Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.1053 – 1060

RESEARCH ARTICLE

AN EFFICIENT APPROACH TO AN IMAGE RETRIEVAL USING PARTICLE SWARM OPTIMIZATION

G. Saranya[1], K. Anitha[2], A. Chilambuchelvan[3]

[1]Student, Department of Computer Science and Engineering, R.M.K. Engineering College,
Anna University, Thiruvallur Dist -601206, T.N., INDIA
[2]Associate Professor, Department of Computer Science and Engineering, R.M.K. Engineering College,
Anna University, Thiruvallur Dist -601206, T.N., INDIA
[3]Professor, Department of Computer Science and Engineering, R.M.K. Engineering College,
Anna University, Thiruvallur Dist -601206, T.N., INDIA
[1] saranyagp86@gmail.com

Abstract: Particle swarm optimization (PSO) is a way to grasp user's semantics through optimized iterative learning. In this paper an adaptive retrieval approach based on the concept of particle swarm optimization is introduced. The content can be in the form of objects, colors, textures, shapes as well as relation between them. By understanding the subjective meaning of a visual query, by converting it into numerical parameters that can be extracted and compared by a computer, is the paramount challenge in the field of intelligent image retrieval. The best compared method reaches its performance at convergence after 10 iterations, while the evolutionary PSO reaches the same result after half of the iterations, then continuing its growth. When a query image is given, only related images are displayed. And hence it reduces work load. This method is robust, reliable and flexible and time efficient for retrieval of images in an efficient way.

Keywords: Content Based Image Retrieval (CBIR), Particle Swarm Optimization (PSO)

I. INTRODUCTION

Storage and retrieval of images in digital libraries has become a real demand in industrial, medical, and other applications. Retrieval can be made by textual descriptors; here words are used to retrieve images in which the result is not precise. Retrieval can be made by query i.e. giving an example image as query. CBIR is more efficient and practical because it is easiest method compared to other techniques, it make use of low level features such as color, shape, texture and size to describe images. CBIR systems analyze the visual content description to organize and find images in database. The retrieval process usually relies on presenting a visual query to the system, and extracting from a database the set of images that best fit the user request. In order to improve the retrieval accuracy of content-based image retrieval systems, several approaches have been designed for sophisticated low-level feature extraction and the 'semantic gap' reduction between the visual features and the richness of human semantics. In this paper, we propose a hybrid approach combining Particle Swarm Optimization (PSO) and Support Vector Machine (SVM) for image retrieval and clustering.

Recent studies in the field of computer shows a great amount of interest in content retrieval from images and videos [1, 2, 3]. This content can be in the form of objects, colors, textures, shapes as well as relation between them.

The algorithm keeps track of three global variables: Target value or condition, Global best (gBest) value indicating which particle's data is currently closest to the Target, And Stopping value indicating when the algorithm should stop if the Target isn't found. Each particle consists of: Data representing a possible solution, A Velocity value indicating how much the Data can be changed, A personal best (pBest) value indicating the closest the particle's Data has ever come to the Target. Each particle's pBest value only indicates the closest the data has ever come to the target since the algorithm started. The gBest value only changes when any particle's pBest value comes closer to the target than gBest. Through each iteration of the algorithm, gBest gradually moves closer and closer to the target until one of the particles reaches the target. It's also common to see PSO algorithms using population topologies, or "neighborhoods", which can be smaller, localized subsets of the global best value. These neighborhoods can involve two or more particles which are predetermined to act together, or subsets of the search space that particles happen into during testing. The use of neighborhoods often helps the algorithm to avoid getting stuck in local minima.

II. RELATED SEARCH

1. Anelia Grigorava discussed an adaptive approach retrieval approach based on the concept of relevance feedback and particle swam optimization, which establishes a link between high level concepts and low level features, but also to dynamically select them within a large collection of parameters. The target is to identify a set of relevant features according to the user query.

2. A.W Smeulders proposed an approach which highlights different problems related to content based image retrieval. The paper starts with discussing the working conditions of content based retrieval, patterns of use and types of pictures. Even though content based image retrieval plays an important role, It suffers from few disadvantages that many number of iterations take place which leads to traffic or congestion. So to avoid this, many techniques and approaches has implemented.

III. PARTICLE SWARM OPTIMIZATION

The algorithm keeps track of three global variables: Target value or condition, Global best (gBest) value indicating which particle's data is currently closest to the Target, And Stopping value indicating when the algorithm should stop if the Target isn't found. Each particle consists of: Data representing a possible solution, A Velocity value indicating how much the Data can be changed, A personal best (pBest) value indicating the closest the particle's Data has ever come to the Target. Each particle's pBest value only indicates the closest the data has ever come to the target since the algorithm started. The gBest value only changes when any particle's pBest value comes closer to the target than gBest. Through each iteration of the algorithm, gBest gradually moves closer and closer to the target until one of the particles reaches the target. It's also common to see PSO algorithms using population topologies, or "neighborhoods", which can be smaller, localized subsets of the global best value. These neighborhoods can involve two or more particles which are predetermined to act together, or subsets of the search space that particles happen into during testing. The use of neighborhoods often helps the algorithm to avoid getting stuck in local minima.

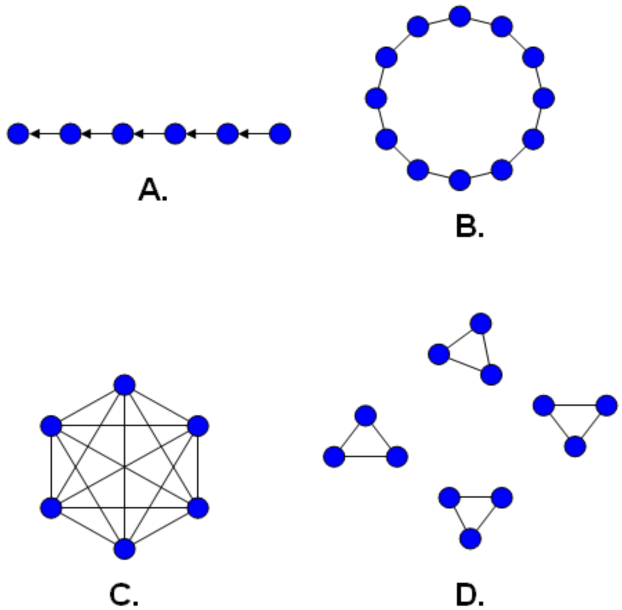


Fig 1 Representation of particle swarm optimization

A few common population topologies (neighborhoods). (A) Single-sighted, where individuals only compare themselves to the next best. (B) Ring topology, where each individual compares only to those to the left and right. (C) Fully connected topology, where everyone is compared together. (D) Isolated, where individuals only compare to those within specified groups.

Particle Swarm Optimization (PSO) consists of the following steps :

Wiener Filter

The Wiener filter purpose is to reduce the amount of noise present in a signal by comparison with an estimation of the desired noiseless signal. It is based on a statistical approach. Typical filters are designed for a desired frequency response. The Wiener filter approaches filtering from a different angle. One is assumed to have knowledge of the spectral properties of the original signal and the noise, and one seeks the LTI filter whose output would come as close to the original signal as possible. Wiener filters are characterized by the following: Assumption: signal and (additive) noise are stationary linear stochastic processes with known spectral characteristics or known autocorrelation and cross correlation. Requirement: the filter must be physically realizable, i.e. causal (this requirement can be dropped, resulting in a non-causal solution). Performance criteria: minimum mean-square error.

To apply this filters there are two possible ways: By convolution of the filter and the image in the temporal domain. The problem of this way is that the Matlab function conv2() performs the 2D convolution of matrices but if [ma,na] = size(A) and [mb,nb] = size(B), then size(C) = [ma+mb-1,na+nb-1]. However, the resulting image size expected is the same as the input size. There are three flags in order to determine the output of conv2. These flags are:'full' which returns the full 2D convolution (default and presented). same which returns only the central part of the convolution that is the same size as A.

Image Retrieval

In contrast to the text-based approach of the system, 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 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.

Color 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 techniques most commonly used, histogram intersection, 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. The results from some of these systems can look quite impressive.

Texture Based 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 has 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 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, which retrieves textured regions in images on the basis of similarity to automatically-derived code words representing important classes of texture within the collection.

Shape Based retrieval

The ability to retrieve by shape is perhaps the most obvious requirement at the primitive level. Unlike texture, shape is a fairly well-defined concept – and there is considerable evidence that natural objects are primarily recognized by their shape A number of features characteristic of object shape (but independent of size or orientation) are computed for every object identified within each stored image. Queries are then answered by computing the same set of features for the query image, and retrieving those stored images whose features most closely match those of the query. Two main types of shape feature are commonly used – *global* features such as aspect ratio, circularity and moment invariants and *local* features such as sets of consecutive boundary segment. Alternative methods proposed for shape matching have included elastic deformation of templates comparison of directional histograms of edges extracted from the image and *shocks*, skeletal representations of object shape that can be compared using graph matching technique. Queries to shape retrieval systems are formulated either by identifying an example image to act as the query, or as a user-drawn sketch.

Shape matching of three-dimensional objects is a more challenging task – particularly where only a single 2-D view of the object in question is available. While no general solution to this problem is possible, some useful inroads have been made into the problem of identifying at least some instances of a given object from different viewpoints. One approach has been to build up a set of plausible 3-D models from the available 2-D image, and match them with other models in the database. Another is to generate a series of alternative 2-D views of each database object, each of which is matched with the query image Related research issues in this area include defining 3-D shape similarity measures and providing a means for users to formulate 3-D shape queries.

Modified Shape Feature Extraction

The image is divided into 8x8 blocks and the two neighbouring blocks are considered simultaneously to determine the structure of the object. Here Edge Elements Extraction by Thresholding technique is used. Most of the edge detection techniques have two steps finding the rate of change of gray levels, i.e. the gradient of the image, and extracting the edge elements for which gradient exceeds a predefined threshold. There are various methods for obtaining the gradient g‟ row, column. A binary image e row, column where pixels row, column contains a label „1‟ if g row, column is an edge pixel or a label „0‟ otherwise. So edge row, column may be obtained as Edge row, column= 1 So, the image subset Se contains only edge elements of g row, column. Here t row, column is the threshold at the pixel r, c and can be found out using the relation T r, c = Φth row, column, Qp row, column Where Qp nrow, column denotes the set of features at pixel row, column. Depending on the variables to determine t row, column, the threshold is called global or space-invariant, local or space-variant and adoptive. In brief, the edge extraction technique then reduces the selection of threshold that transforms The operator Tth satisfies the following two conditions

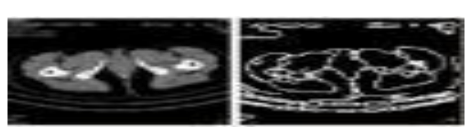
1.Tth is not invertible since it is not one-one

2. Tth can take any value t row, column as threshold from the interval [min row, column g row, column, max row, column g row, column. The Same Strategy is implemented using the Formula Igy=0.2989* Ir+ 0.5870*Ig+ 0.1140*Ib The above equation is the for converting RGB color image to gray scale image. The image I is converted to gray scale image gy and same procedure is used as in.

Texture Feature Co-Occurrence Image Retrieval

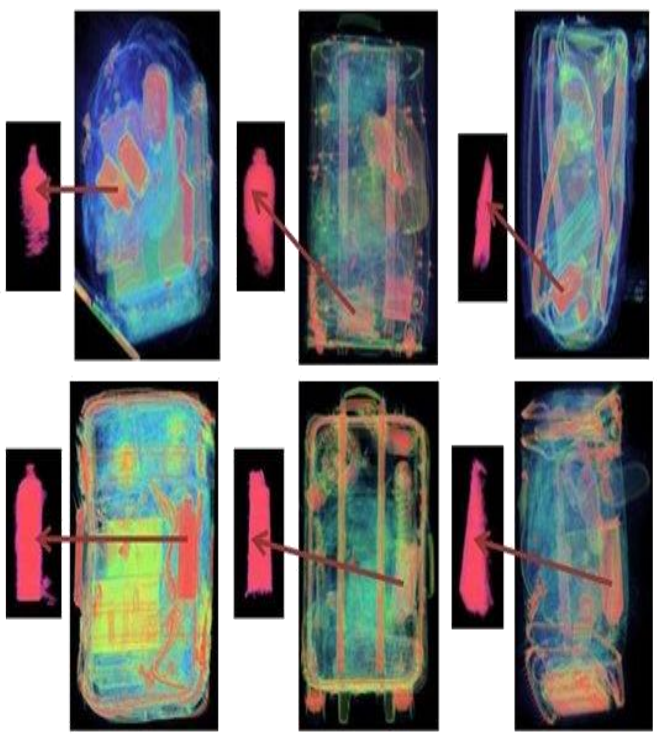
Statistical features of grey levels were one of the earliest methods used to classify textures. Haralick suggested the use of grey level co-occurrence matrices (GLCM) to extract second order statistics from an image. GLCMs have been used very successfully for texture classification in evaluations. Haralick defined the GLCM as a matrix of frequencies at which two pixels, separated by a certain vector, occur in the image. The distribution in the matrix will depend on the angular and distance relationship between pixels. Varying the vector used allows the capturing of different texture characteristics. Once the GLCM has been created, various features can be computed from it. These have been classified into four groups: visual texture characteristics, statistics, information theory and information measures of correlation. We chose the four most commonly used features, listed in Table 1, for our evaluation.

Tamura Image is a notion where we calculate a value for the three features at each pixel and treat these as a spatial joint coarseness-contrast-directionality (CND) distribution, in the same way as images can be viewed as spatial joint RGB distributions. We extract color histogram style features from the Tamura CND image, both marginal and 3D histograms. The regional nature of texture meant that the values at each pixel were computed over a window. A similar 3D histogram feature is used by MARS



Histogram Shape Image Retrieval

Our histogram-based querying approach is basically motivated with the importance of the interrelation among pixels, thus objects are processed in such a way that each pixel provides a piece of information about both the shape and color content of the objects. The human vision system identifies objects with the edges they contain, both on the boundary and in the interior based on the intensity differentiations among pixels. These intensity differentiations are captured with respect to the center of mass of these pixels. Hence, the histogram-based approach processes the shape and color content of the objects similar to the human vision system via considering the pixels in the interior and on the boundary of the objects with respect to the center of mass. the histogram-based querying process can be divided into three phases: object extraction, histogram construction, and querying. In the object extraction phase, the color information is encoded in HSV color space by transforming RGB color vectors for each pixel to HSV. Then, color quantization and color median filtering are applied. The next step is the histogram construction phase where the color information of the filtered image is stored in one color histogram and the shape information is stored in two specialized histograms. The last step, the querying phase, involves an integrated use of the interface and the querying sub-system, in which the user specifies an example for the query. The users may also build up queries by sketching geometric primitives such as rectangles, circles, triangles, etc. The specified query is also passed through the object extraction and histogram construction phases. The results of the query are the 'most similar' objects to the given query object. The three histograms used for storing shape and color information can be described as follows



IV. EXPERIMENTAL RESULTS

The result obtained by using particle swarm optimization algorithm is more robust and advantageous when compared to all the other methods. This is time consuming method, as it predicts the related images very earlier. The images obtained by this method are precise and relevant.

V. FUTURE ENHANCEMENTS

Our future aim is to test the efficiency of particle swarm optimization via other techniques. And finally prove the proposed method reaches the result very quicker and also reduces the workload of users.

VI. CONCLUSION

The best compared method reaches its performance at convergence after 10 iterations, while the evolutionary PSO reaches the same result after half of the iterations, then continuing its growth. A classical deterministic algorithm such as query shifting remains trapped in a local minimum very soon, and the proposed method is able to achieve a similar result after just three to four iterations. As per the user feedback PSO works and performs number of iterations till user obtain the specified result. Feature extraction is the main module description used in PSO for performing iterations. Only related images are retrieved while using this approach.

REFERENCES

[1]. Arnold W.M. Smeulders , Marcel Worring, Simone Santini, Amarnath Gupta , and Ramesh Jain "Content-Based Image Retrieval at the End of the Early Years IEEE Transactions on Pattern analysis and machine intelligence, vol. 22, NO. 12, DECEMBER 2000

[2]. A. Grigorova, F. G. B. De Natale, C. Dagli, and T. S. Huang, "Content based image retrieval by feature adaptation and relevance feedback," IEEE Trans. Multimedia, vol. 9, no. 6, pp. 1183–1192, Oct. 2007.

[3]. A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain,"Content-based image retrieval at the end of early years," IEEE Trans.Pattern Anal. Mach. Intell., vol. 22, no. 12, pp. 1340–1380, Dec. 2006.

[4]. D. Bratton and J. Kennedy, "Defining a standard for particle swarm optimization," in Proc. IEEE Swarm Intelligence Symp., 2007.

[5]. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitism multiobjective genetic algorithm: NSGA-II," IEEE Trans. Evol. Comput.,vol. 6, no. 2, pp. 182–197, Apr. 2008.

[6]. Gerassimos barlas, "An analytical approach to optimizing parallel image registration / retrieval", IEEE transactions on parallel and distributed systems, vol. 21, no. 8, august 2010.

[7]. K. Chandramouli, "Particle swarm optimization and self-organizing maps based image classifier," in Proc. 2nd Int. Workshop Semantic Media Adaptation and Personalization, 2007, pp. 225–228

[8]. K. Chandramouli, T. Kliegr, J. Nemrava, V. Svatek, and E. Izquierdo, "Query refinement and user relevance feedback for contextualized image retrieval," in Proc. VIE 08, 2008.

[9]. Mahdi Saadatmand-Tarzjan and Hamid Abrishami Moghaddam,"A Novel Evolutionary Approach for Optimizing Content-Based Image Indexing Algorithms", IEEE Transactions On Systems, Man, And Cybernetics—part B: Cybernetics, Vol. 37, No. 1, February 2007.

[10]. Mattia Broilo,, and Francesco G. B. De Natale, " A Stochastic Approach to Image Retrieval Using Relevance Feedback and Particle Swarm Optimization" IEEE Transactions on multimedia , JUNE 2010

[11]. M. Broilo, P. Rocca, and F. G. B. De Natale, "Content-based image retrieval by a semi-supervised particle swarm optimization," in Proc.Int. Workshop Multimedia Signal Processing, 2008.

[12]. M. L. Kerfi and D. Ziou, "Image retrieval based on feature weighing and relevance feedback," in Proc. IEEE Int. Conf. Image Processing (ICIP2004), 2004, vol. 1, pp. 689–692.

[13]. R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," ACM Comput. Surv., vol. 40, no. 2, Apr. 2008, article 5

