measures of image texture such as the degree of *contrast*, *coarseness*, *directionality* and *regularity* [Tamura et al, 1978], or *periodicity*, *directionality* and *randomness* [Liu and Picard, 1996]. Alternative methods of texture analysis for retrieval include the use of Gabor filters [Manjunath and Ma, 1996] and fractals [Kaplan et al, 1998]. Texture queries can be formulated in a similar manner to color queries, by selecting examples of desired textures from a palette, or by supplying an example query image. The system then retrieves images with texture measures most similar in value to the query. A recent extension of the technique is the texture thesaurus developed by Ma and Manjunath [1998], which retrieves textured regions in images on the basis of similarity to automatically derived code words representing important classes of texture within the collection. Texture is another important property of images. Various texture representations have been investigated in pattern recognition and computer vision. Basically, texture representation methods can be classified into two categories:

1) Structural and statistical

Structural methods, including *morphological operator* and *adjacency graph*, describe texture by identifying structural primitives and their placement rules. They tend to be most effective when applied to textures that are very regular. Statistical methods, including *Fourier power spectra*, *co-occurrence matrices*, *shift-invariant principal component analysis* (*SPCA*), *Tamura feature*, *Wold decomposition*, *Markov random field*, *fractal model*, and *multi-resolution filtering* techniques such as *Gabor and wavelet transform*, characterize texture by the statistical distribution of the image intensity.

2) Wavelet Transform Features

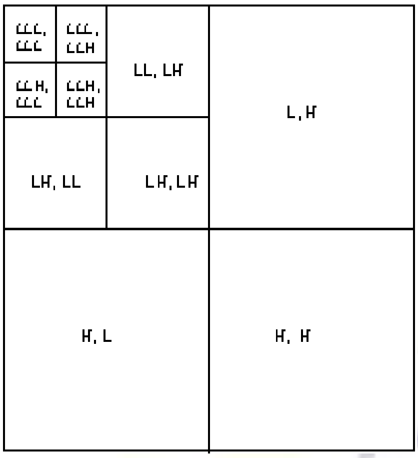


Fig.1: Pyramid Wavelet Transform (Level 3)

The standard Pyramid Wavelet Transform is shown in the Figure 1. The first step is to resize the image size into 256X256 in a matrix format. Then the pyramid wavelet transform is applied to get the sub bands of the image. To find the energy measures of the image Daubechies filter is applied. The decomposition is applied to 6 levels so that we can able to get the low frequency contents in the LL sub band and other frequencies in LH, HL and HH bands separately. Finally we will get the 4X4-sized image. Once the wavelet coefficients of an image are available, features are computed from each sub-band, resulting in 19 features for each image. The mean μ is the energy measure used to compute the features, then the feature vector f, for a particular image is calculated using the given formula [3].

$f = [\mu_{mn}]$, $n \neq 1$ except for the coarsest level, $m=6$

$f = [\mu_{1,2}, \mu_{1,3}, \mu_{1,4}, \mu_{2,2}, \mu_{2,3} \ \ldots \ \mu_{6,1}, \mu_{6,2}, \mu_{6,3}, \mu_{6,4}]$

Where μ_{mn} is the energy measure for the decomposition level and the sub bands. Now we get the energy coefficients and stored in the database. When the user gives the query image then it will be converted into the same above operations and finally gives the energy measure coefficients. The distance between the two images is calculated using Euclidean distance classifier [1]. Thus the similar kind of images can be retrieved using the above-described method and they are optimized using K-nearest neighbor algorithm [1]. This paper is analyzed using these specified methods and the results were shown. Instead of exact matching, content-based image retrieval calculates visual similarities between a query image and images in a database. Accordingly, the retrieval result is not a single image but a list of images ranked by their similarities with the query image. Many similarity measures have been developed for image retrieval based on empirical estimates of the distribution of features in recent years. Different *similarity/distance measures* will affect retrieval performances of an image retrieval system significantly. In this section, we will introduce some commonly used similarity measures. We denote $D(I, J)$ as the distance measure between the query image I and the image J in the database; and $fi(I)$ as the number of pixels in bin i of I.

Quadratic Form (QF) Distance:

The Minkowski distance treats all bins of the feature histogram entirely independently and does not account for the fact that certain pairs of bins correspond to features which are perceptually more similar than other pairs. To solve this problem, *quadratic form distance is introduced.*

$$D(I,J) = \sqrt{(\mathbf{F}_I - \mathbf{F}_J)^T \mathbf{A}(\mathbf{F}_I - \mathbf{F}_J)}$$

where $A=[aij]$ is a similarity matrix, and aij denotes the similarity between bin i and j.

I **F** and J **F** are vectors that list all the entries in $fi(I)$ and $fi(J)$. Quadratic form distance has been used in many retrieval systems [40, 67] for color histogram-based image retrieval. It has been shown that quadratic form distance can lead to perceptually more desirable results than Euclidean distance and histogram intersection method as it considers the cross similarity between colors.

3) *Development environment*

2.4 SYSTEM REQUIREMENTS

The following are the required components and environment for project development.

2.3.1 Software Requirements

- Simulation Software – Mat Lab 6.5 or above

2.3.2 Hardware Requirements

- 256 MB RAM
- Processor speed 2.33 GHz. CPU
- Visual Display Unit

Operating System: Windows 2000 Professional / XP

The shunt active compensator eliminates the supply current harmonics and improves supply power-factor for both nonlinear and linear loads with different load characteristics.

It can also be concluded that, by using Indirect current control technique the maximum switching frequency of the IGBT's of the Active filter is reduced, because this technique controls the compensating currents of AF, by sensing the source current, which are slow varying (sinusoidal) in nature compared to the harmonic components, which are very fast varying.

In this work a Hysteresis current controller and amp-comparator current controller are simulated in Simulink. It is observed that both the controllers have satisfactory performance. With both the controllers, the supply power factor is improved to unity, with the harmonics in the supply current being eliminated for different types of loading conditions. As the switching frequency is fixed in the case ramp comparator current control method it has better performance when compared with hysteresis current control method.

IV. MODULES OF THE PROPOSED SYSTEM

4.1 GENERAL DESCRIPTION

The project will be developed in three different modules such that the output can be viewed in three different manners

1. Color Feature Extraction and Retrieval
2. Texture Feature Extraction and Retrieval
3. Integration and sorting

4.2 COLOR FEATURE EXTRACTIONS AND RETRIEVAL: The color feature based image retrieval includes the following steps

1. Feature Extraction
2. Histogram Computation
3. Similarity Matrix Computation
4. Dissimilarity Computation
5. Sorting Images in ascending

3.2.1 Module Block Diagram

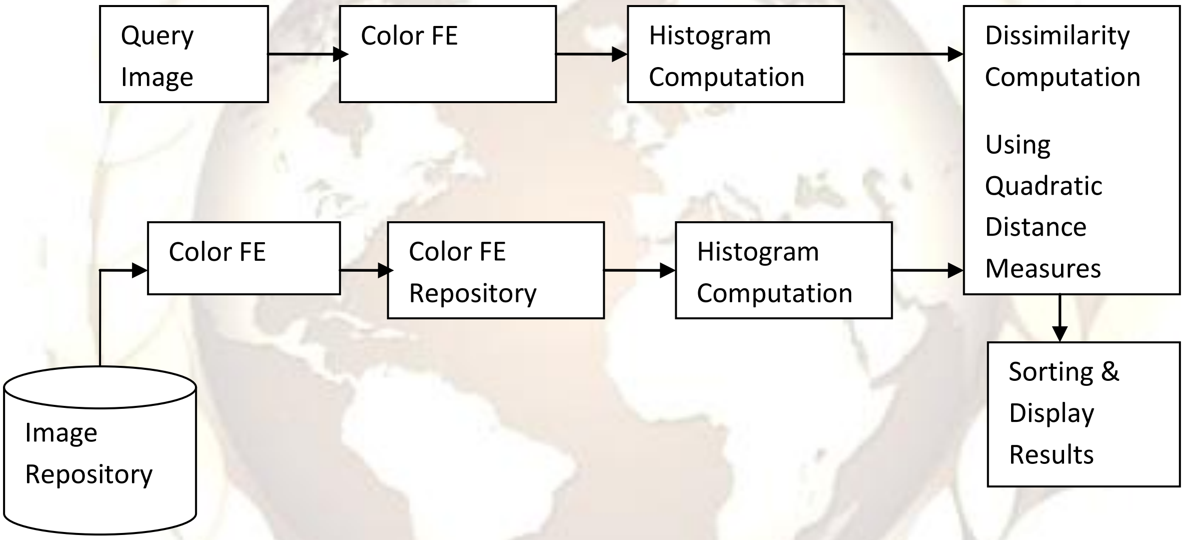


Fig. Color Feature Extraction Block Diagram

3.2.2 Feature Extraction: The color features vector of query and database images are computed in HSV color space. The histogram analysis of an image better in Hue Saturation Value(HSV) space than RGB. Hence our proposed method includes HSV mapping to obtain the color map histogram.

3.2.3 Histogram Quadratic (cross) Distance: There are several distance measure are available for similarity comparison namely, Euclidean distance measure, Quadratic distance measure, Normalized Euclidean distance measure. Among these measures Quadratic distance measures provides good retrieval accuracy.

3.2.4. Dissimilarity Computation: Let ***h*** and ***g*** represent two color histograms. The Euclidean distance between the color histograms ***h*** and ***g*** can be computed as:

$$d^2(h,g) = \sum_A \sum_B \sum_C (h(a,b,c) - g(a,b,c))^2 \qquad (4)$$

In this distance formula, there is only comparison between the identical bins in the respective histograms. Two different bins may represent perceptually similar colors but are not compared cross-wise. All bins contribute equally to the distance.

The histogram quadratic distance was proposed by QBIC for color indexing [3]. It measures the weighted similarity between histograms, which provides more desirable results than ``like-bin" only comparisons. The quadratic distance between histograms h and g is given by:

$$d(h,g) = (h-g)^t A\,(h-g)$$

The cross distance formula considers the cross-correlation between histogram bins based on the perceptual similarity of the colors represented by the bins. And the set of all cross-correlation values are represented by a matrix ***A***, which is called a similarity matrix. And a (*i*,*j*)th element in the similarity matrix ***A*** is given by : for RGB space,

$$a_{ij} = 1 - d_{ij} / \max(d_{ij})$$

where d_{ij} is the L_2 distance between the color *i* and *j* in the RGB space. In the case that quantization of the color space is not perceptually uniform the cross term contributes to the perceptual distance between color bins.

For HSV space it is given in [5] by:

$$a_{ij} = 1 - \frac{1}{\sqrt{5}}[(v_i - v_j)^2 + (s_i \cos h_i - s_j \cos h_j)^2 + (s_i \cos h_i - s_j \cos h_j)^2]^{\frac{1}{2}}$$

which corresponds to the proximity in the HSV color space

3.2.5. Sorting Images in ascending

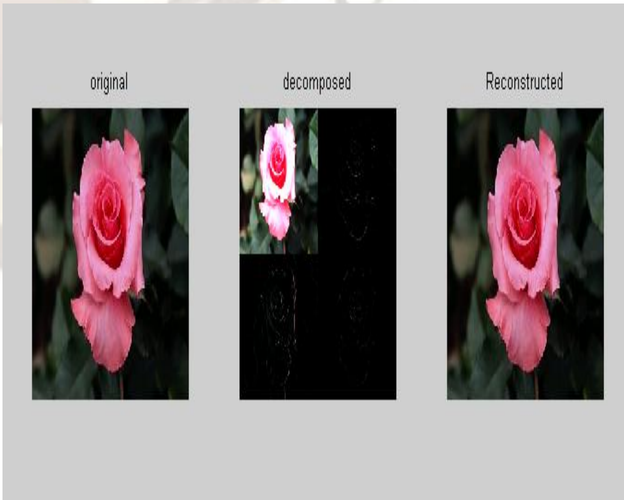
The relevant images are sorted out based on the dissimilarity and displayed in ascending manner.

V. RESULTS

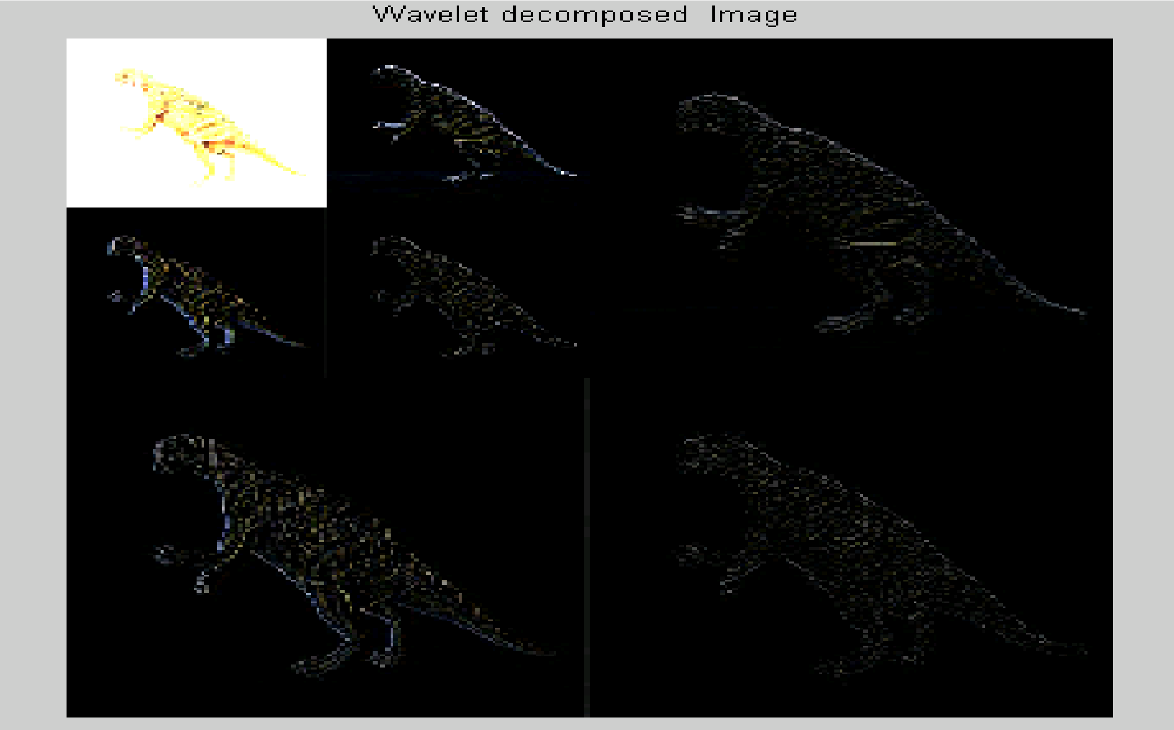
We use two different algorithms: content and texture based. The proximity between two images is calculated using two different techniques too: ED between color histograms and ED between wavelet energies. In order to show advantages and disadvantages of these techniques we choose five pairs of similar images, see Figure 5.1.We have run our application on a database of 150 images, with of these ten images as source image. The parameter for comparison of different methods is average proximity to its pair image, as function of number of quantization colors, where proximity is the relative position of the target image in the order determined by content based.

DATABASE

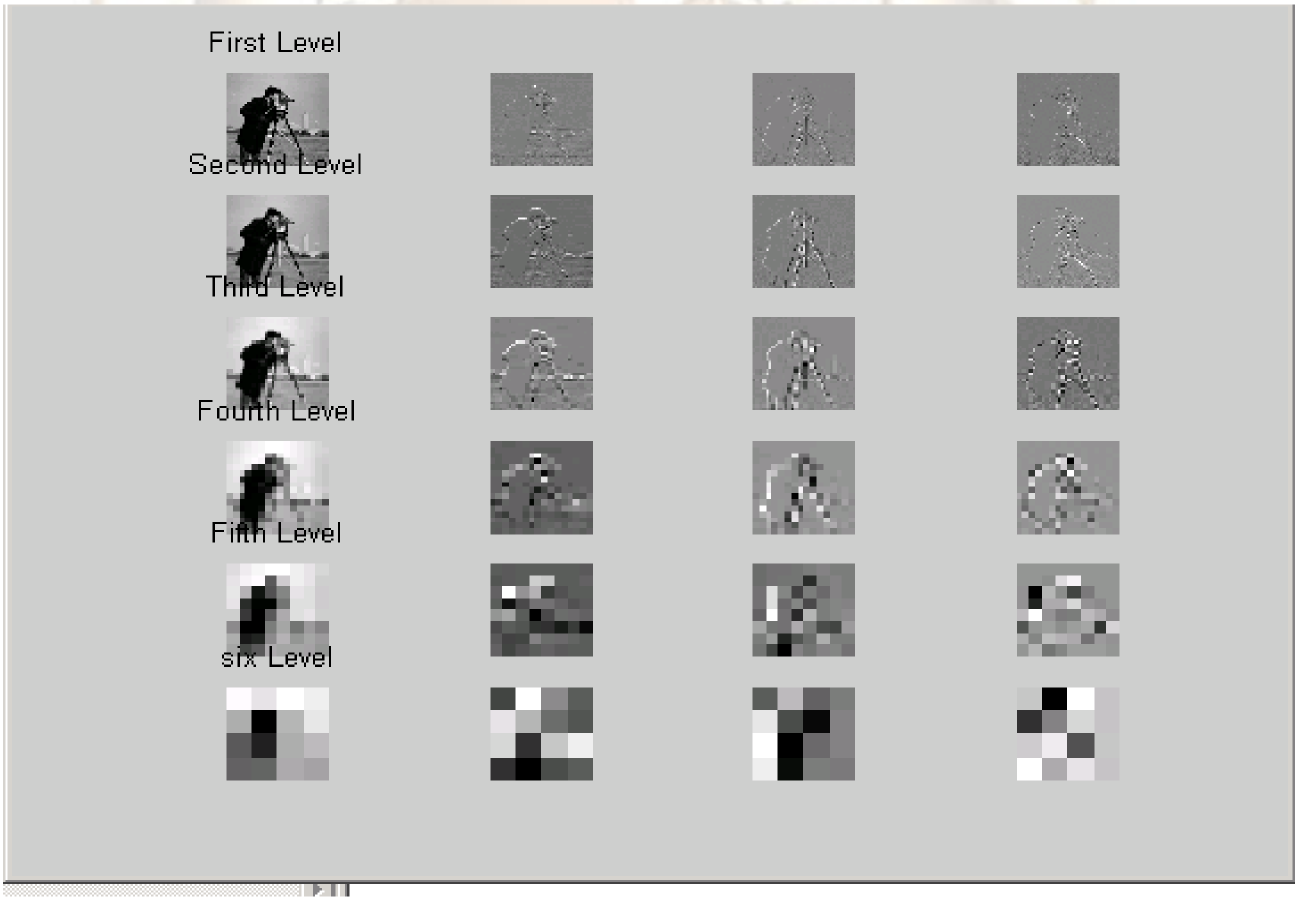
WAVELET DECOMPOSITION



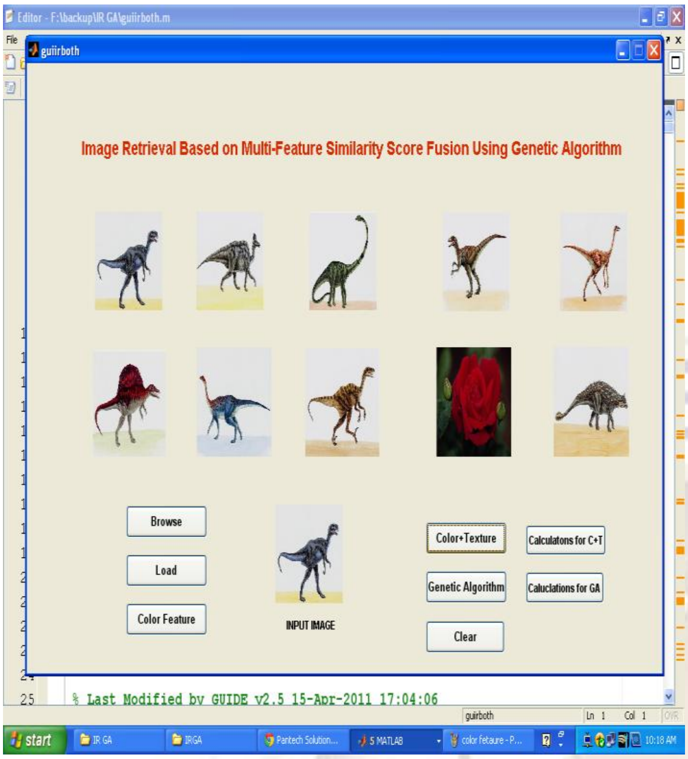
SECOND LEVEL DECOMPOSITION



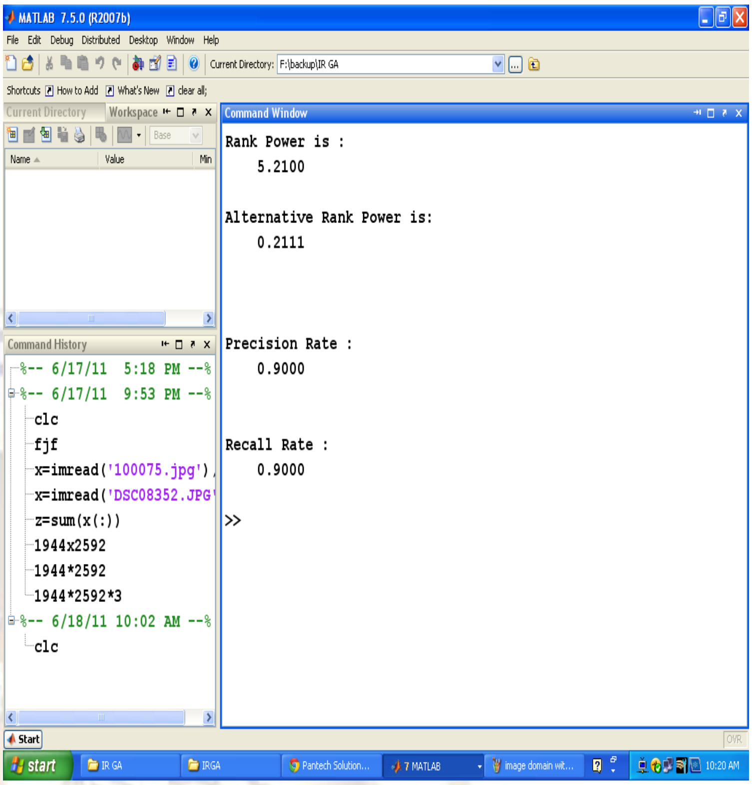
SIX LEVEL DECOMPOSITION



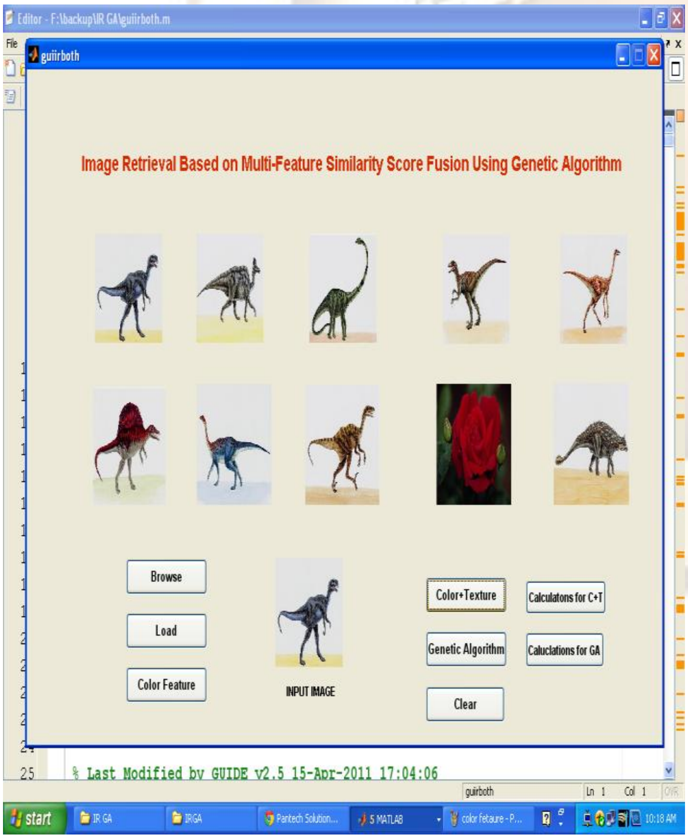
COLOR FEATURE EXTRACTION



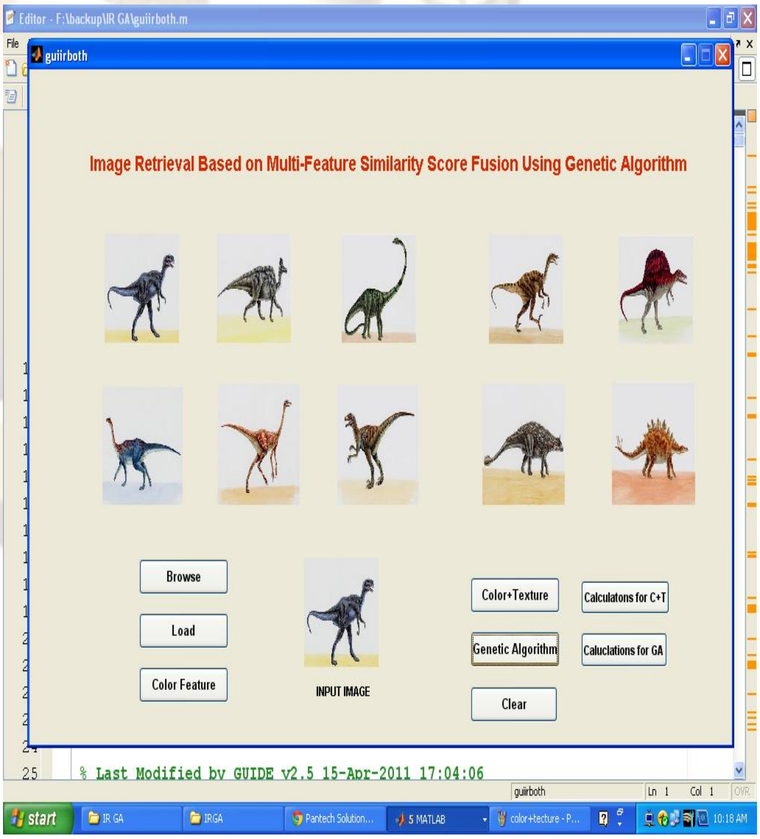
CALCULATIONS FOR COLOR+TEXTURE FEATURE



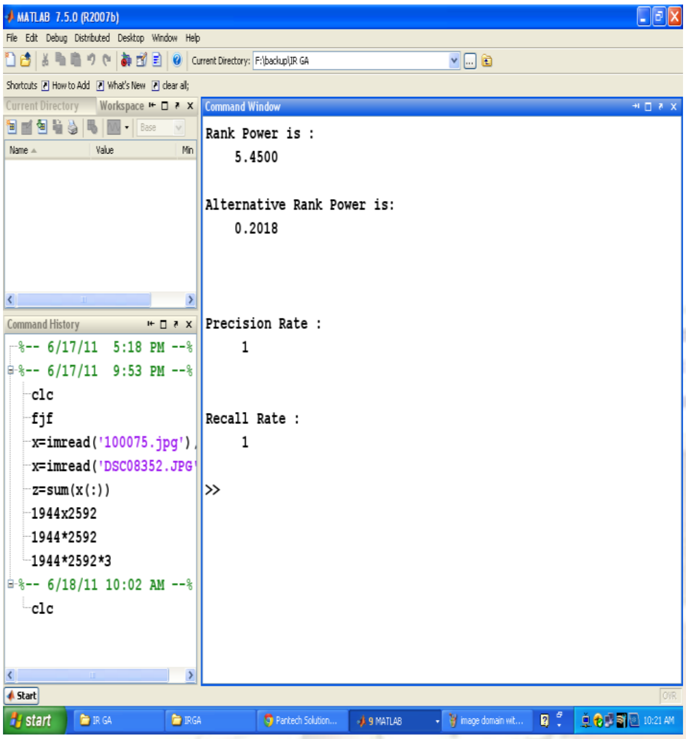
COLOR+TEXTURE FEATURE EXTRACTION



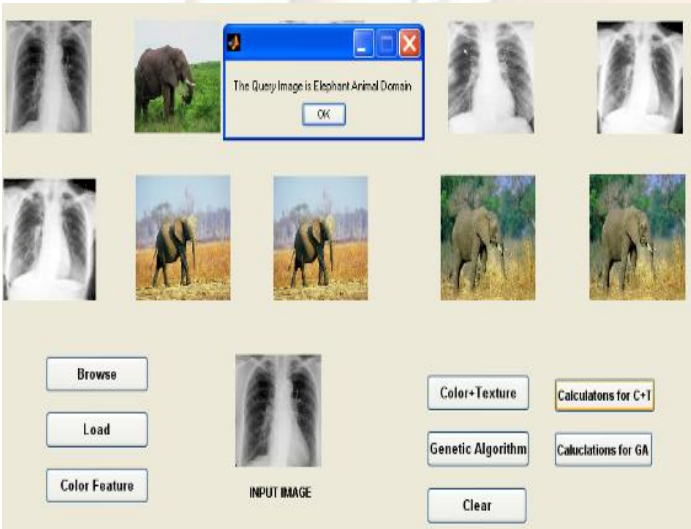
APPLYING GENETIC ALGORITHM



CALCULATIONS FOR COLOR+TEXTURE FEATURE



COLOUR + TEXTURE EXTRACTION



CALCULATIONS FOR COLOR+TEXTURE FEATURE

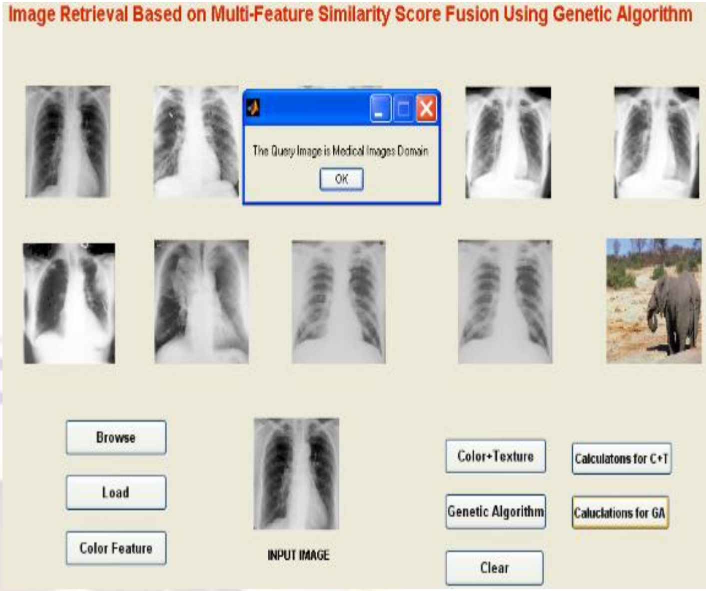
```
Rank Power is :
    2.1717

Alternative Rank Power is:
    0.4847

Precision Rate :
    0.5000

Recall Rate :
    0.2632
```

APPLYING GENETIC ALGORITHM



CALCULATIONS FOR COLOR+TEXTURE FEATURE

```
Optimisation terminated: average change in the fitness v
Rank Power is :
    5.9900

Alternative Rank Power is:
    0.1836

Precision Rate :
    0.9000

Recall Rate :
    0.9000

>>
```

VI. CONCLUSION

In this project we will analyze an integrated approach of color and texture features of image content descriptor based system for medical and art gallery image database. This proposed algorithm will provides an effective approach for query based image retrieval system. The timing results for the integrated approach will be less and accurate, this can be improved by integrating other spatial relationship. In future enhancements we extend our features selections and introduce other distance measures to the user in order to improve the results.

References

[1] Barbeau Jerome, Vignes-Lebbe Regine, and Stamon Georges, "A Signature based on Delaunay Graph and Co-occurrence Matrix," Laboratoire Informatique et Systematique, University of Paris, Paris, France, July 2002,.

[2] Sharmin Siddique, "A Wavelet Based Technique for Analysis and Classification of Texture Images," Carleton University, Ottawa, Canada, Proj. Rep. 70.593, April 2002.
[3] The 23rd International Conference on Very Large Data Bases VLDB'97, Athens, Greece, August 1997
[4] FOLDOC, *Free On-Line Dictionary Of Computing*, "co-occurrence matrix," May 1995, [Online Document].
[5] Colin C. Venteres and Dr. Matthew Cooper, "A Review of Content-Based Image Retrieval Systems", [Online Document.
[6] Linda G. Shapiro, and George C. Stockman, *Computer Vision*, Prentice Hall, 2001.
[7] Shengjiu Wang, "A Robust CBIR Approach Using Local Color Histograms," Department of Computer Science, University of Alberta, Edmonton, Alberta, Canada, Tech. Rep. TR 01-13, October 2001.
[8] Shengjiu Wang, "A Robust CBIR Approach Using Local Color Histograms," Department of Computer Science, University of Alberta, Edmonton, Alberta, Canada, Tech. Rep. TR 01-13, October 2001,.
[9] "Texture," class notes for *Computerized Image Analysis MN2*, Centre for Image Analysis, Uppsala, Sweden, Winter 2002.
[10] G. D. Magoulas, S. A. Karkanis, D. A. Karras and M. N. Vrahatis, "Comparison Study of Textural Descriptors for Training Neural Network Classifiers", in Proceedings of the 3rd IEEE-IMACS World Multi-conference on Circuits, Systems, Communications and Computers, vol. 1, 6221-6226, Athens, Greece, July 1999.
[11] Pravi Techasith, "Image Search Engine," Imperial College, London, UK, Proj. Rep., July 2002.