Pattern Recognition Letters 53 (2015) 109–117

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Improving distance based image retrieval using non-dominated sorting genetic algorithm☆

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

Dept. d'Informàtica, Universitat de València, Avda. Universitat s/n, 46100 Burjassot, Spain

ARTICLE INFO

Article history:
Received 1 March 2014
Available online 29 May 2014

Keywords:
Image retrieval
Multiobjective genetic algorithm
Relevance feedback

ABSTRACT

Relevance feedback has been adopted as a standard in Content Based Image Retrieval (CBIR). One major difficulty that algorithms have to face is to achieve and adequate balance between the exploitation of already known areas of interest and the exploration of the feature space to find other relevant areas. In this paper, we evaluate different ways to combine two existing relevance feedback methods that place unequal emphasis on exploration and exploitation, in the context of distance-based methods. The hybrid approach proposed has been evaluated by using three image databases of various sizes that use different descriptors. Results show that the hybrid technique performs better than any of the original methods, highlighting the benefits of combining exploitation and exploration in relevance feedback tasks.

1. Introduction

Usually, a CBIR system represents the images in the repository as a multi-dimensional feature vector extracted from a series of low level descriptors, such as color, texture or shape. The perceptual similarity between two pictures is then quantified in terms of a distance/similarity function defined on the corresponding multi-dimensional feature space. A major problem with CBIR systems is the so called "semantic gap", which refers to the difficulty of translating the user's intentions into similarities amongst low level features. Relevance feedback, a technique inherited from traditional information retrieval, has been used to increase the efficiency of CBIR systems helping to induce high level semantic concepts from low level descriptors. When relevance feedback is used, a search is considered an iterative process. At each iteration, the system retrieves a series of images ordered according to a pre-defined similarity measure, and requires user interaction to mark the relevant and non relevant retrievals. This data is then used to adapt the similarity measure and produce a new set of pictures, repeating the process until the desired results are obtained.

Relevance feedback has been a major topic of research during the last two decades (see [41,16]). First proposals were based on adapting the similarity measure and moving the query point in the feature space so that more emphasis is placed on relevant elements and less on irrelevant ones [28,35,12]. This type of techniques use the user's judgments to dynamically adjust the weights of each feature, and to produce a new query point that represents his/her interest in a more reliable way. In general, these are the fastest techniques, but they assume the existence of a unique query point. A large number of probabilistic methods have also been proposed [22,42,2]. Most of these are based on estimating posterior probabilities from priors and the relevance judgments provided by the user. One particular way to estimate these probabilities is by using nearest neighbor estimates [24,1]. The use of supervised learning techniques has also been a major trend in the development of relevance feedback mechanisms. In this context, support vector machines have been widely used [44,11,40], despite the difficulties associated with choosing an appropriate tunning [26]. Other successful approaches to CBIR include the use of fuzzy sets [5] or self organized maps [29].

Most of these methods follow a so called exploitation approach, and focus the search on the vicinity of previous relevant retrievals, in an attempt to exploit already known regions of interest in the original multidimensional feature space (or another transformed space). Some recent works also include an exploratory component, and simultaneously attempt to combine the selections to further explore the space to find new regions of interest. In [24,34], an exploratory term is artificially added, by appropriately combining two already existing methods, namely nearest neighbor [26] and Bayesian Query Shifting [25].

☆ This paper has been recommended for acceptance by Jose Ruiz-Shulcloper.

* Corresponding author. Tel.: +34 96 354 3954; fax: +34 96 354 4768.
E-mail address: francesc.ferri@uv.es (F.J. Ferri).

http://dx.doi.org/10.1016/j.patrec.2014.05.008

In other cases, exploitation and exploration have been combined in a more explicit way. For example, the proposal presented in [3] integrates an interactive genetic algorithm with an extended nearest neighbor approach, using adaptive distances and local searches around several promising regions. The application of crossover operations on pairs of relevant samples yields new searches both around them and in new regions located between them. In [32], a multi-objective optimization algorithm is used to produce a set of potentially relevant pictures, conveniently scattered across the search space. A hybrid approach that uses this method only at the first iteration of relevance feedback has also been presented in [4], already outlining the advantages of using hybrid techniques that combine exploration and exploitation.

In this paper we further investigate different hybridization alternatives to combine exploitation and exploration. Following with the works in [32,4], the nearest neighbor method has been combined with the multi-objective optimization approach in several distinct ways. The combination of these two strategies is shown to produce better results than any of the methods alone.

2. Distance-based relevance feedback approaches

Let us assume that we have a repository whose m images are represented by multi-dimensional feature vectors $\mathcal{X} = \{x_i\}_{i=1}^{m}$. Assume also that a similarity function $s(x_i, x_j)$ exists, which is able to produce an estimate of the resemblance between any two images by comparing their feature vectors.

At any particular moment in the relevant feedback process, we have positive (relevant) and negative (non relevant) information about the current query or concept being searched for in the form of sets $\mathcal{P} = \{y_1, \ldots, y_p\}$ and $\mathcal{N} = \{z_1, \ldots, z_n\}$ that correspond to the images already shown to and judged by the user. Let us define $\mathcal{W}$ as the set of unseen images.

Relevance feedback methods in general aim at obtaining a (partial or total) ranking of images in $\mathcal{W}$ by using the (labeled) information in $\mathcal{P}$ and $\mathcal{N}$ and the similarity function.

2.1. Relevance feedback using nearest neighbor estimates

Nearest neighbors (NN) methods are known to produce robust estimates in general learning problems [15]. In the particular context of image retrieval, the similarities to the nearest relevant and nearest non relevant neighbors have been used to define ranking functions (scores) that are representative of the probability of relevance of any image x according to the information in $\mathcal{P}$ and $\mathcal{N}$. A widely used score function is the ratio

$$R(x) = \frac{s(x, \mathcal{P})}{s(x, \mathcal{N})} \quad (1)$$

This particular function or slight variations of it have been used to evaluate the relevance of images [24]. For any set of image representations, $s(x, \mathcal{Q}) = \max\{s(x, x_j) | x_j \in \mathcal{Q}\}$ is the similarity to the nearest (most similar) neighbor in $\mathcal{Q}$.

When multiple (and possibly heterogenous) descriptors are available (e.g. color, texture or shape) the corresponding similarity information needs to be combined either at the level of the similarity function to produce a unique score or by fusing different scores coming from each descriptor [16]. A common approach for both cases is the use of different forms of weighted linear combinations of normalized distances/scores in which the weights reflect the importance of the each descriptor [24].

3. Relevance feedback as a multi-objective search

3.1. Problem formulation

The use of multi-objective optimization in relevance feedback was first introduced in [32], in an attempt to find a sufficiently scattered set of relevant images that allowed the system to improve retrieval results at the following iterations. When a problem has multiple objectives, several optimal solutions may co-exist. These are all possible non-dominated solutions to the problem. A solution is said to be non-dominated if there is no other solution which simultaneously satisfies all the objectives better. In the absence of any further information, these cannot be said to be worse than any other. The set of all non-dominated solutions to a problem is commonly referred to as the Pareto optimal set.

Let us consider the following optimization problem

$$\max_{x \in \mathcal{W}} (s(x, y_1), \ldots, s(x, y_p)) \quad (2)$$

in which each $s(x, y_j)$ constitutes a different objective that needs to be optimized. The p-tuple $[s(x, y_1), \ldots, s(x, y_p)]$ represents both the resemblance between x and relevant information in $\mathcal{P}$ and also the degree of satisfaction of each objective.

Under this multi-objective formulation, images in or close to the Pareto optimal set can be considered to be judged by the user and conveniently update the sets $\mathcal{P}$ and $\mathcal{N}$ for the next iterations in the relevance feedback process. Unfortunately, determining Pareto optimal sets in discrete solution spaces is a simple but also a time consuming operation. Every solution has to be compared against the rest and, in the worst case, this takes $O(p \cdot m^2)$ in our particular case, which constitutes a prohibitive cost even for moderate size databases.

More importantly, the calculation of the strict Pareto optimal set may yield a large number of non-dominated solutions, specially for large sizes of $\mathcal{P}$. This would lead to an impractical number of images to be shown to the user. A convenient solution adopted in this and previous works [32] consists of choosing a representative and small subset of Pareto optimal elements in order to preserve the diversity.

The negative information in $\mathcal{N}$ is taken into account when selecting the definitive order in which images will be shown to the user instead of incorporating as other objectives into the above formulation [32].

3.2. Implementation issues

To reduce the computation needs associated with the direct computation of the Pareto optimal set from the images in $\mathcal{W}$, an alternative process consisting of two steps is followed. At the first step, we consider a continuous solution space, comprising all possible multi-dimensional representations. In this space, we use a heuristic approach to calculate a set of scattered vectors representing the Pareto optimal front. At the second stage, an independent search is conducted in $\mathcal{W}$ for each representative vector. Finally, the results for each search are combined to build the ordering in which pictures in $\mathcal{W}$ will be shown to the user.

For the first (continuous) stage, a multi-objective evolutionary algorithm can be used to determine a spread of solutions which is close to the true Pareto optimal front. In particular, the algorithm NSGA-II [19] has been chosen for this purpose because it leads to two major advantages, the parameterless diversity preservation mechanism of NSGA-II provides a representative spread set of solutions directly, with as many elements as the population size used. This means that the Pareto optimal set does not need to be