



Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: [www.elsevier.com/locate/asoc](http://www.elsevier.com/locate/asoc)



## Distance-based relevance feedback using a hybrid interactive genetic algorithm for image retrieval

Miguel Arevalillo-Herráez\*, Francesc J. Ferri, Salvador Moreno-Picot

Department of Computer Science, University of Valencia, Avda. Vicente Andrés Estellés 1, 46100-Burjassot, Spain

### ARTICLE INFO

#### Article history:

Received 2 July 2009

Received in revised form 18 May 2010

Accepted 30 May 2010

Available online xxx

#### Keywords:

CBIR

Image retrieval

Relevance feedback

Evolutionary computation

Genetic algorithms

### ABSTRACT

Content-based image retrieval (CBIR) systems aim to provide a means to find pictures in large repositories without using any other information except the own content of the images, which is usually represented as a feature vector extracted from low-level descriptors. This paper describes a CBIR algorithm which combines relevance feedback, evolutionary computation concepts and distance-based learning in an attempt to reduce the existing gap between the high level semantic content of the images and the information provided by their low-level descriptors. In particular, a framework which is independent from the particular features used is presented. The effect of different crossover strategies and mutation rates is evaluated, and the performance of the technique is compared to that of other existing algorithms, obtaining considerably better and very promising results.

© 2010 Elsevier B.V. All rights reserved.

### 1. Introduction

Information retrieval (IR) deals with finding digital resources in large databases. Content-based image retrieval (CBIR) systems are a special type of IR system where the elements in the repository are pictures. It is common that a CBIR system represents a picture by a collection of visual descriptors related to different aspects of image appearance such as color, texture or shape. These features describe the image and are used in the retrieval process to judge (subjective) picture similarity. Current CBIR systems make use of relevance feedback techniques in order to reduce the so-called semantic gap that usually exists between image contents as perceived by the users and their corresponding (multiple) representations. When relevance feedback is used, the user is considered an active part of an iterative process. At each step, the user selects the best matches from a series of pictures which are displayed on a graphical user interface (GUI). Then, the selection is processed and a new set of images are presented again on the GUI, repeating the process until a satisfactory group of pictures is retrieved. The system is then able to progressively retrieve better images according to the available information about what the user is looking for and how he/she judges its relevance.

Evolutionary algorithms have been used to solve a wide diversity of problems in the machine learning area. An evolutionary

algorithm attempts to solve a problem applying Darwin's basic principles of evolution [1] on a population of trial solutions to a problem, called individuals. Genetic algorithms (GAs) are a class of evolutionary algorithms, which place a special emphasis on the application of genetic operators, such as mutation and crossover. GAs use an encoding method to represent potential solutions to the problem, and a fitness measure that allows a quantitative evaluation of each candidate according to how close it is to an ideal optimal solution. Starting from an initial population, each candidate is evaluated using the fitness function and the most promising individuals are allowed to reproduce and determine the next generation of individuals. The population evolves according to a series of pre-established rules and the process is repeated until a convenient solution is found or some convergence criteria are met [2].

The retrieval can be conveniently modeled as an optimization problem. If we consider that the user has a picture in mind, the goal of the retrieval system consists of obtaining the fittest pictures from the set of potential solutions to the problem (the images in the repository). Obviously, the corresponding fitness function would need to be related to the subjective or semantic similarity between images in the database and users thoughts. Unfortunately, we lack such an adequate fitness function. Although similarity values between two images may be computed by comparing their feature vectors, the only truly reliable judgments are those produced by a user. Interactive genetic algorithms (IGAs) use human evaluation to overcome this problem. In an IGA, the user directly or indirectly assigns the fitness to the individuals. IGAs have been applied to many different research areas including CBIR [3]. In [4],

\* Corresponding author. Tel.: +34 96 354 39 62.

E-mail addresses: [miguel.arevalillo@uv.es](mailto:miguel.arevalillo@uv.es) (M. Arevalillo-Herráez), [francesc.ferri@uv.es](mailto:francesc.ferri@uv.es) (F.J. Ferri), [salvador.moreno@uv.es](mailto:salvador.moreno@uv.es) (S. Moreno-Picot).

an excellent review is presented including applications in a large number of research fields.

This paper presents an algorithm which combines interactive evolutionary computation concepts with distance-based strategies to produce an efficient relevance feedback algorithm for image retrieval. In particular, an adapted extension of a previously introduced nearest neighbor (NN) approach including feature weighting and local search is plugged into a real-coded IGA in order to produce meaningful relevance scores without requiring a large number of feedback iterations as if a pure genetic approach was used.

The remainder of the paper is organized as follows. First, the state-of-the-art about relevance feedback mechanisms and related work using evolutionary computation techniques in this context are described in Section 2. A convenient formalization of the underlying problem and some general considerations are given in Section 3. The proposed approach is deeply explained in Section 4. Exhaustive experimentation covering both tuning of the methods and comparative evaluation is described in Section 5. Finally, the paper ends with Section 6 in which some conclusions are drawn and further work is briefly explained.

## 2. State-of-the-art

### 2.1. Related work in relevance feedback in CBIR

In recent years, several strategies have been developed to guide the search process in image retrieval systems. Although the space limitations do not make it possible to provide a complete review of all these techniques, we attempt to offer a simple classification of the most relevant ones and describe each group briefly. Indeed, there exist many other approaches and algorithms which have contributed to the advancement of relevance feedback and are not mentioned in this section. For a more comprehensive review, the reader is referred to e.g. [5,6]:

- Classical approaches. The first approaches to relevance feedback were inspired by techniques which were typically used in the context of textual information retrieval. These were based on adapting the similarity measure and moving the (unique) query point so that it appears closer to the relevant results and far from those which are non-relevant. Good examples of this group of technique are e.g. the MindReader system described in [7], and the algorithms proposed in [8,9]. More recently, several efficient query point movement methods have been proposed in [10], proving that their approach is able to reach any given target image with fewer iterations in the worst and average cases. Although this type of mechanisms embraces some of the fastest approaches to relevance feedback, they present two main disadvantages. On the one hand, dependencies between image features are usually ignored [11]. On the other hand, feature weighting techniques treat the feature spaces globally rather than locally [12].
- Probabilistic techniques. This group consists of a large number of algorithms, most of them based on Bayesian frameworks. A representative example is the method used in the PicHunter system, proposed by Cox et al. in [13]. In this system, user preferences are modeled to lead to a probability distribution on the event space (databases and set of all possible history sequences). Both memory requirements and execution time scale linearly with the number of images in the database. Another similar approach was presented in [14], based on the modeling of user preferences as a probability distribution on the image space. This distribution is the prior distribution and its parameters are modified based on the information provided by the user. This yields the posterior from which the predictive distribution is calculated and used to show to the user a new set of images until he/she is satisfied or

the target image has been found. Other algorithms that belong to this approach include [15–17].

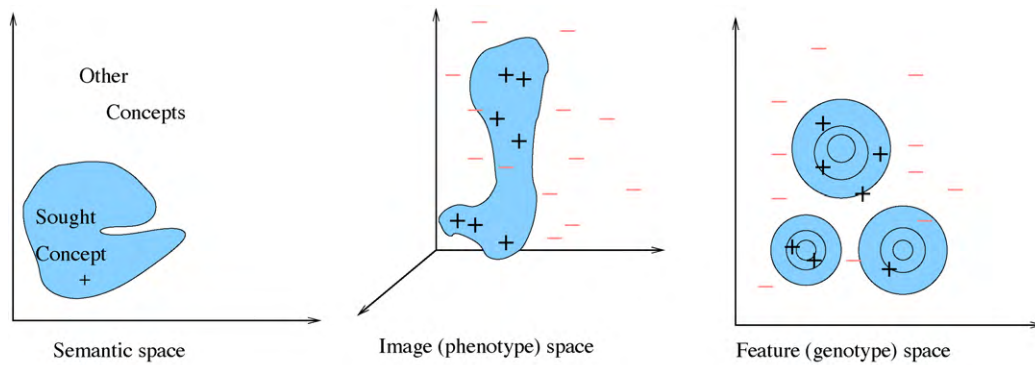
- Supervised learning techniques. A current trend in the design of relevance feedback mechanism is the use of support vector machines (SVMs) [18]. These are a set of supervised learning methods for classification which attempt to find the hyperplane that maximizes the separation between two classes. In a CBIR context, pictures which are close to this hyperplane are considered the most informative, while those localized the farthest from it are regarded as the most/least relevant. Both, methods which rely on 2-class SVMs and require positive and negative selections [19], and others which learn from positive selections only [20–22] have been proposed. The major problem with these approaches is the fine-tuning of the retrieval systems and choosing the optimal set of parameters for the SVM. In [23], an approach based on the NN paradigm is proposed. Each image is ranked according to a relevance score depending on nearest-neighbor distances. This approach is proposed both in low-level feature spaces, and in “dissimilarity spaces”, where images are represented in terms of their dissimilarities from the set of relevant images.
- Unsupervised learning techniques. In [12], a separate self-organizing map (SOM) is built for a number of different content descriptors, and the images in the database are connected to their map units. The units in each SOM are awarded positive and negative scores according to the user selections, and these are passed onto the images contained in the units, to produce a global score for each picture. These scores are then used to yield a new image ranking. In [24], a cluster-based unsupervised learning strategy is used to improve the search performance by fully exploiting the similarity information. Instead of retrieving a set of ordered images, it retrieves groups of pictures by applying a graph-theoretic clustering algorithm to a collection of images in the vicinity of the query.
- Other soft computing techniques. A regression model has been recently proposed in [11]. This algorithm considers the probability of an image belonging to the set of those sought by the user, and models the logit of this probability as the output of a linear model whose inputs are the low-level image features. In [25], the authors propose a technique that defines a fuzzy set so that the degree of membership of each image in the repository to this fuzzy set is related to the user's interest in that image. Then, positive and negative selections are used to determine the degree of membership of each picture to this set. The system attempts to capture the meaning of a selection by modifying a series of parameters at each iteration, according to the user's behavior, becoming more selective as the search progresses.

### 2.2. Related work in evolutionary computation

The use of evolutionary computation in IR is not new. In [26], applications in the following eight problems in the area of IR are reviewed: (1) automatic document indexing, (2) document and term clustering, (3) query definition, (4) matching function learning, (5) image retrieval, (6) design of user profiles for IR on the Internet, (7) web page classification, and (8) design of agents for Internet searching.

In the context of CBIR, evolutionary computation has also been applied to a number of different problems, namely:

- Combining descriptors. Integrating the scores obtained from different descriptors is a major concern in image retrieval. In [27], a framework which uses genetic programming (GP) to yield an optimal combination function is presented. The algorithm evaluates different combinations of addition, multiplication, division and square root operations on the original set of distance functions, in an attempt to find the most adequate compound



**Fig. 1.** Example illustrating a particular semantic concept in an abstract semantic space, a discrete image space (i.e. the image repository) and a given feature space. Images in the repository can be either relevant (plus sign) or non-relevant (minus sign). In the feature space concentric circles mark the location of the ideal query points in  $\mathcal{C}$  that are supposed to approximate the concept in this space. The probability of finding a relevant result decreases as the distance from the nearest element in  $\mathcal{C}$  increases.

distance function. In [28], the composite function is restricted to a linear combination of single feature space metrics, and a multi-objective optimization technique is used to learn the weighting factors from a group of representative salient blocks. In [29], a relevance feedback mechanism which uses a GA to re-calculate feature weights at each iteration is presented. Finally, in [30], the author proposes a technique which is able to combine multiple representation schemes using a multi-objective GA. The system focuses on a structural similarity framework that addresses topological, directional and distance relations among image objects. After a structural query has been formalized as a multi-criteria query, the multi-objective optimization algorithm is used to retrieve a diverse set of solutions in one run.

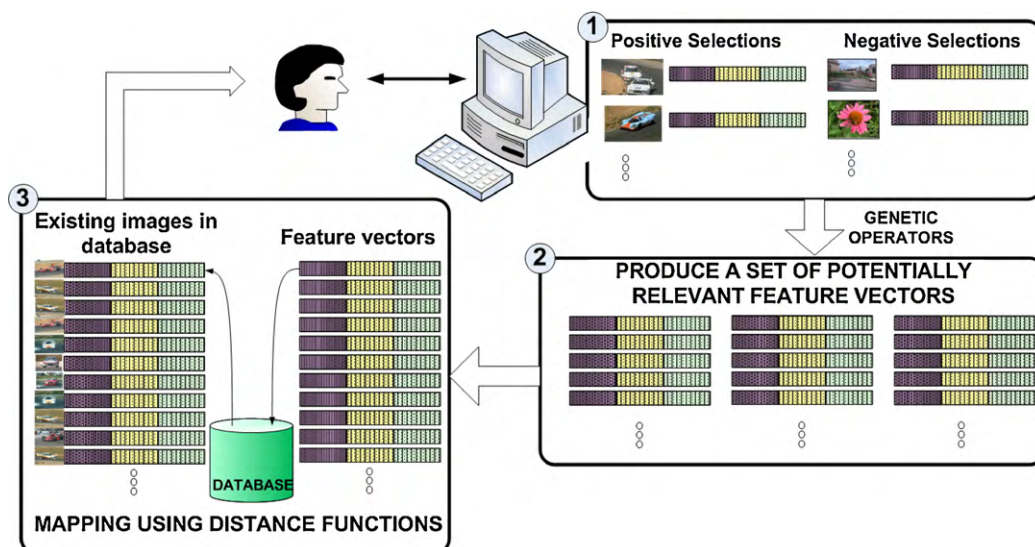
- Image indexing. Recently, a novel evolutionary method called the evolutionary group algorithm (EGA) has been proposed in [31]. This method is used for complex and time-consuming optimization problems such as finding optimal parameters for content-based image indexing algorithms. In this algorithm, the image database is partitioned into several smaller subsets, and each subset is used by an updating process as training patterns for each chromosome during evolution.
- Relevance feedback. In [32], the authors combine a learning method based on a SVM (support vector machine) with an IGA. At each iteration, the user's selections are processed by the SVM and a non-linear classification of the images is performed. The

learning results are then used to assign scores to each image in the repository, and some with the highest scores are added to the population. This increases the number of good individuals at each generation. In [3], the image features are the sign of the wavelet coefficients obtained by decomposing the images using a wavelet transform. At each iteration the user has to assign the fitness to the 12 images shown. Then, the system creates a new population for the next generation using a GA which does not use mutation and displays the most similar images.

The technique that is presented in the current paper is also based on evolutionary computation, and shares with [32,3] the underlying principle of using an IGA. However, the application of evolutionary computation principles is combined with a distance-based approach which extends the NN approach to relevance feedback [23]. This hybridization significantly speeds up the convergence and improves the retrieval results obtained. Contrary to some previous works, we do not restrict the approach neither to a particular set of features nor to a certain type of repository.

### 3. Relevance feedback model under a genetic framework

Let us assume we have a repository of images,  $\mathcal{X}$ , composed of  $m$  individuals  $x_i$ ,  $\mathcal{X} = \{x_i\}_{i=1}^m$ , and that for each of them a vector of  $n$  features,  $\mathbf{x}_i \in \mathbb{R}^n$ , is available.  $n$  is the total dimension of the fea-



**Fig. 2.** A simplistic view of the technique.



ture space that embraces  $t$  different descriptors corresponding to each of the multiple representations (color, texture, etc.) which are available. In genetic terms, each individual  $x_i$  would play the role of a phenotype, and its feature representation,  $\mathbf{x}_i$ , of the genotype. Let us also assume that a particular (and convenient) distance measure,  $d_k$ ,  $k = 1, \dots, t$  is given for each descriptor.

A particular user is interested in retrieving images from  $\mathcal{X}$  related to a particular semantic concept. This concept is an abstract property that one or several images in the database may possess or may fulfill to a different degree. If we could allow the user to see and judge all images in  $\mathcal{X}$  either as relevant or non-relevant with regard to this particular concept, this would give rise to a partition of  $\mathcal{X}$ . When images are represented in the feature space, we hope that relevant images get confined into particular and smooth regions in this space. For example, (single) query point movement techniques [9] rely on the assumption that this region in the feature space can be approximately represented as an hyperspherical or hyperellipsoidal region around an ideal query point. In this work, a setting is adopted in which a set of ideal query points in the feature space is assumed in a similar way as in [33]. This set of ideal query points will be denoted as  $\mathcal{C}$ . It will be implicitly assumed that the distances  $d_k$  allow one to estimate the relative similarity of the feature vectors of any two images, so that the probability of finding a relevant result decreases as the distance from the nearest element in  $\mathcal{C}$  increases (see Fig. 1).

It will be also assumed that the relationship between the genotype and the phenotype is strongly causal (small variations on the genotype space imply small variations on the phenotype space [34]), preserving the neighborhood structure of the mapping (the neighbors of the genotype are also neighbors in the phenotype space). Under this problem formalization, the search can be modeled as an attempt to find the phenotypes  $x_i$  whose genotypes  $\mathbf{x}_i$  are in the close neighborhood of any of the feature vectors in  $\mathcal{C}$  according to an appropriate distance measure. This requires an efficient method to infer the members in  $\mathcal{C}$  and a method to conveniently define distances between the elements in  $\mathcal{X}$  and the elements in  $\mathcal{C}$  from the given distances  $d_k$ . The assumption of strong causality allows us to let the user subjectively evaluate distances from any element in  $\mathcal{X}$  to its closest element in  $\mathcal{C}$  in the phenotype space (by looking at the images), and transfer the results to the genotype (feature) space.

As in most approaches in the literature, relevance feedback processing will be based on representing user feedback on a set of images,  $\mathcal{Q} \subset \mathcal{X}$ , that will be divided into a subset  $\mathcal{Q}^+$  of relevant samples and another subset  $\mathcal{Q}^-$  containing the non-relevant ones, by using some form of user interaction. A simple interpretation of this information is that, from the set  $\mathcal{Q}$ , the members of  $\mathcal{Q}^+$  have a higher probability to appear close to some element in  $\mathcal{C}$ .

The purpose of relevance feedback can be established within this context as to process the subsets  $\mathcal{Q}^+$  and  $\mathcal{Q}^-$  to define a new set of potential points of interest where the likelihood of relevance is maximized. Under a pure genetic framework, a population of genotypes would be evolved aiming at converging to an appropriate approximation of the set of ideal points in  $\mathcal{C}$ . The only available information to this end, is the user judgment on the images in  $\mathcal{Q}$ . One intrinsic problem of this approach is that ideal query points living only in the genotype space cannot be directly judged.

#### 4. Proposed evolutionary relevance feedback mechanism

The general idea of the proposed approach consists of considering a population of  $p$  individuals (images) which will be obtained from  $\mathcal{Q}^+$  by applying genetic operators. Instead of evolving a set of query points and devise a ranking strategy to obtain scores, the same population resulting from genetic operators will be shown to

the user after an appropriate genotype–phenotype mapping. On the one hand, this will lead to a dramatic speed-up with regard to a pure genetic approach. On the other hand, it will give rise to the benefits of distance-based scoring [35] but improved by randomized ranking computation.

The proposed approach can be summarized graphically as in Fig. 2. The user makes a judgment about the images being shown at a particular iteration and the sets of positive (relevant) and negative (non-relevant) selections are updated. Then, genetic operators are applied to distinct pairs of genotypes chosen randomly from the set  $\mathcal{Q}^+$  of positive ones in order to obtain a set of  $p$  potentially relevant feature vectors (genotypes). Finally, each of these feature vectors is mapped to an image in the database ( $\mathcal{X}$ ) and shown to the user again.

The notation used and more specific details are introduced along with the following algorithm that expresses the above ideas in a more systematic way.

$\mathcal{X}$ : database of images.  $\mathcal{Q}^+$ ,  $\mathcal{Q}^-$ : relevant and non-relevant images, respectively.  $\mathcal{S}$ : images shown to the user at each iteration,  $|\mathcal{S}| = p$ .  
Initially,  $\mathcal{S}$  contains  $p$  arbitrary images from  $\mathcal{X}$ .  
Repeat:  
1. Show the images in  $\mathcal{S}$  to the user.  
2. Update  $\mathcal{Q}^+$  and  $\mathcal{Q}^-$  according to user selection.  
3. Evolve: obtain a set of  $p$  offsprings  $\mathcal{O}$  from  $\mathcal{Q}^+$  using genetic operators.  
4. Mapping: produce a new set of  $p$  images  $\mathcal{S} \subset \mathcal{X}$  (one from each offspring in  $\mathcal{O}$ ).

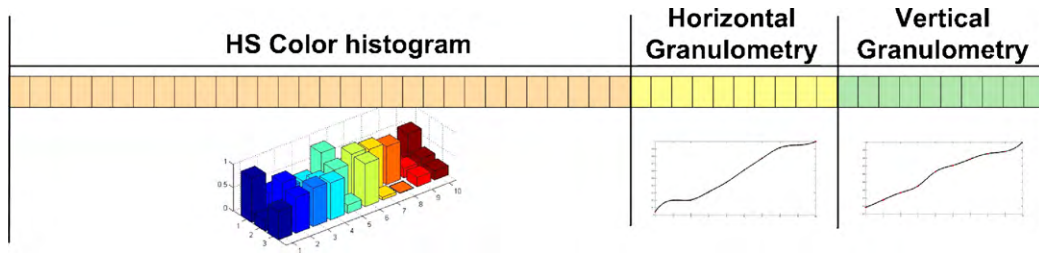
A particular form of elitism has been incorporated in the algorithm by including all positive selections,  $\mathcal{Q}^+$ , in the next generation. This means that only  $p - |\mathcal{Q}^+|$  individuals are generated through genetic operators to complete the set  $\mathcal{O}$  at step 3.

Note that the offspring set  $\mathcal{O}$  is defined in the genotype space and thus consists of  $p$  feature vectors  $\mathbf{o}_i = \{o_{i,1}, \dots, o_{i,n}\} \in \mathbb{R}^n$ . After the mapping, the resulting individuals will constitute a new set of potential parents  $\mathbf{p}_j = \{p_{j,1}, \dots, p_{j,n}\}$  (the feature vectors of the closest elements in  $\mathcal{X}$ ). An example of a typical feature vector (a chromosome) is illustrated in Fig. 3. This corresponds to the case of the small collection used in Section 5.

The goal of this strategy is to be able to explore both the phenotype and genotype spaces at the same time in a convenient way. Genetic operators at step 3 of the above algorithm provide a way of extending the searching region (in the genotype space). The mapping in step 4, aims to obtaining convenient individuals in the phenotype space by taking into account both the generated genotypes and several distance based criteria. This twofold strategy will result an effective way of progressively retrieving more relevant images at each iteration. Although distance-based relevance feedback criteria have already been studied [23], several modifications and extensions will be included in order to exploit the benefits of this proposed hybridization. Besides, several different genetic operators and their corresponding parametrization will be studied in the context of the approach proposed.

##### 4.1. Genetic operators

The chromosomes used in the proposed method are vectors of floating point numbers (image descriptors in general). This is a typical case of real coding, which makes it possible to use large domains for the variables, exploit the principle of strong causality and avoid the encoding and decoding operations required in binary coded problems [37]. In real-coded GA, crossover has been considered the fundamental operator, and a lot of research has been focused



**Fig. 3.** An example of a typical chromosome composed of a total of 50 values. The first 30 contain the  $10 \times 3$  HS color histogram of the picture, normalized so that their sum is 1. The last 20 characterize two granulometry cumulative distributions [36] by using a spline basis (10 values using a horizontal segment as the structuring element and the other 10 using a vertical one). These are all defined in the interval  $[-2, 2]$ .

on developing this type of operator to improve the performance of the algorithms they support. As a result a large number of different strategies have been proposed. For a good description of the most relevant ones, the reader is referred to e.g. [38–40,37].

According to the taxonomy presented in [40], crossover operators can be divided into three major categories, namely:

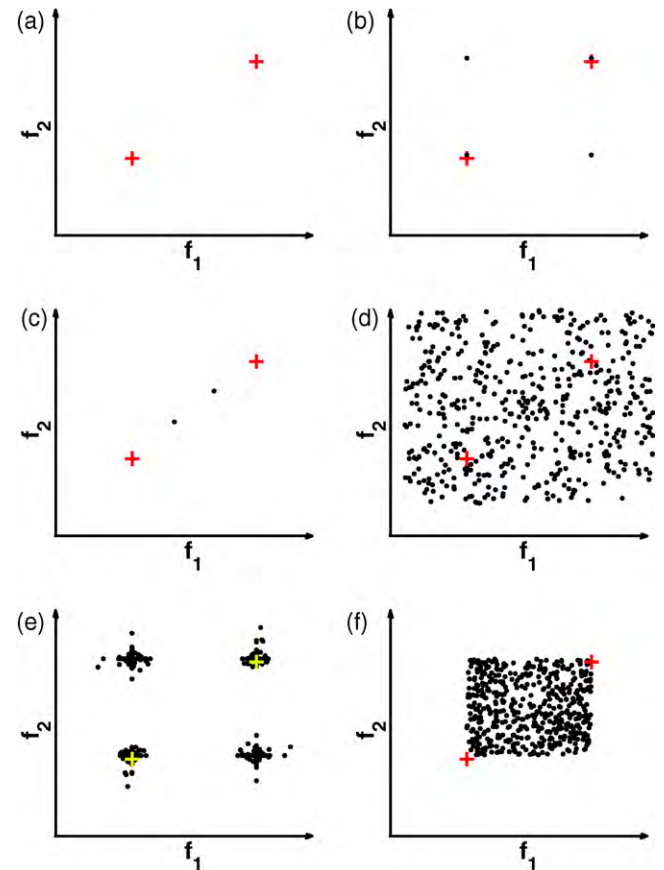
1. Discrete crossover operators. These are operators which were initially proposed for binary encoding but which are also applicable to real coding. When they are used, the value of each gene in the offspring coincides with the value of the same gene in one of the parents. A representative of this group is the discrete crossover operator [41], which generates an offspring  $\mathbf{o} = \{o_1, o_2, \dots, o_n\}$  from two parents  $\mathbf{p}_1 = \{p_{1,1}, \dots, p_{1,n}\}$  and  $\mathbf{p}_2 = \{p_{2,1}, \dots, p_{2,n}\}$  where each gene  $o_j$  takes a random value from the set  $\{p_{1,j}, p_{2,j}\}$ .
2. Aggregation based crossover operators. They are deterministic and use aggregation functions to combine the numeric values of the genes of the parents to generate the offspring. The arithmetical operator [2] has been chosen as a typical example of this group. When using arithmetical crossover, two offsprings are generated such that  $o_{1,j} = \beta \cdot p_{1,j} + (1 - \beta) \cdot p_{2,j}$  and  $o_{2,j} = \beta \cdot p_{2,j} + (1 - \beta) \cdot p_{1,j}$ . In this work, the suggested value of  $\beta = 0.33$  has been adopted.
3. Neighborhood based operator. The genes of the offsprings are extracted from intervals in the neighborhoods of the parents, using probability distributions and including random components. Good representatives of this category are the BLX- $\alpha$  (blend crossover) algorithm [42] and the SBX- $\eta$  (simulated binary crossover), introduced in [43]. With the first one, a single offspring is generated whose gene  $o_i$  takes a uniformly chosen random value in the interval  $[\min_j(p_{1,j}, p_{2,j}) - l \cdot \alpha, \max_j(p_{1,j}, p_{2,j}) + l \cdot \alpha]$ , with  $l$  defined as  $\max_j(p_{1,j}, p_{2,j}) - \min_j(p_{1,j}, p_{2,j})$ . We have used  $\alpha = 0.5$  in this work. The second operator generates two offsprings according to a particular probability distribution that takes higher values around the neighborhoods of the parents, and depends on a parameter,  $\eta$ . A uniform distribution corresponds to  $\eta = 0$  and the higher its value the more peaked the distribution is around the parents. A value of  $\eta = 5$  has been fixed in this work.

Apart from discrete, arithmetical, blend and simulated binary crossovers, the flat crossover [44] has been also considered in this work. Flat crossover constitutes a very simple approach which can be considered as hybrid according to the categorization above. When this operator is used, a single offspring is created, and each gene  $o_j$  takes a uniformly distributed random value in the interval  $[p_{1,j}, p_{2,j}]$ . For illustrative purposes, Fig. 4 demonstrates the application of each of these operators on a 2-dimensional space of features.

Regarding mutation, a simple random uniform mutation operator has been employed [2] in the cases of discrete, SBX and BLX

crossover operators, replacing genes by a random value in the interval whose end points are the minimum and maximum values for the feature. In the case of the flat and arithmetical crossover operators, boundary mutation [45] has been used, replacing genes by either their lower or upper bound, giving special consideration to the arguments provided in [44] regarding crossover operators which systematically bias the search away towards the center of the intervals. On the other hand, although mutation rates of one mutation per individual are a common setting that has been reported to yield good results in a wide diversity of problems [46], different mutation rates will be considered and their impact in the context of the proposed approach will be measured and discussed.

Finally, it is also worth mentioning that possible inconsistencies may arise from the application of crossover and mutation operations. Features which belong to the same concept may have to satisfy certain restrictions. As an example, bins of a normalized



**Fig. 4.** Examples of offsprings generated from parents in (a) on a two feature space using discrete crossover (b), arithmetic crossover (c), BLX-0.5 crossover (d), SBX-5 crossover (e) and flat crossover (f).

color histogram should add to one, and cumulative functions should be monotonic. This means that the genotypes have to be repaired, an operation which depends on the particular features used in each case.

#### 4.2. Mapping genotypes to phenotypes using similarities

Once a convenient set of potentially relevant feature vectors  $\mathcal{O}$  has been obtained in a given relevance feedback iteration, appropriate mechanisms need to be introduced in order to produce the set of images from  $\mathcal{X}$  that best represents  $\mathcal{O}$  (so that they can be shown to the user and be judged in the next iteration). This can be seen as a mapping from feature vectors (genotypes, that might not have a corresponding image) to images in  $\mathcal{X}$  (phenotypes).

Instead of using a trivial mapping preserving meaningful (from the image representation viewpoint) information in  $\mathcal{O}$  only, a more elaborate approach that takes into account both positive (in  $\mathcal{O}$ ) and negative information (from  $\mathcal{Q}^-$ ) will be considered. This mapping will be defined using appropriately defined distances in the feature space from the candidate elements in  $\mathcal{X}$  to genotypes in  $\mathcal{O}$  (relevant distances) and to elements in  $\mathcal{Q}^-$  (non-relevant distances).

##### 4.2.1. Feature weighting

In order to give different importance to different image descriptors according to the concept being searched, and in a similar way as in most query movement approaches that introduce a dynamic feature weighting scheme to improve the retrieval results (e.g. [7,9]), a local weighting strategy is considered. As image feature vectors are supposed to convey multiple representations, each of them with its corresponding distance function, locally defined weights will be assigned to each of these distances. Then, the overall distance between two feature vectors will be computed as a linear combination. In particular, when measuring the proximity of a feature vector  $\mathbf{x}_i$  to an offspring  $\mathbf{o}$  whose parents are  $\mathbf{p}_1$  and  $\mathbf{p}_2$ , the weight for each distance,  $d_k$ , will be set as

$$w_k = \frac{a}{d_s(p_1, p_2)}$$

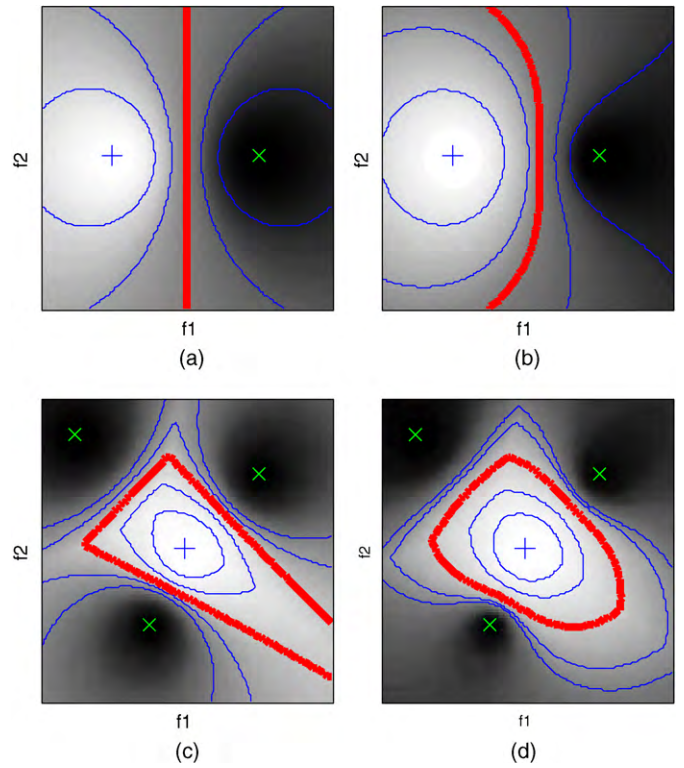
where constant  $a$  is chosen so that  $\sum_{k=1}^t w_k = 1$ . This weighting scheme gives more importance to those representations in which the parents in  $\mathcal{Q}^+$  were closer. In Section 5, we see that this strategy yields a considerable improvement of retrieval results.

##### 4.2.2. Extended nearest-neighbor (NN) approach

Once a composite distance between feature vectors including feature weighting is given, we need to measure the overall relevance of each image,  $x_i$  with regard to a particular concept whose only information available is contained in  $\mathcal{Q}^-$  and  $\mathcal{O}$ . Let  $dR(x_i)$  denote the distance from  $\mathbf{x}_i$  to the nearest relevant feature vector (from  $\mathcal{O}$ ) and let  $dN(x_i)$  be the distance from  $\mathbf{x}_i$  to the nearest non-relevant feature vector (from  $\mathcal{Q}^-$ ). In [35], the application of a nearest-neighbor approach to rank similarities in image retrieval was described. In these works, each image  $x_i$  is assigned a score according to the equation<sup>1</sup>

$$\frac{dN(x_i)}{dR(x_i)} \quad (1)$$

This ratio can be related to the ratio between the densities of relevant and non-relevant images in the feature space and consequently high values correspond to regions with more chances of finding relevant images. Unfortunately, this approach suffers from the small sample size problem and large score values can



**Fig. 5.** Contour plots illustrating the effect of using Eq. (1) or (2) to define image similarity (in a 2-dimensional Euclidean space). Plus and cross signs represent relevant and non-relevant samples, respectively. The contours represent points of equal relevance score and gray levels correspond to their normalized values. (a) and (c) show the results when using Eq. (1). (b) and (d) when using Eq. (2).

be obtained for uninteresting images that are far from both relevant and non-relevant ones. In order to avoid this problem and give precedence to images that are closer to (potentially) relevant images in  $\mathcal{O}$ , Eq. (1) is modified by introducing a new term that penalizes the distance to the nearest offspring,  $1/dR(x_i)$ . This yields the following new expression which has been used in our algorithm:

$$\frac{dN(x_i)}{dR(x_i)^2} \quad (2)$$

Note that this simple modification will stop getting high relevance scores to outliers and will concentrate the search in the regions around feature vectors from  $\mathcal{O}$ . In Fig. 5, differences in using Eq. (1) or (2) are illustrated. The use of the latter results in a higher robustness when a relevant sample is close to other non-relevant samples.

##### 4.2.3. Locally searching for neighbors

Even with the above proposed relevance score which takes into account both positive and negative information there is still a problem related to the fact that the feature space may be very unevenly populated by images in  $\mathcal{X}$ . The observed effect is that the images in  $\mathcal{X}$  that maximize the above score may not necessarily come from all potentially relevant regions. To avoid this, the computation of Eq. (2) for all images in  $\mathcal{X}$  is substituted by a local search around each offspring  $\mathbf{o}_\ell$  in  $\mathcal{O}$  for the image which maximizes the relevance with regard to this offspring only measured as

$$\frac{dN(x_i)}{dR_\ell(x_i)^2}$$

where  $dR_\ell(x_i)$  is the composite distance from  $\mathbf{x}_i$  to the  $\ell$  th offspring,  $\mathbf{o}_\ell$ .

<sup>1</sup> In the context in which this ratio has been previously used, relevant vectors are taken from  $\mathcal{Q}^+$  as there is no set  $\mathcal{O}$ .



As a consequence of this local way of looking for nearest neighbors in  $\mathcal{X}$ , not only relevant images are obtained but also representative ones as all elements in  $\mathcal{O}$  are equally represented in the result.

The combination of the feature weighting scheme and the extended NN approach is aimed at avoiding the above mentioned problems and provides improved results. The first technique allows the CBIR system to focus on the features that the user has emphasized while the second embeds negative selections into the retrieval process, enriching the amount of information handled by the algorithm.

## 5. Experiments and results

### 5.1. Databases used

A number of comparative experiments have been carried out in order to assess the relative merits of the proposed approach in its different aspects and in comparison to other competing or alternative approaches. For this purpose, we have used objective measures of performance on several different representative retrieval experiments using two different databases which are representative of a range of different situations. These databases are:

- The subset of the Corel database used in [35]. This is composed of 30 000 images which were manually classified into 71 categories. The descriptors used are those provided in the KDD-UCI repository (<http://kdd.ics.uci.edu/databases/CorelFeatures>) namely: A 9-dimensional vector with the mean, standard deviation and skewness for each hue, saturation and value in the HSV color space; a 16-dimensional vector with the co-occurrence in horizontal, vertical and the two diagonal directions; a 32-dimensional vector with the  $4 \times 2$  color HS histograms for each of the resulting subimages after one horizontal and one vertical split; and a 32-dimensional vector with the HS histogram for the entire image.
- A small repository which was intentionally assembled for testing, using some images obtained from the Web and others taken by the authors. The 1508 pictures it contains are classified as belonging to 29 different themes such as flowers, horses, paintings, skies, textures, ceramic tiles, buildings, clouds, and trees. This repository is a subset of the one used in the evaluation sections of [14,11]. In this case, the descriptors include a  $10 \times 3$  HS color histogram and texture information in the form of two granulometry cumulative distribution functions [36].

The (dis)similarity between pairs of images in each considered representation space (descriptor) is computed as the Euclidean distance except for the case of HS histograms in which the histogram intersection [47] has been used.

### 5.2. Experimental settings

A residual collection [48] experimental setting similar to those reported in [49,50] has been built. To simulate a search, a target image is chosen from the repository and all other pictures in the collection are ranked according to their distance to the target. To this end, the distances produced by each descriptor are added after a Gaussian normalization [51,8], to put equal emphasis on the distances on each feature space. Then user selections are performed automatically, by using the first  $p$  images in the ranking to build the sets  $Q^+$  and  $Q^-$  according to the class information available. Next, these  $p$  images are removed from the collection and the two sets are used as the input for the corresponding relevance feedback algorithm, to evaluate its performance at retrieving images

**Table 1**

Averaged retrieval precision for  $p = 20$  and  $p = 50$  and improvement with respect to the initial query for each crossover strategy considered.

	Small database		Large database	
	$p = 20$	$p = 50$	$p = 20$	$p = 50$
Initial query				
Precision	0.3425	0.2533	0.1928	0.1368
Discrete				
Precision	0.5055	0.3933	0.2792	0.2318
Improvement	+47.59%	+55.27%	+44.81%	+69.44%
Arithmetical				
Precision	0.5189	0.3996	0.2816	0.2362
Improvement	+51.50%	+57.76%	+46.06%	+72.66%
Flat				
Precision	0.5198	0.4003	0.2811	0.2334
Improvement	+51.77%	+58.03%	+45.80%	+70.61%
SBX				
Precision	0.5106	0.3898	0.2779	0.2361
Improvement	+49.08%	+53.89%	+44.14%	+72.59%
BLX				
Precision	0.5024	0.3858	0.2751	0.2261
Improvement	+46.69%	+52.31%	+42.69%	+65.28%

in the same class as the original target. This is expressed in terms of the precision achieved (the proportion of images which belong to the chosen class), measured on the  $p$  first retrieved images. In the experiments, two different values of  $p$  have been used, namely  $p = 20$  and  $p = 50$ . The former represents a typical value in real CBIR systems; the latter aims at assessing the influence of increasing the amount of feedback provided to the algorithm.

In order to obtain more reliable data, each experiment has been repeated with 500 random searches and the results have been averaged. In all cases (and with all competing algorithms) we have forced the situation that there is at least one relevant sample among the first  $p$  images in the initial order in which pictures are presented.

### 5.3. Evaluation of evolutionary operators

In this section, we analyze the differences in performance associated with the use of different crossover operators and mutation rates. Regarding the former, Table 1 shows the retrieval precision obtained for each of the crossover strategies evaluated, in both repositories. No significant differences in performance can be observed, although the flat and arithmetical operators seem to yield slightly better results.

The effect of varying the mutation rate has been tested for all crossover strategies considered. In Table 2, the averaged results are shown when using the flat crossover operator. Results using other crossover operators were very similar. As it can be observed, consistently worse results are obtained as the mutation rate is increased. This effect could be partially due to the implicit mutation that happens when the offsprings are mapped to the closest existing image in  $\mathcal{X}$ . Or in other words, the (possible) benefits of using mutation in this context gets masked by the distance-based mapping process which is further applied. This reasoning could explain also why very different crossover techniques lead to virtually the same results (although good enough).

### 5.4. Evaluation of distance-based mapping

In this section, we evaluate the separate effect of the feature weighting scheme and the adapted NN approach which constitute the main components in the proposed algorithm.

To study the improvement achieved with each of these techniques independently, the algorithm is run (a) as proposed (labeled

**Table 2**

Algorithm performance on the small repository, using different mutation rates (measured as the average number of mutations per individual), when using the flat crossover operator. Other crossover operators yield very similar results.

	Small database		Large database	
	$p = 20$	$p = 50$	$p = 20$	$p = 50$
Initial query Precision	0.3425	0.2533	0.1928	0.1368
Mutation rate = 0				
Precision	0.5198	0.4003	0.2811	0.2334
Improvement	+51.77%	+58.03%	+45.80%	+70.61%
Mutation rate = 1				
Precision	0.4989	0.3817	0.2726	0.2322
Improvement	+45.66%	+50.69%	+41.39%	+69.74%
Mutation rate = 4				
Precision	0.4256	0.3394	0.2520	0.2166
Improvement	+24.26%	+33.99%	+30.71%	+58.33%
Mutation rate = 16				
Precision	0.2190	0.1907	0.1421	0.1251
Improvement	−36.06%	−24.71%	−26.30%	−8.55%

as weights + NN), (b) the proposed algorithm but without using feature weighting (labeled as NN only), (c) the proposed one but without the extended NN-based way of computing relevances (labeled as weights only) and (d) the proposed one without any of the two improvements.

As an example, the retrieval results obtained (using a flat crossover) are shown in Table 3 (others are consistent with these). Note that the algorithm without any of the two above mentioned components will work under very similar principles as the algorithm presented in [3].

As it can be observed, the relative behavior of the four algorithms considered is the expected one, and a significant and consistent relative improvement is obtained when the extended NN approach proposed here is used.

### 5.5. Comparative analysis

In this section, we compare the performance of the algorithm against other representative CBIR techniques. These other CBIR strategies have been chosen so that they are representatives of the groups of methods presented in Section 2. In particular, (a) a classical feature weighting and query movement approach, imple-

**Table 3**

Improvement achieved by using the weighting scheme and the nearest-neighbor approach.

	Small database		Large database	
	$p = 20$	$p = 50$	$p = 20$	$p = 50$
Initial query Precision	0.3425	0.2533	0.1928	0.1368
Weights + NN				
Precision	0.5198	0.4003	0.2811	0.2334
Improvement	+51.77%	+58.03%	+45.80%	+70.61%
NN only				
Precision	0.5084	0.3892	0.2743	0.2257
Improvement	+48.44%	+53.65%	+42.27%	+64.99%
Weights only				
Precision	0.4771	0.3694	0.2604	0.2129
Improvement	+39.30%	+45.83%	+35.06%	+55.63%
None				
Precision	0.4604	0.3631	0.2549	0.2093
Improvement	+34.42%	+43.35%	+32.21%	+53.00%

**Table 4**

Averaged retrieval precision for  $p = 20$  and  $p = 50$  and improvement with respect to the initial query for each algorithm considered in the comparison.

	Small database		Large database	
	$p = 20$	$p = 50$	$p = 20$	$p = 50$
Initial query Precision	0.3425	0.2533	0.1928	0.1368
Proposed				
Precision	0.5198	0.4003	0.2811	0.2334
Improvement	+51.77%	+58.03%	+45.80%	+70.61%
NN				
Precision	0.4814	0.3694	0.2463	0.1918
Improvement	+40.55%	+45.83%	+27.75%	+40.20%
SOM				
Precision	0.3839	0.3035	0.2128	0.1868
Improvement	+12.09%	+19.82%	+10.37%	+36.55%
QPM				
Precision	0.3501	0.2719	0.2182	0.1756
Improvement	+2.22%	+7.34%	+13.17%	+28.36%
Fuzzy				
Precision	0.4100	0.3464	0.1204	0.0856
Improvement	+19.71%	+36.75%	−37.55%	−37.43%

mented as in [9,52]; (b) a classification based engine that uses similar principles to that used in the PicSOM system [12]; (c) a fuzzy strategy implemented as described in [25]; and (d) the nearest-neighbor approach presented in [35]. We will refer to these algorithms as QPM (query point movement), SOM, fuzzy and NN, respectively. The SOM-based approach uses  $64 \times 64$  SOMs for the large repository and  $16 \times 16$  SOMs for the small one. Apart from minor details, all different methods including any parameter settings have been implemented as in the corresponding references, and adapted to work with the same feature sets to allow for a fair comparison.

As a first experiment we have used the same experimental setting as above. The averaged retrieval results obtained are shown in Table 4 (using a flat crossover). It can be observed that, except for the fuzzy approach, the results obtained with the rest of the algorithms are consistent across the two image collections. In both cases the proposed approach behaves significantly better than all other competing algorithms considered, and the plain NN approach consistently gives the second best result confirming that this is also a very convenient and robust algorithm. The fuzzy technique performs well on the small collection but behaves worse than sorting by distance to the query in the large database. This may in part be due to the smaller amount of positive samples provided in the case of the large repository.

A second experiment has the purpose to further verify the validity of the approach by comparing the behavior of the techniques as the amount of feedback grows. To this end, an experimental setup similar to those reported in [23,53,54] has also been considered. In this case, initial feedback is provided from a population of  $p$  images extracted at random from the collection (forcing the situation that there is at least one positive sample), and several relevance feedback iterations are run. To allow for a fair comparison, all relevant images from previous iterations are plugged into the retrieved results which are the input for the next iteration, causing monotonically better results as iterations increase (note that the proposed algorithm does not benefit from this because a similar effect is already caused by the elitism strategy defined). Again, performance at each iteration is measured as the precision achieved on the  $p$  first retrieved images.



**Table 5**Averaged retrieval precision on the first  $p = 50$  retrieved images for all algorithms considered.

Database	Method	Iteration									
		1	2	3	4	5	6	7	8	9	10
Small	Proposed	0.4882	0.6412	0.7056	0.7393	0.7625	0.7778	0.7909	0.8010	0.8088	0.8160
	NN	0.4834	0.6151	0.6762	0.7106	0.7344	0.7511	0.7650	0.7758	0.7854	0.7934
	SOM	0.4331	0.5622	0.5934	0.6138	0.6222	0.6300	0.6345	0.6371	0.6393	0.6406
	QPM	0.4389	0.5260	0.5616	0.5728	0.5794	0.5816	0.5844	0.5849	0.5865	0.5868
	Fuzzy	0.4550	0.5892	0.6680	0.7093	0.7356	0.7562	0.7716	0.7834	0.7928	0.8003
Large	Proposed	0.2190	0.3792	0.4864	0.5623	0.6170	0.6601	0.6962	0.7270	0.7510	0.7710
	NN	0.2380	0.3703	0.4450	0.5128	0.5599	0.6016	0.6348	0.6630	0.6876	0.7097
	SOM	0.1974	0.3207	0.3706	0.4097	0.4280	0.4453	0.4548	0.4627	0.4677	0.4722
	QPM	0.1958	0.2836	0.3299	0.3551	0.3705	0.3802	0.3868	0.3901	0.3931	0.3947
	Fuzzy	0.1504	0.2680	0.3943	0.4767	0.5500	0.6058	0.6450	0.6805	0.7074	0.7304

For this experiment we have fixed  $p = 50$  and run 10 feedback iterations. Although these are not a common value for realistic CBIR systems (note that this would involve 500 user evaluations in a real setting), lesser values of  $p$  would cause that all algorithms reach very high precision values rapidly and hinder the appreciation of differences in performance. In this sense, the value  $p = 50$  fits the purpose of this experiment better. Results (using a flat crossover) are shown in both Table 5 and Fig. 6, and are consistent with those reported in Table 4. Again, the proposed approach outperforms all the other competing algorithms considered. It can also be observed that the plain NN approach gives the second best result in the first four relevance feedback iterations while the fuzzy approach exhibits a better behavior from this point on in the two databases. This suggests that the latter method is able to learn comparatively faster than the others as more information is given.

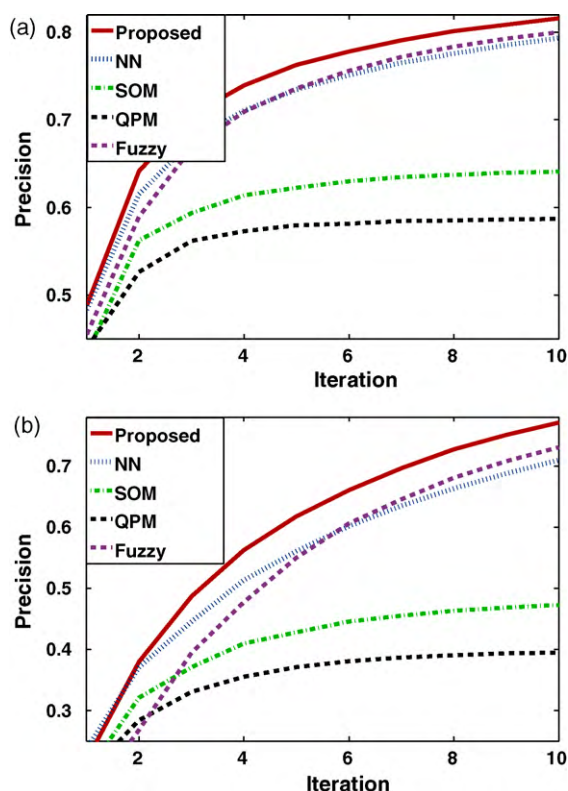
## 6. Concluding remarks and further work

A new hybrid approach to relevance feedback content-based image retrieval has been introduced. This technique combines an interactive genetic algorithm with an extended nearest-neighbor approach using adaptive distances and local searches around several promising regions, instead of computing a single ranking. Although the proposed method has many components with corresponding parameters, the performance results proved to be relative insensitive to many of these. Three main factors influence the good performance that the method exhibits according to an exhaustive empirical evaluation, part of which has been shown here. First, the genetic and consequently parallel way of looking for new promising regions along with the old ones. Second, the adaptive/weighted composite distance (conveniently tuned or modulated by the negative information available) used for computing relevance scores. And third, a local search approach around promising regions that effectively retrieve images. The results obtained show significant improvements in comparison to the base methods which are directly related to the presented ones, according to the experimental setting which has been inherited from the corresponding original works.

Many possibilities both to extend the approach presented and to validate the method in different scenarios and against other state-of-the-art methods have been left as further work. The main extension is related to the way the method has been evaluated and aims at introducing real users. Although the method has been implemented and has been run in practice with acceptable results, doing a thorough evaluation with real users falls out of the scope of the presented work. Moreover, more fine tuning should be done (in all competing methods also) in order to involve real users. In this direction, another interesting possibility aimed at improving convergence while minimizing user fatigue would consist of running several intermediate GA iterations without user judgments using alternative objective fitness measures. The second obvious extension which is already in its way, consists of including feedback as grades of relevance instead of yes/no answers. Finally, some minor aspects of the presented approach are being now specifically studied to assess their relative importance in the approach. This includes several modifications to the local search for retrieving images and introducing different forgetting policies to manage positive and negative information across iterations when concept drift occurs.

## Acknowledgments

This work has been funded by FEDER and the Spanish and Valencian governments through projects Consolider Ingenio 2010 CSD2007-00018, TIN2009-14205-C04-03, TIN2006-12890, GV-2008/032 and GV-2005/194 ATRAPA.



**Fig. 6.** Averaged retrieval precision ( $p = 50$ ) at each iteration for each algorithm considered: (a) in the small repository and (b) in the large repository.

The authors would like to thank Dr. G. Giacinto for his help on the evaluation of this work, by facilitating the manually performed classification of the large repository. We would also like to thank Dr. M. Ortega for providing the thumbnails of these images.

## References

- [1] C. Darwin, *The Origin of Species*, Dent Gordon, London, 1973.
- [2] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, New York, 1992.
- [3] S.-B. Cho, J.-Y. Lee, A human-oriented image retrieval system using interactive genetic algorithms, *IEEE Transactions on Systems, Man, and Cybernetics. Part A: Systems and Humans* 32 (3) (2002) 452–458.
- [4] H. Takagi, Interactive evolutionary computation: fusion of the capabilities of EC optimization and human evaluation, *Proceedings of the IEEE* 89 (9) (2001) 1275–1296.
- [5] X. Zhou, T. Huang, Relevance feedback for image retrieval: a comprehensive review, *Multimedia Systems* 8 (6) (2003) 536–544.
- [6] R. Datta, D. Joshi, J. Li, J.Z. Wang, Image retrieval: ideas, influences, and trends of the new age, *ACM Computing Surveys* 40 (2) (2008) 1–60, <http://doi.acm.org/10.1145/1348246.1348248>.
- [7] Y. Ishikawa, R. Subramanya, C. Faloutsos, Mindreader: querying databases through multiple examples, in: *Proceedings of the 24th International Conference on Very Large Data Bases, VLDB*, New York, USA, 1998, pp. 433–438.
- [8] Y. Rui, S. Huang, M. Ortega, S. Mehrotra, Relevance feedback: a power tool for interactive content-based image retrieval, *IEEE Transactions on Circuits and Video Technology* 8 (5) (1998) 25–36.
- [9] G. Ciocca, R. Schettini, A relevance feedback mechanism for content-based image retrieval, *Information Processing and Management* 35 (1) (1999) 605–632.
- [10] D. Liu, K.A. Hua, K. Vu, N. Yu, Fast query point movement techniques for large cbir systems, *IEEE Transactions on Knowledge and Data Engineering* 21 (2009) 729–743, <http://dx.doi.org/10.1109/TKDE.2008.188>.
- [11] T. León, P. Zuccarello, G. Ayala, E. de Ves, J. Domingo, Applying logistic regression to relevance feedback in image retrieval systems, *Pattern Recognition* 40 (10) (2007) 2621–2632, <http://dx.doi.org/10.1016/j.patcog.2007.02.002>.
- [12] J. Laaksonen, M. Koskela, E. Oja, PicSOM: self-organizing image retrieval with mpeg-7 content descriptors, *IEEE Transactions on Neural Networks* 13 (4) (2002) 841–853.
- [13] I. Cox, M. Miller, T. Minka, T.V. Papathomas, P. Yianilos, The bayesian image retrieval system, PicHunter: theory, implementation and psychophysical experiments, *IEEE Transactions on Image Processing* 9 (1) (2000) 20–37.
- [14] E. de Ves, J. Domingo, G. Ayala, P. Zuccarello, A novel bayesian framework for relevance feedback in image content-based retrieval systems, *Pattern Recognition* 39 (2006) 1622–1632.
- [15] N. Vasconcelos, A. Lippman, Learning from user feedback in image retrieval systems, *Advances in Neural Information Processing Systems* 12 (2000) 977–986.
- [16] Z. Su, H.J. Zhang, S. Li, S. Ma, Relevance feedback in content-based image retrieval: Bayesian framework, feature sub-spaces, and progressive learning, *IEEE Transactions on Image Processing* 12 (2003) 924–937.
- [17] G. Giacinto, F. Roli, Bayesian relevance feedback for content-based image retrieval, *Pattern Recognition* 37 (7) (2004) 1499–1508.
- [18] N. Cristianini, J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, 2000.
- [19] S. Tong, E. Chang, Support vector machine active learning for image retrieval, in: *Proceedings of the 9th ACM International Conference on Multimedia*, ACM Press, New York, NY, USA, 2001, pp. 107–118.
- [20] Y. Chen, X.S. Zhou, T.S. Huang, One-class svm for learning in image retrieval, in: *Proceedings of the IEEE International Conference on Image Processing*, 2001, pp. 34–37.
- [21] I. Gondra, R. Heisterkamp, J. Peng, Improving image retrieval performance by inter-query learning with one-class support vector machines, *Neural Computing & Applications* 13 (2) (2004) 130–139.
- [22] D. Tao, X. Tang, X. Li, X. Wu, Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (7) (2006) 1088–1099.
- [23] G. Giacinto, A nearest-neighbor approach to relevance feedback in content based image retrieval, in: *Proceedings of the 6th ACM International Conference on Image and Video Retrieval (CIVR'07)*, ACM Press, Amsterdam, The Netherlands, 2007, pp. 456–463.
- [24] Y. Chen, J. Wang, R. Krovetz, Clue: cluster-based retrieval of images by unsupervised learning, *IEEE Transactions on Image Processing* 14 (8) (2005) 1187–1201.
- [25] M. Arevalillo-Herráez, M. Zacarés, X. Benavent, E. de Ves, A relevance feedback CBIR algorithm based on fuzzy sets, *Signal Processing: Image Communication* 23 (7) (2008) 490–504.
- [26] O. Córdón, E. Herrera-Viedma, C. López-Pujalte, M. Luque, C. Zarco, A review on the application of evolutionary computation to information retrieval, *International Journal of Approximate Reasoning* 34 (2003) 241–264.
- [27] R. da S. Torres, A.X. Falco, M.A. Goncalves, B. Zhang, W. Fan, E.A. Fox, P. Calado, A new framework to combine descriptors for content-based image retrieval, in: *Fourteenth Conference on Information and Knowledge Management, Bremen, Germany*, 2005, pp. 335–336.
- [28] Q. Zhang, E. Izquierdo, Adaptive salient block-based image retrieval in multi-feature space, *Signal Processing: Image Communication* 22 (2007) 591–603.
- [29] S. Bissol, P. Mulhem, Y. Chiaramella, Dynamic learning of indexing concepts for home image retrieval, in: *Content-Based Multimedia Indexing (CBMI'03)*, Rennes, France, 2003, pp. 87–93.
- [30] K.D. Tran, Content-based retrieval using a multi-objective genetic algorithm, in: *Proceedings IEEE SoutheastCon*, 2005, pp. 561–569.
- [31] M. Saadatmand-Tarzjan, H. Abrishami Moghaddam, A novel evolutionary approach for optimizing content-based image indexing algorithms, *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 37 (1) (2007) 139–153.
- [32] S.-F. Wang, X.-F. Wang, J. Xue, An improved interactive genetic algorithm incorporating relevant feedback, in: *Proceedings of the 4th International Conference on Machine Learning and Cybernetics, Guangzhou*, 2005, pp. 2996–3001.
- [33] J. Urban, J.M. Jose, Evidence combination for multi-point query learning in content-based image retrieval, in: *ISMSE'04: Proceedings of the IEEE Sixth International Symposium on Multimedia Software Engineering*, IEEE Computer Society, Washington, DC, USA, 2004, pp. 583–586.
- [34] B. Sendho, M. Kreutz, W. von Seelen, A condition for the genotype-phenotype mapping: causality, in: *Bäck (Ed.), Proceedings of the 7th International Conference on Genetic Algorithms*, Morgan Kaufmann, San Francisco, 1997, pp. 2996–3001.
- [35] G. Giacinto, F. Roli, Nearest-prototype relevance feedback for content based image retrieval, in: *ICPR'04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04)*, vol. 2, IEEE Computer Society, Washington, DC, USA, 2004, pp. 989–992.
- [36] P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag, Berlin, 2003.
- [37] F. Herrera, M. Lozano, J.L. Verdegay, Tackling real-coded genetic algorithms: operators and tools for behavioural analysis, *Artificial Intelligence Review* 12 (4) (1998) 265–319.
- [38] M. Raghuwanshi, O. Kakde, Survey on multiobjective evolutionary and real coded genetic algorithms, *Complexity International* 11 (2005) 150–161.
- [39] K. Deep, M. Thakur, A new crossover operator for real coded genetic algorithms, *Applied Mathematics and Computation* 188 (2007) 895–911.
- [40] F. Herrera, M. Lozano, A. Sánchez, Hybrid crossover operators for real coded genetic algorithms: an experimental study, *Soft Computing* 9 (2005) 280–298.
- [41] H. Mühlenbein, D. Schlierkamp-Voosen, Predictive models for the breeder genetic algorithm. I. Continuous parameter optimization, *Evolutionary Computation* 1 (1993) 25–49.
- [42] L. Eshelman, J. Shaffer, Real coded genetic algorithms and interval schemata, in: *Proceedings of the 2nd Workshop on Foundations of Genetic Algorithms (FOGA'02)*, Morgan Kaufmann, 1992, pp. 187–202.
- [43] K. Deb, R. Agrawal, Simulated binary crossover for continuous search space, *Complex Systems* 9 (1995) 115–148.
- [44] N. Radcliffe, Equivalence class analysis of genetic algorithms, *Complex Systems* 5 (2) (1991) 183–205.
- [45] Z. Michalewicz, M. Schoenauer, Evolutionary algorithms for constrained parameter optimization problems, *Evolutionary Computation* 4 (1) (1996) 1–32.
- [46] T. Bäck, Optimal mutation rates in genetic search, in: M. Kaufmann (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms*, San Mateo, CA, 1993, pp. 2–8.
- [47] M.J. Swain, D.H. Ballard, Color indexing, *International Journal of Computer Vision* 7 (1) (1991) 11–32, <http://dx.doi.org/10.1007/BF00130487>.
- [48] Y. Chang, C. Cirillo, J. Razor, Evaluation of feedback retrieval using modified freezing, residual collection and test and control groups, in: G. Salton (Ed.), *The Smart Retrieval System—Experiments in Document Processing*, Prentice Hall Inc., Englewood Cliffs, NJ, 1971, pp. 355–370.
- [49] G. Salton, C. Buckley, Improving retrieval performance by relevance feedback, *Journal of the American Society for Information Science* 41 (4) (1990) 288–297.
- [50] C. López-Pujalte, V.P. Guerrero-Bote, Order-based tness functions for genetic algorithms applied to relevance feedback, *Journal of the American Society for Information Science and Technology* 54 (2) (2003) 152–160.
- [51] Q. Iqbal, J. Aggarwal, Combining structure color and texture for image retrieval: a performance evaluation, in: *16th International Conference on Pattern Recognition (ICPR)*, vol. 2, 2002, pp. 438–443.
- [52] G. Ciocca, R. Schettini, Content-based similarity retrieval of trademarks using relevance feedback, *Pattern Recognition* 34 (8) (2001) 1639–1655.
- [53] D. Tao, X. Li, S. Maybank, Negative samples analysis in relevance feedback, *IEEE Transactions on Knowledge and Data Engineering* 19 (4) (2007) 568–580.
- [54] M. Arevalillo-Herráez, F.J. Ferri, J. Domingo, A naive relevance feedback model for content-based image retrieval using multiple similarity measures, *Pattern Recognition* 43 (3) (2010) 619–629.