

# Genetic Algorithm for Weights Assignment in Dissimilarity Function for Trademark Retrieval

David Yuk-Ming Chan and Irwin King

Department of Computer Science and Engineering  
The Chinese University of Hong Kong  
Shatin, N.T., Hong Kong  
{ymchan,king}@cse.cuhk.edu.hk

**Abstract.** Trademark image retrieval is becoming an important application for logo registry, verification, and design. There are two major problems about the current approaches to trademark image retrieval based on shape features. First, researchers often focus on using a single feature, e.g., Fourier descriptors, invariant moments or Zernike moments, without combining them for possible better results. Second, even if they combine the shape features, the weighting factors assigned to the various shape features are often determined with an ad hoc procedure. Hence, we propose to group different shape features together and suggest a technique to determine a suitable weighting factors for different shape features in trademark image retrieval.

In this paper, we use a supervised learning method for finding the weighting factors in the dissimilarity function by integrating five shape features using a genetic algorithm (GA). We tested the learned dissimilarity function using a database of 1360 monochromatic trademarks and the results are promising. The retrieved images by our system agreed well with that obtained by human subjects and the searching time for each query was less than 1 second.

## 1 Introduction

A trademark is a complex symbol that is used to distinguish a company's logo from the others. Up to now, the number of trademarks is over one million and is growing rapidly. Therefore, the task of registering a new trademark becomes difficult without inadvertent infringement of copyright of an existing trademark.

Currently, some researchers use a single shape attribute to represent a trademark. The results are not satisfactory especially for large databases. On the other hand, some suggest to use a combination of features; however, the importance among various features are often difficult to determine. Often this is done in an ad hoc manner. To tackle these shortcomings, we propose to integrate several shape features and suggest a technique to learn the weighting factors of the dissimilarity function for trademark retrieval.

In our system, five shape features are used to capture the contour and the inner parts of a trademark. Fourier descriptors are used to capture the approximated boundary of a trademark. For the inner parts of a trademark, invariant

moments, Euler number, eccentricity, and circularity are used. To integrate the shape features, a supervised learning method using a genetic algorithm is proposed. The weighting factors of the dissimilarity function learned can be used for trademark retrieval afterward. The results show that the weighting factors found could improve the accuracy of trademark retrieval and the retrieved images by our system agreed well with human perception.

The paper is organized as follows. In Section 2, previous work on the content-based image retrieval is reviewed. The problem is defined in Section 3. The genetic algorithm for finding the weighting factors is introduced in Section 4. Section 5 describes our trademark retrieval model and the features used. The experimental results are given in Section 6 and Section 7 concludes our report.

## 2 Literature Review

There are various shape representation, matching and similarity measuring methods proposed in the past few years. The QBIC (Query By Image Content) system [1] serves as an image database filter which allows queries of large image databases based on visual image content such as color percentages, color layout, and textures occurring in the images. Although user can use multiple cues as a query, one needs to be well trained before using the QBIC system effectively. STAR (System for Trademark Archival and Retrieval) system [12, 16] uses features based on R, G, and B color components, invariant moments, and Fourier descriptors extracted from manually isolated objects. Kim *et. al.* [11] developed a trademark retrieval system which uses Zernike or pseudo-Zernike moments of an image as a feature set. The retrieval scheme is based on visually salient feature that dominantly affects the global shape of the trademark. A two-stage approach for trademark retrieval is proposed by Jain *et. al.* [10]. In the first stage, easily computable features like edge angles and moment invariants are used while in the second stage, the plausible retrievals from the first stage are screened using a deformable template matching process. They proposed to integrate the associated dissimilarity values of two shape-based retrievals; however, they could not provide an effective way for assigning weights for different shape features. Logo similarity matching based on positive and negative shape features was suggested by Soffer *et. al.* [15]. The proposed methods seem to be accurate but the testing database size is too small (130 logos) and for each insertion, the representative component of a logo must be selected by the user.

## 3 Problem Definition

The problem of finding weights for the dissimilarity function for trademark retrieval can be formalized as follows:

**Definition 1.** An image database  $DB$  is defined as

$$DB = \{I_i\}_{i=1}^n, \quad (1)$$

where  $I_i$  is an image in the database.

**Definition 2.** Given an image  $I$  and a set of feature parameters  $\theta = \{\theta_i\}_{i=1}^n$ , a feature extraction function  $f$  is defined as

$$f : I \times \theta \rightarrow \mathcal{R}^d , \quad (2)$$

which extracts a real-valued  $d$ -dimensional feature vector.

**Definition 3.** Let  $x_f^I$  be a feature vector of an image  $I$  on the basis of a feature extraction function  $f$ . Then, the integrated dissimilarity function,  $D_t$ , between two images  $I_1$  and  $I_2$  is defined as

$$D_t(I_1, I_2) = \frac{\sum_{i=1}^n w_i D_{f_i}}{\sum_{i=1}^n w_i} , \quad (3)$$

where  $f_i$  is a feature extraction function,  $D_{f_i}$  is the Euclidean distance between the feature vector  $x_{f_i}^{I_1}$  and feature vector  $x_{f_i}^{I_2}$ ,  $w_i$  is the weight assigned to feature vector set  $i$  and  $n$  is the number of the feature vector sets.

Since the use of a single shape attribute for retrieval may not have enough discriminatory information, in order to increase the accuracy of retrievals, the integration of attributes was suggested [9, 10, 16].

**Definition 4.** Given a training database  $db \subset DB$ , where  $DB$  is an image database. A training pair  $TP$  is defined as

$$TP = (I_T, I_S) , \quad (4)$$

where  $I_T \in db$  is the target image for a query and  $I_S \in db$  is the user defined best matched image.

**Definition 5.** Given an integrated dissimilarity function  $D_t$ ,  $n$  training pairs for a given training database  $db$ , the total count  $TC(\mathbf{w})$  is defined as the number of correct hits given by  $D_t$  with the set of weights  $\mathbf{w}$  for searching in  $db$ . For example, given a training pair  $TP = (I_T, I_S)$ , if  $I_S$  is selected by the integrated dissimilarity function  $D_t$ , then this is a correct hit.

**Definition 6.** The set of weights  $\mathbf{w}$  in a dissimilarity function  $D_t$  is defined as  $\mathbf{w} = \{w_i\}_{i=1}^n$ , where  $w_i$  is the weight assigned to feature vector set  $i$ .

The problem is to find  $\mathbf{w}$  such that  $TC(\mathbf{w})$  is maximized, i.e.,

$$\arg \max_{\mathbf{w}} TC(\mathbf{w}) . \quad (5)$$

## 4 Finding Weights Using a Genetic Algorithm

A genetic algorithm (GA) is an optimization method based on the evolutionary metaphor. It has been shown to outperform both gradient methods and random search in solving the optimization problems with few cost function evaluations [2]. A short introduction of genetic algorithm can be found in [14]. Our problem defined in Section 3 is an optimization problem; therefore, we suggest to use a genetic algorithm to solve it. The details of our GA are as follows.

## 4.1 Chromosome Representation

A chromosome representation is used to describe an individual in the population of interest. A chromosome in our GA is defined as

$$c = (w_1, w_2, \dots, w_i, \dots, w_n), \quad (6)$$

where  $w_i$  is the weight assigned to feature vector set  $i$  and  $n$  is the number of feature vector sets which is the same as the number of genes in the chromosome. Note that a population  $P$  is defined as  $P = \{c_1, c_2, \dots, c_i, \dots, c_{PopSize}\}$ , where  $PopSize$  is the number of individuals in the population and  $c_i$  is a chromosome.

## 4.2 Selection Function

A selection function plays a vital role in a genetic algorithm because it selects individuals to reproduce successive generations. A probabilistic selection is performed based on the individual's fitness such that the better individuals have an increased chance of being selected.

The selection method, Roulette wheel, proposed by Holland [5] was used in our implementation. The probability,  $P_i$ , for each individual is defined by

$$P[\text{Individual } i \text{ is chosen}] = \frac{F_i}{\sum_{j=1}^{PopSize} F_j}, \quad (7)$$

where  $F_i$  is equal to the fitness of individual  $i$ . A series of  $N$  random numbers is generated and compared against the cumulative probability,  $C_i = \sum_{j=1}^i P_j$ , of the population. If  $C_{i-1} < U(0, 1) \leq C_i$ , the individual is selected.

## 4.3 Genetic Operators

Genetic operators provide the basic searching mechanism of the GA. The operators are used to create new solutions based on existing solutions in the population. The operators including *arithmetic crossover*, *heuristic crossover*, *simple crossover*, *boundary mutation*, *multi-non-uniform mutation*, *non-uniform mutation* and *uniform mutation* are used in our implementation. For more details, please refer to [6].

## 4.4 Initialization, Termination and Evaluation Function

The initial population is randomly generated. The stopping criterion is a predefined maximum number of generations and the evaluation function is the total count  $TC(\mathbf{w})$  defined in Definition 5.

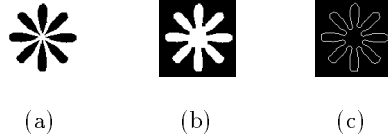


Fig. 1. (a) Original image. (b) Closed image. (c) Approximated boundary extracted.

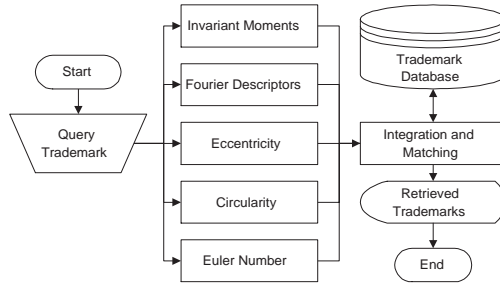


Fig. 2. The proposed shape-based trademark retrieval system.

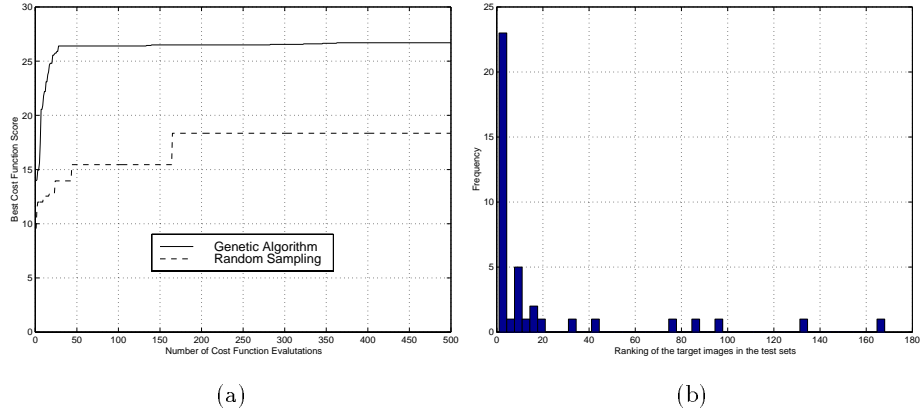
## 5 Image Retrieval model

In our system, five features that are scale, rotational and translation invariant are chosen to represent a trademark (see Fig. 2). They are Fourier descriptors [3, 13] of approximated boundary, seven invariant moments [7], eccentricity [8], circularity [8] and Euler number [4]. Since there can be more than one component in a trademark, the image will be connected by applying a closing operator [3] if this is the case when calculating the Fourier descriptors and circularity. Fig. 1 shows an example of approximated boundary extraction of a trademark. The extracted feature vectors of the images are stored in the database. When a user raises a query, the features of the query trademark are first extracted. Then the extracted features are matched linearly with the features in the database and then intergrated by the dissimilarity function (see Definition 3). The trademarks are then displayed in the order of similarity.

## 6 Experimental Results

### 6.1 Genetic Algorithm

The proposed genetic algorithm was tested with the following setup. The population size,  $PopSize$ , was 30 and the maximum number of iterations was set to 500. In addition, there were 40 training pairs ( $TP$ ) and the size of the training database,  $db$ , was 200. The values of the weights (genes) were bounded by 0



**Fig. 3.** (a) Best cost function value versus the number of cost function evaluations for random sampling and GA. (b) The ranking histogram.

and 1. The probability of application of crossover was 0.6 and the probability of application of mutation was 0.05. The experiments were repeated 10 times with different random seeds.

Note that partial credit is given to the imperfect match in the evaluation function of the GA. For example, if the target is ranked in the first position,  $TC(\mathbf{w})$  is increased by one. If it is ranked in the second position,  $TC(\mathbf{w})$  is increased by 0.95. If it is ranked in the third position,  $TC(\mathbf{w})$  is increased by 0.9 and so on. This method helps the GA to search for a better solution in a smaller number of iterations.

From Fig. 3(a), the genetic algorithm improved the accuracy of the retrievals by changing the weights. It converged in 500 iterations with a total count of 26.7 which means on average, the distance function ranks the target of a query in a training pair to the eighth position. The performance of the final distance function is further analyzed by the ranking histogram in Fig. 3(b). From the histogram, 70% of the targets were ranked at the top ten positions and 83% of the targets were ranked at the top twenty positions. However, there were several targets ranked very low. This is because trademarks that appear to be perceptually similar need not be similar in their shape. As some examples, Fig. 4 shows several pairs of images which may perceptually look similar, but have low image correlation. However, we believe that by using more shape features, the accuracy can be improved.

We also tested the dissimilarity function with equal weights ( $w_1 = w_2 = w_3 = w_4 = w_5 = 0.2$ ) and the total count,  $TC(\mathbf{w})$ , was 14.3. This is because some shape features match the human perception better than the others and therefore higher weights should be assigned to them.

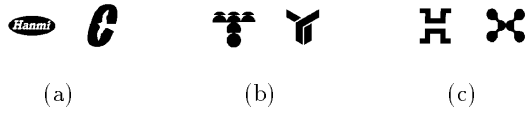


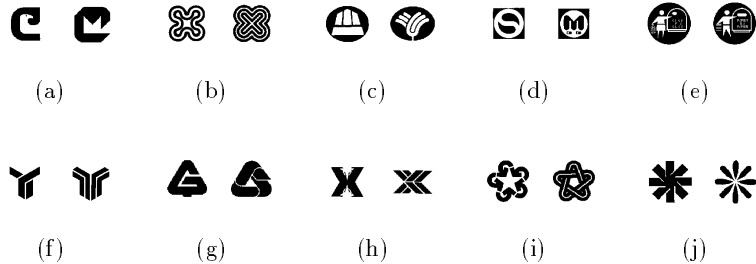
Fig. 4. Perceptually similar images that have low image correlation.

Query trademark	Similar trademarks retrieved								

Fig. 5. Results of the similar trademarks retrieval. The first column are the query trademarks and the others are the retrieved similar trademarks.

### 6.2 Trademark Retrieval

After the GA training has terminated, the chromosome with the highest  $TC(w)$  is selected as the final weights for retrieval. The weights found were tested with a database of 1360 trademarks on an Ultra 5 Machine. The system was developed using Matlab 5.2 under the Unix operating system. To verify the performance of the proposed method, a set of trademarks were used as queries to the trademark database. Several sample query results are shown in Fig. 5. From the results obtained, the trademarks retrieved agreed well with human perception. In addition, the average time of 10 trials for feature extraction and database querying were 2.96s and 0.08s respectively.



**Fig. 6.** Query trademarks (LHS) and the best matches (RHS) retrieved by the volunteers.

Query Nature	$n = 1$ (%)	$n \leq 2$ (%)	$n \leq 3$ (%)	$n \leq 4$ (%)	$n \leq 5$ (%)	$n \leq 20$ (%)	Not retrieved (%)
Normal Query	20	30	50	70	70	100	0
Rotated Query	10	20	40	70	70	100	0
Scaled Query	20	20	50	60	60	100	0

**Table 1.** Retrieval results on the basis of the dissimilarity function with the weights found by the GA:  $n$  refers to the position of the correct retrieval. The last column indicates the percentage of non-retrieved images in the top 20 matches.

Since shape similarity is a subjective issue, in order to further evaluate the proposed method, five volunteers were asked to perform similarity retrieval based on shape on the database of 1360 images.

Given a query, they were asked to choose the best match from the database. Fig. 6 shows the query trademarks and the best matches retrieved by the volunteers and Table 1 presents the results of retrieval of our system, where  $n$  corresponds to the position of the correct retrieval. As it is observed from the results, our simple shape measure was effective in retrieving rotated and scaled images. This is because the shape features that we chose are all rotation and scaling invariant. Moreover, all the target images were ranked in the top twenty positions. This reveals that the dissimilarity function models the human perception quite well.

## 7 Conclusion and Future Work

In this paper, we suggested to integrate different trademark features by using a dissimilarity function for retrieving similar trademarks. A method for finding the weighting factors of the difference function using a GA has been proposed. The results show that the weighting factors found by the GA improves the accuracy of



trademark retrieval. Future research will consider to use more shape features like Zernike moments or edge angles to further improve the accuracy of retrieval. The weights of the distance function can be found by the proposed GA easily. Besides, this method can be applied to optimize other general dissimilarity measures (other than a sum of weight Euclidean). This can be easily done by changing the distance function to be optimized in the GA and using the parameters of the distance function as genes in the chromosome representation of the GA.

## References

1. M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by Image and Video Content: The QBIC System. *Computer*, 28(9):23–32, September 1995.
2. D. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
3. R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley, 1992.
4. R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*, volume 2. Addison-Wesley, 1993.
5. J. Holland. *Adaptation in natural and artificial systems*. The University of Michigan Press, 1975.
6. C. Houck, J. Joines, and M. Kay. A genetic algorithm for function optimization: A matlab implementation. *NCSU-IE*, 95-09, 1995.
7. M. K. Hu. Visual Pattern Recognition by Moment Invariants. *IRE Trans. on Information Theory*, 8, 1962.
8. B. Jahne. *Digital Image Processing: Concepts, Algorithms and Scientific Applications*. Springer-Verlag, Berlin; New York, 4 edition, 1997.
9. A. K. Jain and A. Vailaya. Image Retrieval using Color and Shape. *Pattern Recognition*, 29(8):1233–1244, 1996.
10. A. K. Jain and A. Vailaya. Shape-Based Retrieval: A Case Study with Trademark Image Databases. *Pattern Recognition*, 31(9):1369–1390, 1998.
11. Y. S. Kim and W. Y. Kim. Content-Based Trademark Retrieval System Using Visually Salient feature. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 307–312, 1997.
12. C. P. Lam, J. K. Wu, and B. Mehtre. STAR - A System for Trademark Archival and Retrieval. In *2nd Asian Conf. on Computer Vision*, volume 3, pages 214–217, 1995.
13. E. Persoon and K. S. Fu. Shape discrimination using Fourier descriptors. *IEEE Trans. on Systems, Man Cybernetics*, 7(2):170–179, 77.
14. G. Roth and M. D. Levine. Geometric Primitive Extraction Using a Genetic Algorithm. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(9), Sep 1994.
15. A. Soffer and H. Samet. Using Negative Shape Features for Logo Similarity Matching. In *14th International Conf. on Pattern Recognition*, volume 1, pages 571–573, 1998.
16. J. K. Wu, B. M. Mehtre, Y. J. Gao, C. P. Lam, and A. D. Narasimhalu. STAR—A Multimedia Database System For Trademark Registration. In Witold Litwin and Tore Risch, editors, *Applications of Databases, First International Conference*, volume 819 of *Lecture Notes in Computer Science*, pages 109–122, Vadstena, Sweden, 21–23 June 1994. Springer.

This article was processed using the L<sup>A</sup>T<sub>E</sub>X macro package with LLNCS style