반도체및디스플레이장비학회지 제7권 제4호(2008년 12월)
Journal of the Semiconductor & Display Equipment Technology, Vol. 7, No. 4. December 2008.

Genetic Algorithm based Relevance Feedback for Content-based Image Retrieval

Kwang-Kyu Seo[†]

[†]Dept. of Industrial Information & Systems Eng., Sangmyung University
San 98-20, Anso-Dong, Chonan, Chungnam 330-720, Korea

ABSTRACT

This paper explores a content-based image retrieval framework with relevance feedback based on genetic algorithm (GA). This framework adopts GA to learn the user preferences using the similarity functions defined for all available descriptors. The objective of the GA-based learning methods is to learn the user preferences using the similarity functions and to find a descriptor combination function that best represents the user perception. Experiments were performed to validate the proposed frameworks. The experiments employed the natural image databases and color and texture descriptors to represent the content of database images. The proposed frameworks were compared with the other two relevance feedback methods regarding effectiveness in image retrieval tasks. Experiment results demonstrate the superiority of the proposed method.

Key Words : Genetic Algorithm, Relevance Feedback, Content-based Image Retrieval

1. INTRODUCTION

The development of computer and communication technologies makes the demand of multimedia information services significant. Recent research on image retrieval methods has become very important in various areas such as entertainment, education, digital libraries, and medical image retrieval. Content-based image retrieval (CBIR) techniques are becoming increasingly important in multimedia information systems and have the searching process consists of, for a given query image, finding the most similar images stored in the database[1].

Basically, the image retrieval process with relevance feedback is comprised of the following four steps: (i) showing a small number of retrieved images to the user; (ii) user indication of relevant and non-relevant images; (iii) learning the user needs from her feedbacks; and (iv) selecting a new set of images to be shown. This procedure is repeated until a satisfactory result is reached.

An important element of a relevance feedback technique is the learning process. Several relevance feedback methods designed for CBIR systems implement the learning of the user needs by assigning different weights to the descriptors used in the searching process [2, 3, 4]. This strategy allows only a linear combination of the similarity values defined by each descriptor. However, more complex combination functions may be necessary to express specific user visual perceptions. Another common drawback of existing RF methods is concerned with the fact that they, in general, ignore the similarity function defined for each available descriptor. In some RF approaches the learning process is based only on the image feature vectors[5,6,7]. Others define specific distance functions for computing the similarity between two images[2,3]. In both cases, the overall CBIR system effectiveness may decrease if the similarity functions of the descriptors are not used. In fact, the effectiveness of a descriptor depends not only on the feature vector codification, but also on the similarity function defined.

This paper explores a novel relevance feedback

[†]E-mail : kwangkyu@smu.ac.kr

method for interactive image search. The previous GA based relevance feedback[7] is used to GA as a near optimal selection of the feature vectors for image retrieval but the proposed approach adopts GA to learn user preferences in a query session. The core contribution of this paper is the proposal of the novel RF framework that uses GA to find a function that combines non-linearly similarity values computed by different descriptors and the similarity functions defined for each available descriptor. The effectiveness of the proposed methods is compared with other relevance feedback techniques[2,6] for image retrieval tasks.

2. RESEARCH BACKGROUND

2.1. Genetic Algorithm

GA is an artificial intelligence problem-solving technique based on the principles of biological inheritance and evolution. In GA approach, the individuals undergo evolution. The fitness evaluation consists in executing these programs, and measuring their degrees of evolution. GA, then, involves an evolution-directed search in the space of possible computer programs that best solve a given problem.

The fitness of an individual is determined by its effectiveness in producing the correct outputs for all cases in a training set. The training set contains inputs and their previously known correspondent outputs. To evolve the population, and optimize the desired objectives, it is necessary to choose the correct individuals to be subject to genetic operators. Thus, selection operators are employed to select the individuals based on their fitness. Examples of selection method are roulette wheel, tournament, and rank-based selections[8].

Genetic operators introduce variability in the individuals and make evolution possible, which may produce better individuals in posterior generations. The crossover operator exchanges sub-trees from a pair of individuals, generating two others. Mutation operator replaces a randomly chosen sub-tree from an individual by a sub-tree randomly generated. The reproduction operator simply copies individuals and inserts them in the next generation.

2.2. Color Feature

We used HSV color model for representing color because this model is closely related to human visual perception. Color quantization is useful for reducing the calculation cost. Furthermore, it provides better performance in image classification because it can eliminate the detailed color components that can be considered noises. The human visual system is more sensitive to hue than saturation and value so that hue should be quantized finer than saturation and value. In the experiments, we uniformly quantized HSV space into 18 bins for hue (each bin consisting of a range of 20 degree), 3 bins for saturation and 3 bins for value for lower resolution.

In order to represent the local color histogram, we divided image into equal-sized 3×3 rectangular regions and extract HSV joint histogram that has quantized 162 bins for each region. Although these contain local color information, the resulting representation is not compact enough. To obtain compact representation, we extract from each joint histogram the bin that has the maximum peak. Take hue h, saturation s, and value v associated to the bin as representing features in that rectangular region and normalize to be within the same range of[0,1]. Thus, each image has the 3×3×3 (=27) dimensional color vector [1].

2.3. Texture Feature

Texture analysis is an important and useful area of study in computer vision. Most natural images include textures. Scenes containing pictures of wood, grass, etc. can be easily classified based on the texture rather than color or shape. Therefore, it may be useful to extract texture features for image clustering. Like as color feature, we include a texture feature extracted from localized image region.

The co-occurrence matrix is a two-dimensional histogram which estimates the pair-wise statistics of gray level. The $(i,j)^{th}$ element of the co-occurrence matrix represents the estimated probability that gray level i co-occurs with gray level j at a specified displacement d and angle θ. By choosing the values of d and θ, a separate co-occurrence matrix is obtained. From each co-occurrence matrix a number of textural features can be extracted. For image clustering, we used entropy, which is mostly used in many applications. Feature extraction is performed by the following process:

(i) Conversion of color image to gray image

(ii) Dividing image into 3×3 rectangular regions as in color case.

(iii) Obtaining co-occurrence matrix for four (horizontal 0^o, vertical 90^o and two diagonal 45^o and 135^o) orientation in region and normalize entries of four matrices to [0, 1] by dividing each entry by total number of pixels.

(iv) Extracting average entropy value from four matrices.

$$e = \frac{-\sum_k \sum_i \sum_j p(i,j)\log(p(i,j))}{4}, \quad k = 1, 2, 3, 4$$

(v) Constructing texture feature vector by concatenating entropies over all rectangular regions.

Thus, each image has the 3×3 (=9) dimensional texture vector[1].

3. THE GA BASED RF FRAMEWORK FOR CBIR

Let $D^* = (D, \delta_D)$ be a composite descriptor employed to rank N database images defined as DB = $\{db_1, db_2, ..., db_N\}$. The set of K simple descriptors of D^* is represented by $D = \{D_1, D_2, ..., D_K\}$. The similarity between two images I_l and I_m, computed by D_i, is represented by $d_{iI_lI_m}$. All similarities $d_{iI_lI_m}$ are normalized between 0 and 1. A Gaussian normalization can be employed for this task.

Let L be the number of images displayed on each iteration. Let Q be the query pattern $Q = \{q_1, q_2, ..., q_M\}$, where M is the number of elements in Q, formed by the query image q_1 and all images defined as relevant during a retrieval session.

Fig. 1 shows an overview of the retrieval process based on GA proposed in this paper.

The user interactions are indicated in italic. At the beginning of the retrieval process, the user indicates the query image q_1. Based on this image, an initial set of images is selected to be shown to the user. Thus, the user is able to indicate the relevant images, from this initial set, starting the relevance feedback iterations. Each iteration involves the following steps: user indication of relevant images; the update of the query pattern; the learning of the user preference by using GA; database images ranking; and the exhibition of the L most similar images.

User indication of query image q_1
Show the initial set of images
while the user is not satisfied **do**
  User indication of the relevant images
  Update query pattern Q
  Apply GA to find the best individuals (similarity composition functions)
    {Generate a initial population of individuals
     for N generations do
     Calculate the fitness of each individual
       { for all $q_j \in Q$ do
         for all $t_k \in T$ do
           $rk_{j\delta_i}[k].key \leftarrow \delta_i(q_j, t_k)$
           $rk_{j\delta_i}[k].element \leftarrow t_k$
         end for
         Sort $rk_{j\delta_i}$
           $f_{j\delta_i} \leftarrow f(rk_{j\delta_i})$
         end for
         $F_{\delta_i} \leftarrow \frac{\sum_{j=1}^{M} f_{j\delta_i}}{M}$
         return F_{δ_i} }
     Select the individuals to genetic operations
     Apply GA operators (reproduction, crossover and mutation)
     end for }
  Rank the database images
    { for all $\delta_i \in S$ do
         for all $db_j \in DB$ do
           $rk_i[j].key \leftarrow Simility_{\delta_i}(Q, db_j)$
           $rk_i[j].element \leftarrow db_j$
         end for
         Sort $rk_i[j]$ rki[j]
         for $j \leftarrow 1$ to β do
     $votes[rk_i[j].element] \leftarrow votes[rk_i[j].element\} + 1/j$
         end for
       end for
       Sort DB images regarding their votes
       return the L most voted images }
  Show the L most similar images
end **while**

Fig. 1. The proposed algorithm of the retrieval process based on GA.

A GA individual usually represents a program and is encoded in a tree. In this encoding, an individual contains two kinds of nodes, terminals (leaf nodes) and functions (intern nodes). In this study, each GA individual represents a candidate function δ_D which is a similarity combination function and is encoded in a tree structure. Intern nodes contain arithmetic operators. Leaf nodes have similarity values $d_{iI_lI_m}$, where $1 \le i \le K$. Figure 2 shows an example of an individual.

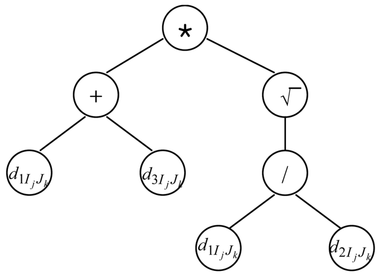

Fig. 2. The example of an individual.

The goal of the proposed fitness computation process is to assign the highest fitness values to the individuals that best encode the user preferences. In this study, the fitness computation is based on the ranking of the database images defined by each individual. Individuals that rank relevant images at the first positions must receive a high fitness value. The fitness of an individual δ_i is computed based on the similarity between the query pattern and all images from the training set. The fitness computation process is divided into three phases. On the first one, M ranked list are computed, each one considering the similarity, according to δ_i among all training set images and each image in the query pattern. On the second phase, these rankings are evaluated. Finally, on the last phase, the final individual fitness is computed. Figure 3 illustrates this process applied to an individual and the training set is defined as a pair where and is a

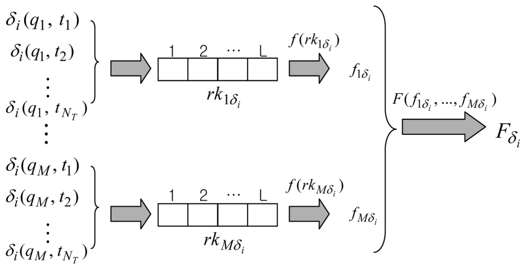

Fig. 3. Fitness function computation of individuals.

function that indicates the user feedback for each image in T.

4. EXPERIMENTS

To show the effectiveness of the proposed method, we use two different measures such as precision vs. recall curve and retrieved relevant images vs. number of iterations curve. The former curve is a common effectiveness evaluation criterion used in information retrieval systems that have been employed to evaluate CBIR systems. The latter curves are used to show the percentage of relevant images retrieved to the user given a number of RF iterations. This curve allows evaluating how the number of retrieved relevant images grows over iterations. In this paper, the proposed framework was compared with other two approaches such as QS_{str} [2], and $SV_{Mactive}$ [6]. The first approach is based on weight assignment, while the last one relies on the use of SVM to classify database images as relevant and non-relevant.

We experimented with 3,000 natural images DB collected from Photoshop and Corel collections and most of them have dimensions of 192×128 pixels. The images can be divided into 14 categories such as such as flower, near, dolphin, zebra, elephant, airplane, horse, lion, polar bear, rose, sunset, tiger, valley and eagle. Experimets were conducted on a 2.5GHz Pentium 4 with 2G RAM. In our experiments, the presence of users is simulated. In this simulation, all images belonging to the same class of the query image are considered relevant. 10 iterations were considered for each query. Experiments also evaluated the effectiveness of the proposed method when 20 and 40 images are showed to the user on each iteration.

The implementation of the proposed framework requires the definition of several GA parameters. Table 1 shows the best values found for each GA parameter. In our experiment, the cross-validation was performed by various population sizes, number of generations, etc. and the parameters of table 1 show the best effectiveness.

Fig. 4 provides the experimental results shown retrieved relevant images vs. number of iterations

Table 1. Best Parameter Values for the proposed framework

Parameter	Value
population size	50
number of generations	10
initial tree depth	2-5
maximum tree depth	15
selection method	Tournament
crossover rate	0.9
mutation rate	0.1
training set size	100
voting selection ratio threshold	0.99

curves and Fig. 5 presents the precision vs. recall curves.

Considering retrieved relevant images vs. number of iterations curves in Fig. 4, the proposed approach presents better results than the other two methods. In fact, SVMactive is close to the proposed approach from the sixth iteration on and QSstr present better results to SVMactive until fourth iteration. The superiority of the proposed approach than the other method was confirmed by the statistical test such as the Wilcoxon test not shown here. For the precision vs. recall curves in Fig. 5, all evaluated methods present similar results until a recall value equal to 0.6. From this point on, the proposed methods start yielding significant better results. This was also confirmed by the Wilcoxon test.

Consequently the proposed framework based on GA presents better results than the other two approaches as shown in Fig. 4 and Fig. 5. The proposed framework shows much better than QSstr and slightly better than SVMactive. Especially, the experiment results of SVMactive method are closer to the proposed frameworks.

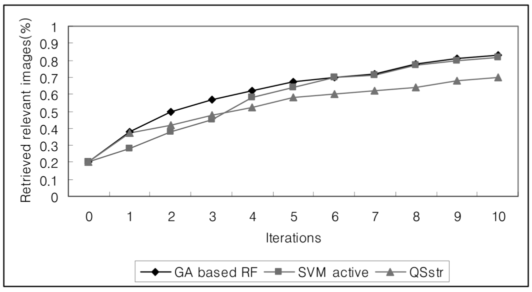

Fig. 4. Experiment results: retrieved relevant images vs. number of iterations curves.

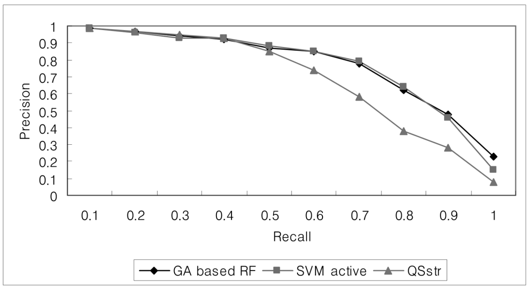

Fig. 5. Experiment results: precision vs. recall curves.

5. CONCLUSION

This paper presented the GA based relevance feedback framework for CBIR. The GA is applied to learn the user preferences using the similarity functions and to find a descriptor combination function that best represents the user perception. Experiments were performed on natural images using color and texture features. In our experiments, the proposed approach method was compared with two other relevance feedback techniques published recently regarding their effectiveness in image retrieval process. Experiment results showed the superiority of the proposed approach and the proposed frameworks are suitable methods for creating CBIR systems. There are, however, research issues which need to be addressed to improve the performance. Semantically similar images are visually very different from each other and human perception of images is subjective. Therefore, extracting semantic contents of the images is always challenging problems in CBIR.

REFERECES

1. Park, S.-S., Seo, K.-K., Jang, D.-S., “Expert system based on artificial neural networks for content-based image retrieval”, Expert Systems with Applications, Vol. 29(3), pp. 589-597, 2005.
2. Doulamis, N., Doulamis, A., “Evaluation of relevance feedback schemes in content-based in retrieval systems”, Signal Processing: Image Communication, Vol. 21(4), pp. 334-357, 2006.
3. Rui, Y., Huang, T., “Optimizing learning in image re-

trieval", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2000.
4. Rui, Y., Huang, T. S., Ortega, M., Mehrotra, S., "Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 8(5), pp. 644-655, 1998.
5. Cord, M., Gosselin, P. H., Philipp-Foliguet, S., "Stochastic exploration and active learning for image retrieval", Image and Vision Computing, Vol. 25(1), pp. 14-23, 2007.
6. Tong, S., Chang, E. Y., "Support vector machine active learning for image retrieval", Proceedings of 9th ACM international conference on Multimedia, 2001.
7. Stejic, Z., Takama, Y., Hirota, K., "Genetic algorithm-based relevance feedback for image retrieval using local similarity patterns", Information Processing and Management, Vol. 39, pp. 1-23, 2003.
8. T. Back, Fogel, D. B., Michalewicz, Z., Evolutionary Computation 1 Algorithms and Operators, Institute of Physics Publishing, 2002.

접수일: 2008년 9월 18일, 심사일: 2008년 10월 6일
게재확정일: 2008년 10월 31일