ISSN No. 0976-5697

Volume 3, No. 3, May-June 2012

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

An Intelligence Image Retrieval System Based On Evolutionary Algorithm

S. Malarvizhi*
[1]Scholar, ME, S.A Engineering College,
Anna University, Chennai, India
malar.skr@gmail.com

N.Magadevi
[2]Assistant professor, S.A Engineering College,
Anna University, Chennai, India
mahadevinirmalkumar@gmail.com

Abstract- Digital image libraries and other multimedia databases have been dramatically expanded in recent years. In order to effectively and precisely retrieve the desired images from a large image database, the development of a content-based image retrieval (CBIR) system has been used. However, most of the proposed approaches emphasize on finding the best representation for different image features. In this paper, a user-interactive mechanism for CBIR method based on Particle Swarm Optimization (PSO) is proposed. Color attributes like the mean value, the standard deviation, and the image bitmap of a color image are used as the features for retrieval. In addition, the entropy based on the gray level co-occurrence matrix and the edge histogram of an image is also considered as the texture features. Furthermore, to reduce the gap between the retrieval results and the users' expectation, the PSO is employed to help the users identify the images that are most satisfied to the users' need .Experimental results and comparisons demonstrate the feasibility of the proposed approach.

Index Terms— Content-based image retrieval (CBIR), Particle Swarm Optimization (PSO)

I. INTRODUCTION

In recent years, rapid advances in science and technology have produced a large amount of image data in diverse areas, such as entertainment, art galleries, fashion design, education, medicine, industry, etc. We often need to efficiently store and retrieve image data to perform assigned tasks and to make a decision. Therefore, developing proper tools for the retrieval image from large image collections is challenging. Two different types of approaches, i.e., text- and content based, are usually adopted in image retrieval. In the text-based system, the images are manually annotated by text descriptors and then used by a database management system to perform image retrieval. However, there are two limitations of using keywords to achieve image retrieval: the vast amount of labor required in manual image annotation and the task of describing image content is highly subjective. That is, the perspective of textual descriptions given by an annotator could be different from the perspective of a user. In other words, there are inconsistencies between user textual queries and image annotations or descriptions. To alleviate the inconsistency problem, the image retrieval is carried out according to the image contents. Such strategy is the so-called content-based image retrieval (CBIR).

The primary goal of the CBIR system is to construct meaningful descriptions of physical attributes from images to facilitate efficient and effective retrieval [1], [2].CBIR has become an active and fast-advancing research area in image retrieval in the last decade. By and large, research activities in CBIR have progressed in four major directions: global image properties based, region-level features based, relevance feedback, and semantic based. Initially, developed algorithms exploit the low-level features of the image such as color, texture, and shape of an object to help retrieve images. They are easy to implement and perform well for images that are either simple or contain few semantic contents. However, the semantics of an image are difficult to be revealed by the visual features, and these algorithms have many limitations when dealing with broad content image database. Therefore, in order to improve the retrieval accuracy of CBIR systems, region based image retrieval methods via image segmentation were introduced. These methods attempt to overcome the drawbacks of global features by representing images at object level, which is intended to be close to the perception of human visual system.

However, the performance of these methods mainly relies on the results of segmentation. The difference between the user's information need and the image representation is called the semantic gap in CBIR systems. The limited retrieval accuracy of image centric retrieval systems is essentially due to the inherent semantic gap. In order to reduce the gap, the interactive relevance feedback is introduced into CBIR. The basic idea behind relevance feedback is to incorporate human perception subjectivity into the query process and provide users with the opportunity to evaluate the retrieval results. The similarity measures are automatically refined on the basis of these evaluations. However, although relevance feedback can significantly improve the retrieval performance, its applicability still suffers from a few drawbacks [3].

The semantic-based image retrieval methods try to discover the real semantic meaning of an image and use it to retrieve relevant images. However, understanding and discovering the semantics of a piece of information are high level cognitive tasks and thus hard to automate. A wide variety of CBIR algorithms has been proposed, but most of them focus on the similarity computation phase to efficiently find a specific image or a group of images that are similar to the given query. In order to achieve a better approximation of the user's information need for the following search in the image database, involving user's interaction is necessary for a CBIR system. In this paper, we propose a user-oriented CBIR system that uses the interactive particle swarm optimization(PSO) to infer which images in the databases would be of most interest to the user. Three visual features, color, texture, and edge, of an image are utilized in our approach. PSO provides an interactive mechanism to better capture user's intention.

There are very few CBIR systems considering human's knowledge, but [6] is the representative one. They considered the red, green, and blue (RGB) color model and wavelet coefficients to extract image features. In their system, the query procedure is based on association (e.g., the user browses an image collection to choose the most suitable ones). The main properties of this paper that are different from it can be identified as follows: 1) low-level image features-color features from the hue, saturation, value (HSV) color space, as well as texture and edge descriptors, are adopted in our approach and 2) search technique—our system adopts the query-by-example strategy (i.e., the user provides an image query). In addition, we hybrid the user's subjective evaluation and intrinsic characteristics of the images in the image matching against only considering human judgment [6].The remainder of this paper is organized as follows. Related works about CBIR are briefly reviewed in Section II. Section III describes the considered image features. The proposed approach is presented in Section IV. Section V gives the experimental results and provides comparative performances. Finally, Section VI presents the conclusions of this paper.

II. RELATED WORK

There are some literatures that survey the most important CBIR systems [7], [8]. Also, there are some papers that overview and compare the current techniques in this area [9], [10]. Since the early studies on CBIR, various color descriptors have been adopted. Yoo *et al.* [11] proposed a signature-based color-spatial image retrieval system. Color and its spatial distribution within the image are used for the features. In [12], a CBIR scheme based on the global and local color distributions in an image is presented. Vadivel *et al.* [13] have introduced an integrated approach for capturing spatial variation of both color and intensity levels and shown its usefulness in image retrieval applications.

Texture is also an essential visual feature in defining high level semantics for image retrieval purposes. In [14], a novel, effective, and efficient characterization of wavelet sub bands by bit-plane extractions in texture image retrieval was presented. Inorder to overcome some limitations, such as computational expensive approaches or poor retrieval accuracy, in a few texture based image retrieval methods, Kokare *et al.* [15] concentrated on the problem of finding good texture features for CBIR. They designed 2-D rotated complex wavelet filters to efficiently handle texture images and formulate a new texture-retrieval algorithm using the proposed filters. Pi and Li [16] combined fractal parameters and collage error to propose a set of new statistical fractal signatures. These signatures effectively extract the statistical properties intrinsic in texture images to enhance retrieval rate. Liapis and Tziritas [17] explored image retrieval mechanisms based on a combination of texture and color features. Texture features are extracted using discrete wavelet frame analysis .Two- or one-dimensional histograms of the CIE Lab chromaticity coordinates are used as color features. CBIR method based on an efficient combination of multi resolution color and texture features.

As its color features, color auto correlograms of the hue and saturation component images in HSV color space are used. As its texture features, block difference of inverse probabilities and block variation of local correlation coefficient moments of the value are extracted in multi resolution wavelet domain and then combined. In order to well model the high-level concepts in an image and user's subjectivity, recent approaches introduce human–computer interaction into CBIR. Takagi *et al.* [4] evaluated the performance of the IGA-based image retrieval system that uses wavelet coefficients to represent physical features of images. PSO to solve the problems of fashion design and emotion-based image retrieval. He used wavelet transform to extract image features and PSO to search the image that the user has in mind.

III. IMAGE FEATURES

One of the key issues in querying image databases by similarity is the choice of appropriate image descriptors and corresponding similarity measures. In this section, we first present a brief review of considered low-level visual features in our approach and then review the basic concept of the PSO.

A. *Color Descriptor:*

A color image can be represented using three primaries of a color space. Since the RGB space does not correspond to the human way of perceiving the colors and does not separate the luminance component from the chrominance ones, we used the HSV color space in our approach. HSV is an intuitive color space in the sense that each component contributes directly to visual perception, and it is common for image retrieval systems. Hue is used to distinguish colors, whereas saturation gives a measure of the percentage of white light added to a pure color. Value refers to the perceived light intensity. The important advantages of HSV color space are as follows: good compatibility with human intuition, separability of chromatic and achromatic components, and possibility of preferring one component to other .The color distribution of pixels in an image contains sufficient information. The mean of pixel colors states the principal color of the image, and the standard deviation of pixel colors can depict the variation of pixel colors. The variation degree of pixel colors in an image is called the color complexity of the image. We can use these two features to represent the global properties of an image.

Hence, a feature called binary bitmap can be used to capture the local color information of an image. The basic concept of binary bitmap comes from the block truncation coding [25], which is a relatively simple image coding technique and has been successfully employed in many image processing applications. There are three steps to generate the image binary bitmap. This method first divides an image into several no overlapping blocks.

B. *Texture Descriptor:*

Texture is an important attribute that refers to innate surface properties of an object and their relationship to the surrounding environment. If we could choose appropriate texture descriptors, the performance of the CBIR should be improved. We use a gray level co-occurrence matrix (GLCM), which is a simple and effective method for representing texture [26]. The GLCM represents the probability $p(i, j; d, \theta)$ that two pixels in an image, which are located with distance d and angle θ, have gray levels i and j. The GLCM is mathematically defined as follows:

$$p(i, j; d, \theta) = \#\{(x1, y1)(x2, y2) | g(x1, y1)=i, g(x2, y2)=j, |(x1, y1) - (x2, y2)| = d, \square((x1, y2), (x2, y2)) = \theta\} \quad (7)$$

where # denotes the number of occurrences inside the window,with i and j being the intensity levels of the first pixel and thesecond pixel at positions ($x1, y1$) and ($x2, y2$), respectively.In order to simplify and reduce the computation effort.

C. Edge Descriptor:

Edges in images constitute an important feature to represent their content. 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. We adopt the edge histogram descriptor (EHD) [27] to describe edge distribution with a histogram based on local edge distribution in an image. The extraction process of EHD consists of the following stages.

a) An image is divided into 4 × 4 sub images.
b) Each sub image is further partitioned into no overlapping image blocks with a small size.
c) The edges in each image block are categorized into five
Types: vertical, horizontal, 45◦ diagonal, 135◦ diagonal and no directional edges.
d) Thus, the histogram for each sub image represents the relative frequency of occurrence of the five types of edges in the corresponding sub image.
e) After examining all image blocks in the sub image, the five-bin values are normalized by the total number of blocks in the sub image. Finally, the normalized bin values are quantized for the binary representation. These normalized and quantized bins constitute the EHD.

IV. PROPOSED APPROACH

Particle Swarm Optimization (PSO) is a technique used to explore the search space of a given problem to find the settings or parameters required to maximize a particular objective. This technique, first described by James Kennedy and Russell C. Eberhart in 1995 [18], originates from two separate concepts: the idea of swarm intelligence based off the observation of swarming habits by certain kinds of animals (such as birds and fish); and the field of evolutionary computation. This short tutorial first discusses optimization in general terms, then describes the basics of the particle swarm optimization algorithm.

A. Optimization:

Optimization is the mechanism by which one finds the maximum or minimum value of a function or process. This mechanism is used in fields such as physics, chemistry, economics, and engineering where the goalis to maximize efficiency, production, or some other measure. Optimization can refer to either minimization or maximization; maximization of a function f is equivalent to minimization of the opposite of this
function, −f .
Mathematically, a minimization task is defined as:

Given f : Rn ! R

Find ˆx 2 Rn such that f(ˆx) _ f(x), 8x 2 Rn

Similarly, a maximization task is defined as:

Given f : Rn ! R

Find ˆx 2 Rn such that f(ˆx) _ f(x), 8x 2 Rn

The domain Rn of f is referred to as the search space (or parameter space). Each element of Rn is called a candidate solution in the search space, with ˆx being the optimal solution. The value n denotes the number of dimensions of the search space, and thus the number of parameters involved in the optimization problem. The function f is called the objective function, which maps the search space to the function space. Since a function has only one output, this function space is usually one-dimensional. The function space is then mapped to the one-dimensional fitness space, providing a single fitness value for each set of parameters.

This single fitness value determines the optimality of the set of parameters for the desired task. In most cases, including all the cases discussed in this paper, the function space can be directly mapped to the fitness space. However, the distinction between function space and fitness space is important in cases such as multi objective optimization tasks, which include several objective functions drawing input from the same parameter space [2, 5]. For a known (differentiable) function f, calculus can fairly easily provide us with the minima and maxima of f. However, in real-life optimization tasks, this objective function f is often not directly known. Instead, the objective function is a "black box" to which we apply parameters (the candidate solution) and receive an output value. The result of this evaluation of a candidate solution becomes the solution's fitness. The final goal of an optimization task is to find the parameters in the search space that maximize or minimize this fitness. In some optimization tasks, called constrained optimization tasks, the elements in a candidate solution can be subject to certain constraints (such as being greater than or less than zero). For the purposes of this paper, we will focus on unconstrained optimization tasks. A simple example of function optimization can be seen in Figure 1.

This figure shows a selected region the function f, demonstrated as the curve seen in the diagram. This function maps from a one-dimensional parameter space—the set of real numbers R on the horizontal x-axis—to a one-dimensional function space—the set of real numbers R on the vertical y-axis. The x-axis represents the candidate solutions, and the y-axis represents the results of the objective function when applied to these candidate solutions. This type of diagram demonstrates what is called the fitness landscape of an optimization problem. The fitness landscape plots the n-dimensional parameter space against the one-dimensional fitness for each of these parameters.

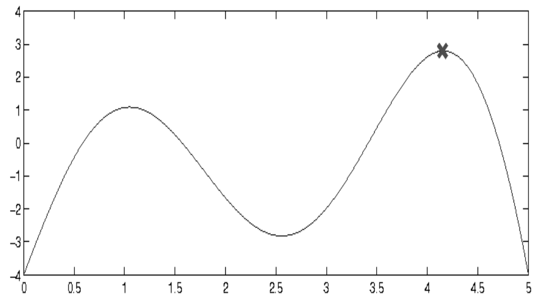

Figure 1: Function Maximum

Figure 1 also shows the presence of a local maximum in addition to the marked global maximum. A local maximum is a candidate solution that has a higher value from the objective function than any candidate solution in a particular region of the search space. For example, if we choose the interval [0,2.5] in Figure 1, the objective function has a local maximum located at the approximate value x = 1.05. Many optimization algorithms are only designed to find the local maximum, ignoring other local maxima and the global maximum. However, the PSO algorithm as described in this paper is intended to find the global maximum.

B. PSO Algorithm:

The PSO algorithm works by simultaneously maintaining several candidate solutions in the search space. During each iteration of the algorithm, each candidate solution is evaluated by the objective function being optimized, determining the fitness of that solution. Each candidate solution can be thought of as a particle "flying" through the fitness landscape finding the maximum or minimum of the objective function.

Initially, the PSO algorithm chooses candidate solutions randomly within the search space. Figure 2 shows the initial state of a four-particle PSO algorithm seeking the global maximum in a one-dimensional search space.

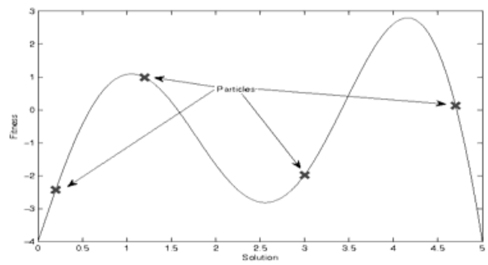

Figure 2: Initial PSO State

The search space is composed of all the possible solutions along the x-axis; the curve denotes the objective function. It should be noted that the PSO algorithm has no knowledge of the underlying objective function, and thus has no way of knowing if any of the candidate solutions are near to or far away 2 from a local or global maximum. The PSO algorithm simply uses the objective function to evaluate its candidate solutions, and operates upon the resultant fitness values. Each particle maintains its position, composed of the candidate solution and its evaluated fitness, and its velocity. Additionally, it remembers the best fitness value it has achieved thus far during the operation of the algorithm, referred to as the individual best fitness, and the candidate solution that achieved this fitness, referred to as the individual best position or individual best candidate solution. Finally, the PSO algorithm maintains the best fitness value achieved among all particles in the swarm, called the global best fitness, and the candidate solution that achieved this fitness, called the global best position or global best candidate solution. The PSO algorithm consists of just three steps, which are repeated until some stopping condition is met:

a. Evaluate the fitness of each particle
b. Update individual and global best fitnesses and positions
c. Update velocity and position of each particle

The first two steps are fairly trivial. Fitness evaluation is conducted by supplying the candidate solution to the objective function. Individual and global best fitnesses and positions are updated by comparing the newly evaluated fitnesses against the previous individual and global best fitnesses, and replacing the best fitnesses and positions as necessary.

C. PSO Variations:

Apart from the canonical PSO algorithm, many variations of the PSO algorithm exist. For instance, the inertia weight coefficient was originally not a part of the PSO algorithm, but was a later modification that became generally accepted. Additionally, some variations of the PSO do not include a single global best aspect of the algorithm, and instead use multiple global best that are shared by separate subpopulations of the particles.

D. Stastical Wavelet Based Analysis:

An extensive image database is created, using Discrete Wavelet Transform, the approximation and detailed co-efficient are obtained, for these coefficients, statistical parameters (such as average, Variance, mean) are determined, a suitable threshold is fixed, the statistical parameters of the key image are compared with that of the image database, decision is made based on the threshold value and related database images are displayed.

E. Wavelet Adaptive Threshold:

Thresholding is one of the techniques for image segmentation. It differentiates the image regions as objects or background. Global thresholding is suitable when the region of interest has a constant gray level and the background also with more or less same intensity. But in certain cases the background intensities are not constant and the intensities of the objects are also variant. Hence, a threshold that is suitable for one area may not work for another area.

We go for adaptive thresholding because images in the database may be with different intensities for which local or global thresholding is not suitable. The adaptive threshold depends on the spatial co–ordinates of the pixel, its intensity and as well as the local characteristics.

In this paper, adaptive thresholding is employed on wavelets to calculate the threshold value. The threshold value is obtained by performing an equation on each pixel with its neighboring pixels. Two mask operators are employed to obtain such an equation and the threshold value is calculated for each pixel in the 3 detail sub-bands. Basically, the adaptive thresholding method assigns different threshold values for different images.

V. CONCLUSION

In this paper**,** has presented a user-oriented framework in interactive CBIR system. In contrast to conventional approaches that are based on visual features, our method provides an interactive mechanism to bridge the gap between the visual features and the human perception. The color distributions, the mean value, the standard deviation, and image bitmap are used as color information of an image. In addition, the entropy and edge histogram are considered as texture descriptors to help characterize the images. In particular, the PSO can be considered and used as a semi-automated exploration tool with the help of a user that can navigate a complex universe of images. Experimental results of the proposed approach have shown the significant improvement in retrieval performance. Further work can be extended to a comparison of various evolutionary algorithms with the performance of PSO to test the proposed system accuracy.

VI. ACKNOWLEDGMENT

I must thankful to Mrs. UMARANI SRIKANTH.M.E., (Ph.D) Head of the Department, PG studies and project coordinator Mrs N.MAGADEVI.M.E Assistant Professor, Department of PG studies, without whose guidance and patience, this dissertation would not be possible. And engineering, project panel members, Professors of the Department of Electrical and Electronics Engineering for their consistent encouragement and ideas.

VII. REFERENCES

[1]. M. Antonelli, S. G. Dellepiane, and M. Goccia, “Design and implementation of Web-based systems for image segmentation and CBIR,” IEEE Trans. Instrum. Meas., vol. 55, no. 6, pp. 1869–1877, Dec. 2006.

[2]. N. Jhanwar, S. Chaudhuri, G. Seetharaman, and B. Zavidovique, “Content based image retrieval using motif cooccurrence matrix,” Image Vis. Comput., vol. 22, no. 14, pp. 1211–1220, Dec. 2004.

[3]. J. Han, K. N. Ngan, M. Li, and H.-J. Zhang, “A memory learning framework for effective image retrieval,” IEEE Trans. Image Process., vol. 14,no. 4, pp. 511–524, Apr. 2005.

[4]. H. Takagi, S.-B. Cho, and T. Noda, “Evaluation of an IGA-based image retrieval system using wavelet coefficients,” in Proc. IEEE Int. Fuzzy Syst.Conf., 1999, vol. 3, pp. 1775–1780.

[5]. H. Takagi, “Interactive evolutionary computation: Fusion of the capacities of EC optimization and human evaluation,” Proc. IEEE, vol. 89, no. 9, pp. 1275–1296, Sep. 2001.

[6]. S.-B. Cho and J.-Y. Lee, “A human-oriented image retrieval system using interactive genetic algorithm,” IEEE Trans. Syst., Man, Cybern. A, Syst., Humans, vol. 32, no. 3, pp. 452–458, May 2002.

[7]. Y. Liu, D. Zhang, G. Lu, andW.-Y.Ma, “A survey of content-based image retrieval with high-level semantics,” Pattern Recognit., vol. 40, no. 1, pp. 262–282, Jan. 2007.

[8]. A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, “Content-based image retrieval at the end of the early years,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 12, pp. 1349–1380, Dec. 2000.

[9]. S. Antani, R. Kasturi, and R. Jain, “A survey of the use of pattern recognition methods for abstraction, indexing and retrieval of images and video,” Pattern Recognit., vol. 35, no. 4, pp. 945–965, Apr. 2002.

[10]. X. S. Zhou and T. S. Huang, “Relevance feedback in content-based image retrieval: Some recent advances,” Inf. Sci., vol. 148, no. 1–4, pp. 129–137, Dec. 2002.

[11]. H.-W. Yoo, H.-S. Park, and D.-S. Jang, “Expert system for color image retrieval,” Expert Syst. Appl., vol. 28, no. 2, pp. 347–357, Feb. 2005.

[12]. T.-C. Lu and C.-C. Chang, “Color image retrieval technique based on color features and image bitmap,” Inf. Process. Manage., vol. 43, no. 2, pp. 461–472, Mar. 2007.

[13]. A. Vadivel, S. Sural, and A. K. Majumdar, “An integrated color and intensity co-occurrence matrix,” Pattern Recognit. Lett., vol. 28, no. 8, pp. 974–983, Jun. 2007.

[14]. M. H. Pi, C. S. Tong, S. K. Choy, and H. Zhang, “A fast and effective model for wavelet subband histograms and its application in texture image retrieval,” IEEE Trans. Image Process., vol. 15, no. 10, pp. 3078–3088, Oct. 2006.

[15]. M. Kokare, P. K. Biswas, and B. N. Chatterji, “Texture image retrieval using new rotated complex wavelet filters,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 35, no. 6, pp. 1168–1178, Dec. 2005.

[16]. M. Pi and H. Li, “Fractal indexing with the joint statistical properties and its application in texture image retrieval,” IET Image Process., vol. 2, no. 4, pp. 218–230, Aug. 2008.

[17]. S. Liapis and G. Tziritas, “Color and texture image retrieval using chromaticity histograms and wavelet frames,” IEEE Trans. Multimedia, vol. 6, no. 5, pp. 676–686, Oct.2004.

[18]. James Kennedy and Russell Eberhart. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, volume IV, pages 1942–1948, Piscataway, NJ, 1995. IEEE Press.

[19]. James Kennedy, Russell Eberhart, and Yuhui Shi. Swarm Intelligence. Morgan Kaufmann, 2001.

[20]. Yuhui Shi and Russell Eberhart. A modified particle swarm optimizer. Proceedings of the IEEE International Conference on Evolutionary Computation, pages 69–73, 1998.

[21]. Frans van den Bergh. An Analysis of Particle Swarm Optimizers. PhD thesis, University of Pretoria, 2001.

[22]. E. Zitzler, M. Laumanns, and S. Bleuler. A tutorial on evolutionary multiobjective optimization, 2002.