COMPUTATIONAL INTELLIGENCE TECHNIQUES AND THEIR APPLICATIONS IN CONTENT-BASED IMAGE RETRIEVAL

[a]*Kambiz Jarrah,* [b]*Matthew Kyan*, [a]*Sri Krishnan, and* [a]*Ling Guan*

[a] Department of Electrical and Computer Engineering, Ryerson University, Toronto, Canada
[b] School of Electrical and Information Engineering, University of Sydney, Australia

ABSTRACT

The main focus of this paper is to present a methodology for optimizing relevance identification in content-based image retrieval (CBIR) systems through the principle of feature weight detection. The purpose of relevance identification is to find a collection of images that are statistically similar to, or match with, an original query image within a large visual database. The novelty of this scheme is two-fold: using a base-10 Genetic Algorithm method to accurately determine the contribution of individual feature vectors for a successful retrieval in the so-called feature weight detection process, and defining a new unsupervised learning algorithm, the Directed self-organizing tree map (DSOTM), for the purpose of classification in the automatic relevance identification module of the search engine. Comprehensive experiments demonstrate feasibility of the proposed methodology.

1. INTRODUCTION

One of the most important issues in constructing CBIR systems is to emphasize a subset of more discriminative feature descriptors that are able to characterize similar images in a more significant way than that of the original feature space. It is usually thought that applying more features to describe individual images will result in constructing a better retrieval system. It has been observed that beyond a certain point, inclusion of additional features leads to worse rather than better performance results. Moreover, the choice of features to represent the patterns affects several aspects of the pattern recognition problem such as accuracy, required learning time, and necessary number of samples [1] and [2].

Feature weight detection refers to the task of identifying and selecting an effective subset of features to represent pattern similarities from a larger set of redundant or even irrelevant features. Multiple criteria influence the weight detection process including relationships among individual features and accuracy of classification. Genetic algorithms offer a particularly attractive approach to address these criteria; since they are generally quite effective in the rapid global search of large and nonlinear spaces.

In the past [3], we developed an automatic method based on an unsupervised learning approach using a combination of a Multi-Class and Two-Class self-organizing tree maps (SOTM). In that work, the process of weight detection involved in repeatedly assigning different weights to each feature descriptor was studied, along with the system's behaviour against imposed feature weights.

In this paper, the above methodology is replaced with an evolutionary process, the Genetic Algorithm (GA) [1]: a method capable of detecting multiple feature weights for all three types of descriptors (colour, texture, and shape) fast, concurrently, and effectively. Additionally, a new unsupervised clustering approach is proposed to track query position in order to direct the spawning of new centres accordingly. We have called this clustering approach the Directed SOTM (DSOTM). Moreover, similarity is measured as the tree structure of the DSOTM grows. Experimental results demonstrate feasibility of the proposed methods for designing a more robust image retrieval system.

2. AUTOMATIC RELEVANCE FEEDBACK

GA-based automatic retrieval system (GA-CBIR) uses an unsupervised machine learning algorithm to identify relevancy of retrieved images, as an alternative for users' relevance feedbacks [4] (see Fig. 1.)

In Fig. 1, the second unit of *Initial Search* module, the *Feature Extraction 1*, deals with calculating features from a high volume image database. Consequently, standard set of content descriptors (an example might be MPEG-7), are extracted in this module to provide a more generic and rapid interface to existing databases based on the chosen standard. Extracted features are used to retrieve the most similar images (in the relative sense) based on predefined distance metric. The top *Q* images are then displayed back to the user through an *Interface* block. Subsequently, the user may request an automatic search, which operates independently. Upon this request, control of the system is switched to path *(a)* of the *GA-based automatic search* module, wherein, another set of features (usually of higher perceptual quality) are extracted from the top *Q* retrieved images from the initial search, using *Feature Extraction 2*. Although computation of those features could be intensive, they become feasible at this stage since such features are only computed for a few retrieved images from the previously ranked-based process. This module allows for the use of more proprietary or specialty features which may enhance perceptual discrimination beyond that which might otherwise be possible through standard features alone. These features are then used as seeds to train the unsupervised classifier.

In particular, Colour Histograms, Colour Moments, Wavelet Moments, and Fourier Descriptors were used in the *Feature Extraction 1*, whereas Hu's seven moment invariants (HSMI) and Gabor Descriptors accompanied with Colour Histograms and Colour Moments were used in *Feature Extraction 2*.

Colour histograms and colour moments were computed in the HSV and RGB colour spaces. Wavelet Moments were extracted from mean, μ, and standard deviation, σ, of three-level wavelet transform applied on an image. Boundary-based Fourier shape parameters were extracted by converting the edge parameters from Cartesian to Polar coordinates and, subsequently, applying the Fast Fourier Transform (FFT) to obtain top 10 low-frequency components. Texture features were computed from μ and σ of Gabor filtered images to construct 48 dimensional feature vectors, and finally, region-based HSMI shape parameters were extracted

1-4244-0367-7/06/$20.00 ©2006 IEEE

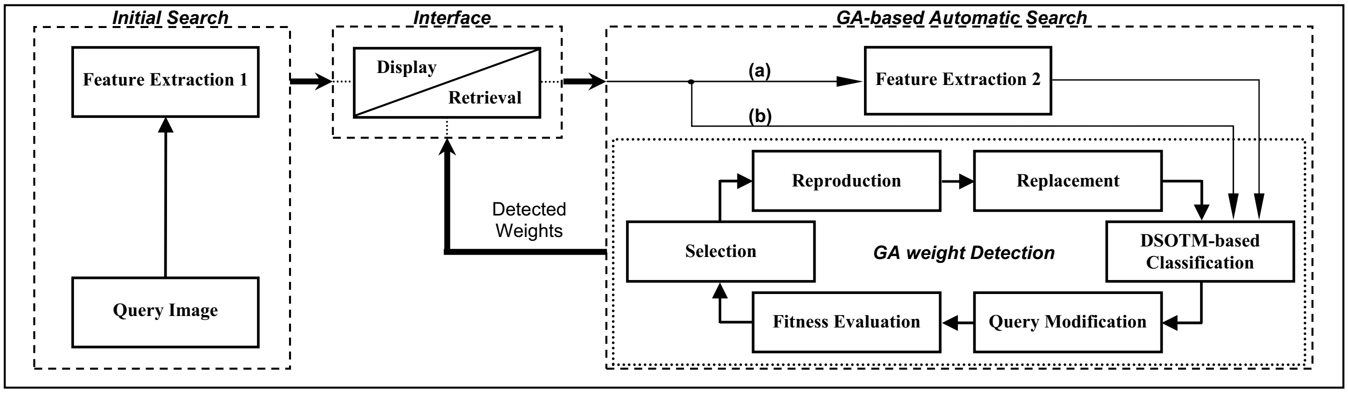

Fig. 1: GA-CBIR System.

by converting the colour images into binary segmented images and then extracting the shape parameters from those segmented images.

In the GA Weight Detection module, a genetic-based algorithm is used to accurately detect feature weights with regard to past behaviours of the system. An initial population made up of strings of integers (chromosomes) is randomly generated. Each chromosome is composed of digits (genes) which can take on different values. Each chromosome represents a given number of traits: the feature weights in our case, which are the actual parameters to be varied to optimize the fitness function. In this process, the system is first transferred to the distance measurement phase to identify the top Q similar images. System's response to different weights is subsequently quantified by measuring the ability of *DSOTM* in correctly classifying selected images by taking the path (b.) A new query, based on selected images from the previous iterations, is then adapted to substantially represent the relevant class through the *Query Modification* module [4]. The system response to identified weights, in terms of retrieval accuracy, is next detected by the system based upon the pre-classified database in the *Fitness Evaluation* block. A new population is next generated in the *Replacement* module by preparing the fittest chromosomes to breed through the *Selection* and *Reproduction* block. The detected feature weights as well as final retrieval results are then displayed back to the user upon completion of GA iterations.

2.1. Directed Self-Organizing Tree Maps (DSOTM)

SOTM [5] is an unsupervised machine learning algorithm and is inspired by principles found in Kohonen's self-organizing feature map (SOFM.) The tree structure of the SOTM is constructed by randomly selecting an isolated root node (centre) and repeatedly presenting the remaining of the patterns to the network. The pattern (sample) which is found to be closest to the centre with respect to current similarity measurement is declared to be the winner. Every such presentation of input patterns slightly modifies the winning node's position in the network: a position that eventually evolves toward the centre of mass of the current class. This gradual adaptation of the node's position is controlled by an exponential decaying function, the *learning rate*. The learning rate resets to its initial value each time a new centre is generated. Therefore, sufficient time is given to the network to adapt itself to the presence of new samples, thus, the tree grows larger and the similarity measurement tends to be more accurate. The generation of new centres (branches of the tree) is controlled by a hierarchy function, called the *threshold function*, which decreases over time. If an input node is encountered whose similarity exceeds this threshold function (i.e. is significantly different from all nodes in the current SOTM map) a new node is generated. The new node is attached as a leaf node of its closest representation in the current SOTM mapping, thus over time, a tree structure evolves [3]. By preserving the topological relationships between the patterns in the original input space, the tree structure of the SOTM tends to form neighbourhood relationships that reflect a degree of similarity between new and already classified patterns.

Problem with the SOTM algorithm is two-fold: it unsuitably decides on the relevant number of classes; and often loses track of the true query position. Decision on the relevance of clusters in SOTM is postponed until the very last stage of the algorithm. This is because there is no innate controlling process available for the algorithm to influence cluster generation around the query centre. It is possible for new centres to destroy any sense of supervision provided by the initial query location if they force it to a non-relevant region of the feature space. If the query centre deviates substantially from its initial position, the likelihood of a boundary forming within the original query's class is increased. This is undesirable as the centre that is assigned as relevant can represent a sub-portion of the true relevant class. In Fig. 2, the SOTM forms a boundary near the query contaminating relevant samples, where as some supervision is maintained in the DSOTM case, preventing unnecessary boundaries from forming.

Due to the limitations of SOTM, we propose the Directed SOTM (DSOTM) algorithm in this work. In the DSOTM algorithm, decision on association of input pattern to query image is gradually made as each sample is presented to the system. It also contains a controlling mechanism that keeps track of the query centre by forcing the centre of relevant class to remain in the vicinity of the query position. Therefore, it can dynamically control generation of new centres and can determine the relevance of input patterns, with respect to the query features, as the tree structure grows. On the other hand, it limits the synaptic vector adjustments according to its reinforced learning rules and constrains cluster generation by preventing the spawning of redundant centres around the query position; since this part of the map is already occupied by relevant class centre.

The DSOTM algorithm is summarized in the following steps:

Initialization: Choose a root node, $\{\mathbf{w}_i\}_{j=1}^{J}$, from the available set of input vectors, $\{\mathbf{x}_k\}_{k=1}^{K}$, in a random manner. J is the total number of centroids (initially set to 1) and K is the total number of inputs (i.e. images);

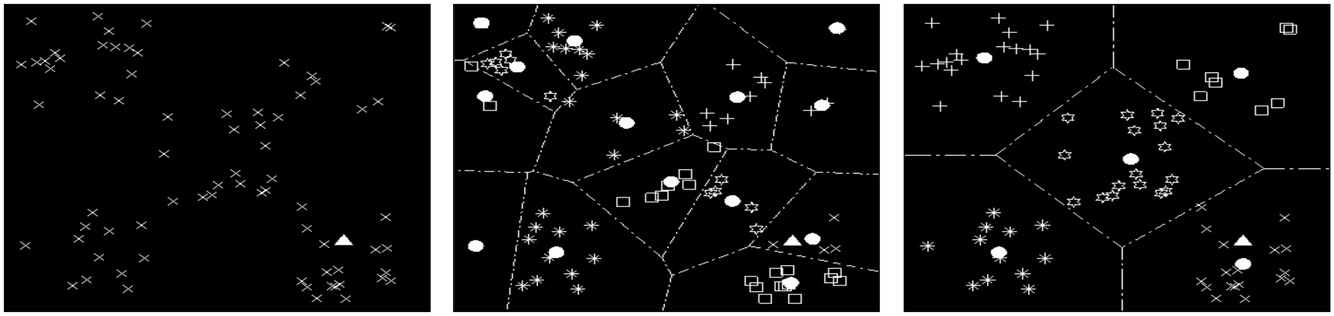

Fig 2: Two-dimensional mapping: (Left) Input pattern with 5 distinct clusters, (Middle) 14 generated centers using SOTM, and (Right) 5 generated centers using DSOTM. Over-classification around the query (triangle) will result in erroneous relevance identification.

Similarity Measurement: Randomly select a new data point, **x**, find the best-matching (winning) centroid, j^*, by minimizing a predefined Euclidean distance criterion in Eq. 1.

$$\mathbf{w}_{j^*}(t) = \arg\min_{j} \left\|\mathbf{x}(t) - \mathbf{w}_j(t)\right\|, \qquad j = 1,2,\ldots,J. \qquad (1)$$

Update: If $\left\|\mathbf{x}(t) - \mathbf{w}_{j^*}\right\| \le H(t)$, where $H(t)$ is the hierarchy function used to control the levels of the tree and decays exponentially over time from its initial value, $H(t_0) > \sigma_{\mathbf{x}}$, according to $H(t+1) = \lambda \cdot H(t) \cdot \exp(-t/\rho)$, where λ is the threshold constant, $0 < \lambda < 1$, and $\rho = \max(t)/\log_{10}[H(t)]$; ***then*** assign $\mathbf{x}(t)$ to the j^{th} centroid, and adjust the synaptic vector according to the reinforce learning rule:

$$\mathbf{w}_{j^*}(t+1) = \mathbf{w}_{j^*}(t) + \alpha(t) \cdot \beta(\mathbf{z},\mathbf{x},t) \cdot [\mathbf{x}(t) - \mathbf{w}_{j^*}(t)], \qquad (2)$$

where α(t) is the learning rate, which exponentially decays over time as more neurons are allocated, $\alpha(t) = \alpha(t_0) \cdot \exp[-t/\max(t)]$, $0.01 \le \alpha(t) \le \alpha(t_0)$ and $\alpha(t_0) = 0.1$; and $\beta(\mathbf{z},\mathbf{x},t)$ is the ranking function that exponentially measures the similarity between query feature vector, **z**, and input feature vector, **x**, at automatic relevance feedback iteration in the previous search operation as is indicated in Eq. 3:

$$\beta(\mathbf{z},\mathbf{x},t) = \sum_{i=1}^{P} G_i(x_i - z_i) = \sum_{i=1}^{P} \exp\left(-\frac{(x_i - z_i)^2}{2\sigma_i^2}\right), \qquad (3)$$

In this equation, P is the total number of features, $\sigma_i = \eta \max_i |x_i - z_i|$ is the tuning parameter, and η is an additional factor to ensure a large output for $\beta(\mathbf{z},\mathbf{x},t)$. Large value of $\beta(\mathbf{z},\mathbf{x},t)$ indicates a high relevance of the feature vector compare to the respective query feature at the time t. As the result, the synaptic vectors are adjusted so that they learn more from statistically similar inputs and less from statistically irrelevant one. ***Else*** form a new centroid node starting with **x**, reset the learning rate to its initial value (i.e. $\alpha(t_0)$), and increment j by 1.

Cluster Adjustment and Relevance Identification: If $\arg\min_{j^*} \left\|\mathbf{z} - \mathbf{w}_{j^*}(t)\right\|$ and $\left\|\mathbf{x}(t) - \mathbf{w}_{j^*}(t)\right\| \le H(t)$ that is if the closest center to the current input data is also closest center to the query, ***then*** mark $\mathbf{x}(t)$ as a relevant sample ***and*** update its centroid (winning neuron) toward the query position according to the degree of resemblance of the sample using:

$$\mathbf{w}_{j^*}(t+1) = \mathbf{w}_{j^*}(t) + \alpha(t) \cdot \beta(\mathbf{z},\mathbf{x},t) \cdot [\mathbf{z} - \mathbf{w}_{j^*}(t)]; \qquad (4)$$

else mark $\mathbf{x}(t)$ as an irrelevant pattern ***and*** update its centroid using Eq. 2. Subsequently, find and move center of relevant class further toward the query center using Eq. 4.

Continuation: Continue with *Similarity Measurement* step until maximum number of iterations is reached, the maximum number of clusters is generated, and/or no noticeable change in the feature map is observed.

In modifying the SOTM algorithm to impose some constraints on cluster generation near the query position we avoid over classification around it. As a result, a better sense of relevance measurements can be achieved as the tree structure forms.

3. FEATURE WEIGHT DETECTION

The purpose of feature weight detection, as a general principle of feature selection, is to de-emphasize certain features that do not necessarily contribute significant information about how relevant an image is in the retrieval process. Previously [3], it was observed that assigning proper weights for individual features can significantly increase performance behavior of CBIR systems in terms of retrieval rate (*RR*). In this work, a GA based approach is proposed to measure importance of individual features and degree of their contribution toward improving the retrieval process and overall performance of the system.

The GA is one of the most commonly used optimization techniques among set of evolutionary schemes. The basic concept of the algorithm is to model Darwin's evolutionary process, where potential solutions within a population compete with one another for survival, based on their inherent behaviour. These solutions are usually encoded as strings called chromosomes, and each chromosome is associated with a fitness value, which indicates its optimality with respect to the current optimization criterion. Hence, the evolutionary process aims to select and reproduce new generations of candidate solutions, which are a better fit with respect to the current fitness criteria.

To avoid the need for encoding and decoding the chromosomes we employed a base-10 GA, which operates similarly to the binary GA described in [1] and [2] except the genes vary over the range between 0 and 9 and there is an extra digit for representing the sign. These genes were combined with each other to create an initial population of M chromosomes in the range of (m,1), where m is a user defined negative value specifying the boundary of navigations for the GA algorithm. An empirical range for m is found to be (-1,0). This range will give the GA variety of options to explore the negative numbers to properly emphasize or de-emphasize features to a greater extent; since input feature vectors were normalized in the range of (-1,1).

A chromosome in the base-10 GA is defined as $C = \{C_{Colour}, C_{Texture}, C_{Shape}\}$, where $m \le C \le 1$. Length of the chromosome depends on the desired precision of weights, in terms of digit decimal fraction. The process of fitness calculation for each chromosome is summarized as follows:

A. Let $\mathbf{z}$ = [$\mathbf{z}_c$, $\mathbf{z}_t$, $\mathbf{z}_s$] represent the selected query image described by its colour, texture, and shape feature vectors;

B. Select the top Q statistically relevant images using minimum Manhattan distance criterion;

C. Create a new query feature vector, $\mathbf{z}_{New}$, where $\mathbf{z}_{New} = [(C_{Colour} \times \mathbf{z}_c), (C_{Texture} \times \mathbf{z}_t), (C_{Shape} \times \mathbf{z}_s)]$;

D. Construct the training matrix, **R**, from the relevant class identified by the DSOTM algorithm [4];

E. Modify the query position to center of mass of the relevant class using the method described in [4];

F. Find response of the GA-CBIR system to presence of new feature weights. Performance is measured based on number of retrieved images, out of Q, that fall into query class;

G. Detect precision of the retrieval, P_r (N_c), in terms of *RR*;

H. Calculate the fitness function, $F(w)$, using:

$$F(w) = \left[\exp(P_r(N_C)\right]^{-1}. \qquad (5)$$

The process of GA weight detection is as follows: ***Initialization:*** create a Random initial population, $P_0 = \{C_1, C_2, \ldots, C_m, \ldots, C_M\}^T$, where M is the population size; ***Evaluation:*** Comput the fitness value (Eq. 5) for each individual chromosome in the current population and score them accordingly. ***Selection:*** Select the fittest members of the current population (parents) for reproduction. This selection is a linear search through a roulette wheel with slots weighted in proportion to chromosomes' fitness scores. The reproduction part of GA is a process in which individual strings are copied according to their fitness to create a mating pool. As a result, chromosomes with higher fitness scores will have higher chances for reproduction. ***Reproduction:*** Create new individuals (children) using the genetic operators – Crossover and Mutation; ***Generation:*** Generate a new population by replacing the children with current population.

The above procedure enables GA to search among possible solutions in the provided range and pick up optimum weights for each individual feature vector to maximize the *RR*.

4. EXPERIMENTAL RESULTS

A number of experiments were conducted to compare the behaviors of the automatic CBIR and GA-CBIR engines using both SOTM and DSOTM clustering algorithms.

The simulations were carried out using a subset of the Corel image database consisting of nearly 5100 JPEG colour images, covering a wide range of real-life photos, from 51 different categories. Each category consisted of 100 visually associated objects to simplify the measurements of the retrieval accuracy during the experiments. 3 sets of 51 images were drawn from the database. In sets *A* and *B* images were randomly selected from the entire set without regard for class, whereas in set *C* images were randomly selected such that no two images were from the same class. Retrieval results were statistically calculated from each of the 3 sets. In the simulations, a total of 16 most relevant images were retrieved to evaluate the performance of the retrieval. In these experiments a base-10 GA algorithm with initial population of 50 chromosomes, defined within the range of $(-0.5,1)$, with maximum of 100 iterations was employed. A single point crossover operation with probability of 80% was utilized in this system. The mutation probability was also set to 5%. A scaling based function was used to make the scaled value of a chromosome proportional to its raw fitness score. The Roulette Wheel method was also used to select chromosomes in proportion to their score as indicated by the fitness function for reproduction. As a result, a chromosome with higher fitness score will have a better chance to be selected for reproduction.

Table 1: Experimental Results in terms of *RR*

Query Sets	CBIR with SOTM	CBIR with DSOTM	GA-CBIR with SOTM	GA-CBIR with DSOTM
A	47.5%	58.0%	66.8%	78.3%
B	47.4%	59.6%	72.1%	76.7%
C	51.0%	56.8%	74.4%	80.5%
Ave.	***48.6%***	***58.1%***	***71.1%***	***78.5%***

The experiment results are illustrated in Table 1. DSOTM outperforms the SOTM in both CBIR and GA-CBIR systems. Effectiveness of using genetic based algorithm in the structure of CBIR systems is also evident in the above results.

The above results only compare CBIR systems with similar architectures. To the best of our knowledge, no other automated relevance feedback CBIR engines, other than the SOTM-based systems being compared in this paper are found in the literature.

5. CONCLUSIONS

In this paper a frame work of the computer controlled CBIR system based on DSOTM and the GA feature weight detection technique was demonstrated. In this work a new unsupervised learning algorithm, DSOTM, which is capable of tracking the query centre and constantly adjusting the spawned centre positions of clusters to avoid over-classification around the query position was defined. As a result of using DSOTM, similarity measurements can also be carried out locally and as each input sample is presented to the system; since the information about the relevant class (query) is constantly provided for the algorithm. The usefulness of using the base-10 Genetic Algorithm for feature weight detection purposes was also studied in this paper. It was mentioned that this method not only takes the relationship between individual features into account to achieve more accurate classification results but also reduces the need for encoding and decoding of chromosomes as opposed to the binary based Genetic Algorithm. The experimental results demonstrate the effectiveness of the proposed methods.

6. REFERENCES

[1] D.E.Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, New York, 1989.

[2] W.R.Smart and M.Zhang, "Classification Strategies for Image Classification in Genetic Algorithm", *IEEE Image and Vision Computing* NZ, Palmerston North, pp 402-407, November 2003.

[3] K.Jarrah, M.Kyan, I.Lee, and L.Guan, "Application of image visual characterization and soft feature selection in Content-Based Image Retrieval", to be appeared in the *International Society for Optical Engineering (SPIE)*, San Jose, California, USA, 2006.

[4] P.Muneesawang and L.Guan, "Minimizing user interaction by automatic and semiautomatic relevance feedback for image retrieval", *Proc. IEEE Int. Conf. on Image Processing, Rochester*, USA, pp. 601-604, vol.2, Sept. 2002.

[5] H.S.Kong, The Self-Organizing Tree Map, and its Applications in Digital Image Processing, PhD Thesis, University of Sydney, Australia, 1998.