

Contribution for Content based Image Retrieval by Multiple Descriptors

Abdelkhalak Bahri¹, Hamid Zouaki² and Youssef Bouras³

^{1,2,3}LIMA Laboratory, Faculty of Science, Chouaib Doukkali University,
El Jadida, Morocco

¹bahri.abdelkhalak@gmail.com, ²hamid_zouaki@yahoo.fr,

³youssef.bouras@gmail.com

Abstract

One of the current challenges in the CBIR (Content Based Image Retrieval) domain is to find the best combination of several descriptors. Recently, several studies have been led to overcome this problem by integrating the genetic programming, however this technique required an important calculation time. The high computational cost is due to the high number of function calls required by this method and the high dimension of image descriptors used. To reduce this calculation time, we present a new search method that uses a multi-resolution BOF (Bag of Features) and genetic programming. In this context, for each descriptor type (e.i. color, shape, etc.), each image is described by a pyramid of histograms (pyramid of BOF), with each histogram represents the lower resolution of its above histogram. We start the similarity search by applying the GP (Genetic programming) to the lower histograms (top level of pyramid) for each descriptor type. In addition, we repeat this process just on the result obtained but by using the histograms correspond to the next levels of each pyramid of each image to refine the result obtained. The proposed method improves the search time without deteriorates the result quality in term of precision.

We validate our method against three databases. We have shown that our approach provides interesting and powerful experimental results.

Keywords: Bag of visual word, BOF, BOC, Multi-resolution, Genetic programming

1. Introduction

Recent advances technological in multimedia data acquisition led to an exponential growth in the number of images available. Image databases are now found in all areas of society. Browsing and searching information's in image databases is a crucial activity for the users. Traditionally, the images are annotated and searched by using textual words. The approaches and algorithms used in this case are directly from the world of natural language processing. Currently, information research is associated with any information type, textual, visual or otherwise. However, due to limitations of the textual methods, development of methods based on visual content has become paramount. However, this explains by diverse application areas related to imaging science [1] [2] and the intense research activity devoted to CBIR (Content Based Image Retrieval) system in recent years. The objective of CBIR is automatic structuring image collections from selected visual criteria [3]. The features offered cover efficient access to one or more images in a big collection, or even identify their contents through the detection or recognition of objects. There are many sectors handling this type of data, such as audiovisual, culture, security, health, scientific research, etc. Indexing an image from its visual content, it requires to produce from him a visual summary of its content for a particular purpose. In

Received (January 6, 2017), Review Result (August 31, 2017), Accepted (September 30, 2017)

terms of image descriptors, literature is now very rich: several families of descriptors exist, and in each family many approaches coexist[4]. For more information on feature extraction, the reader can be referred to [5].

In the area of image detection and recognition, more and more researches have studied local feature sets as a whole. In 2003, J. Sivic et al. were the first to have proposed the concept of Bag of Features (BOF) [6], that has become popular in recent years [6] thanks to its efficiency in the field of classification and annotation of images. In this context the image is represented by a single vector instead of multiple vectors (several SIFT descriptors). In this paper, we will focus on the BOF (bag of features) descriptors family [6]; we use the multi-resolution BOF approach to describe images [7], where each image is represented by a pyramid of BOF. Despite its success, image search by content has its own several limitations. It's often confronted with the problem of research relevance and the research time. In this paper, we focus on image search by content by combining multiple descriptors. However, the majority of the works in the literature ignore the research time. However, the search by several descriptors increases research time. In this work, we try to reduce this time without deterioration of the quality of search in terms of precision enjoying benefits of EBVW [7] description method and genetic programming [8] for the combination of several descriptors. Further in-depth introduction to our prior work could be found in the literatures [9] [10]. The rest of the paper is organized as follows: Section 2 describes the related work. In section 3 we present the details of our contribution. Section 4 evaluates the performances of our contribution experimentally. Finally, Section 5 concludes the paper.

2. Related Work

Many descriptors don't describe the same information and do not have the same invariance properties. The most features correspond to specific image properties: color histogram describes only the color distribution of images, SURF describes only the properties of local forms, etc. It's appropriate to combine their features in order to better describe the content of the image. For most scenarios, the combination of descriptors is typically the most successful approach. This combination can be implemented in different ways. We find in the literature several approaches, some merge the returned answers by each descriptor selected a priori by presupposing the complementarities of the latter [6] [11], others combine the descriptors in sequence [12]. Among the most recent, we find approaches dedicated to the categorization of images that learn an optimal combination of descriptors [13], which often means selecting the most discriminating characteristics for a given image [14].

In the same area, the authors have proposed in [15] different scenarios to perform the combination of several characteristics in general information retrieval and biometric recognition. In the particular case of CBIR, a multi-objective optimization technique based on a Pareto Archive Evolution Strategy (PAES) [16] has been used in [17] to define a global measure as an optimal linear combination of partial similarity functions. In the same part, the authors have proposed in [18] a probabilistic strategy to combine similarity measures.

It should be noted here that the majority of recent approaches to the literature are dedicated to image categorization and thus take little account of research time which will be reduced. In this paper, we propose a method which can reduce the search time by combining the advantages of multi-resolution BOF descriptors and genetic programming (GP) to combine the metrics applied to the signatures. In this work, we focus on image search by using color and shape descriptors. In the case of color, we use the pyramid of BOF descriptor [7] and for the case of the shape, we use the pyramid of BOF descriptor [7]. The use of GP is motivated in this work by the previous success of using this technique in information retrieval [19]. This

strategy is considered closely related to our work; Details of our contribution are presented in Section 3.

3. Multi-resolution Genetic Programming for Content based Image Retrieval

This section describes our contribution for content based image retrieval. Our key idea of our contribution is to find the best combination of color and shape for fast efficient nearest neighbor image search. For this, we propose to use the genetic programming [8] to find the best combination of metrics and the EBVW [7] method for image description.

3.1. Image Description using Multi-resolution BOF

In this sub section, we present the image description using multi-resolution BOF. This description is represented by a pyramid of signatures (histograms) using an EBVW method that we have introduced in [7] and the multi-resolution technique. The schema of this description is presented in Figure 1.

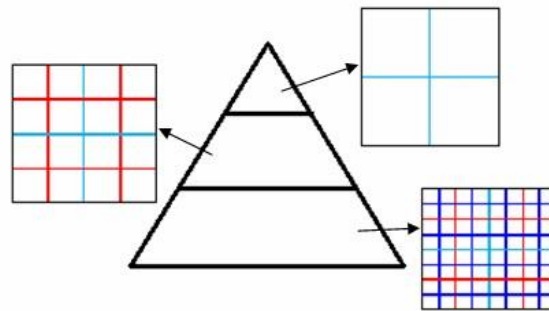


Figure 1. Image Description using EBVW Method and Multi-resolution Approach

3.2. Genetic Programming (GP)

3.2.1. Basic Concepts for GP

Genetic programming (GP) is introduced by Koza in [8], it's a type of evolutionary algorithm that is characterized by its ability to do evolve symbolic expressions. The term "symbolic expressions" here includes any structure that can be expressed as one or more sequences of actions or choices, such shares being finite in number. It's a hyper-heuristic optimization approach that has been applied to a wide range of problems involving symbolic representations or complex data structures. However, the method can be severely hindered by the increased computational resources required and premature convergence caused by uncontrolled code growth. The algorithms, computer programs, as well as most mathematical or Boolean expressions may be represented in genetic programming, as individuals. These individuals, grouped in what is known as a population, have evolved iteratively. Each iteration being in connection with biology, called generation.

A solution can be encoded in different ways in an individual. In The literature, lists (Linear GP), graph (Cartesian GP) or machine code were used as representation to better adapt to different applications. However, the most common variant GP is undoubtedly the variant originally proposed by Koza. It's based on an encoding form of trees. An example of an individual is given in Figure 2.

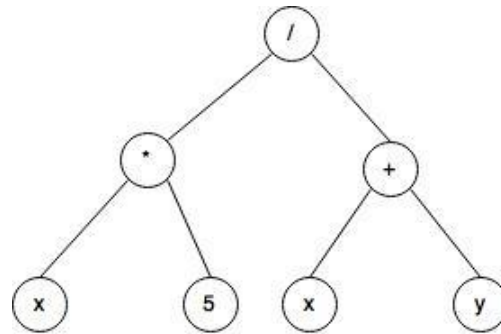


Figure 2. This Schema Represents the Mathematical Function
 $f(x) = (x * 5) / (x + y)$

Generally, genetic programming requires changing the definition of four main elements: a set of primitives, an evaluation method, or a variation operators and a heuristic selection.

Genetic programming allows computers to automatically solve a problem by starting a high-level instruction, in what it's necessary to do so. Specifically, genetic programming iteratively transforms a population of computer programs to a new generation of programs while applying the genetic operators inspiring from nature. The operators used in genetic programming process are: the crossover, mutation, reproduction, genetics duplication and selection.

3.2.2 Parameters of Genetic programming

In genetic programming, the population is a set of individuals. These individuals are a machine programs randomly chosen early in the process by combining a set of terminals with a set of functions. One of these programs is written in the form of a tree in a way to respect the order of the execution of its expression, it is written in its notion prefix to result the program tree.

In our method, each GP individual represents a candidate function of elementary distances, that is, a similarity combination function. This is encoded in a tree structure. Intern nodes contain arithmetic operators +, /, *, p, etc. Leaf nodes have similarity values d_i . Each d_i represents the similarity distance for a distinct descriptors. We recall that in this paper, we use two distinct descriptors (pyramid of BOC and pyramid of BOF).

The author [20] said that there are six preliminary steps to solve a problem using genetic programming. The terminals, functions, the fitness function, the control parameters and the stopping criterion.

The choice of program components (that is to say terminals and functions) and the choice of the fitness function largely determines the space which genetic programming, and therefore how difficult is the research, how successful will be obtained. It's guide the evolution of a population of genetic programming. In addition, fitness function not only gives a great reward to the correct solution, but it preferentially recognizes unproven solutions while running the GP, starting with the creation of the initial population until the discovery of the final solution. The parameter controls included the population size, rate crossing and mutation, the number of generation, selection method and is what the process is elitist or not, etc. The stopping criterion is simply a rule to stop the execution of GP. Typically, the rule to stop this, is to find a program that solves the problem or after a given number of generations. In the last step, the architecture of advanced programs is chosen.

3.2.3. General Algorithm of an Evolution in Genetic Programming

Algorithm 1 shows a very general example of what may be an evolution in GP.

As can be seen, the genetic programming requires very little knowledge on the problem studied: a function capable of evaluating the performance of an individual is enough. The stopping criterion mentioned can be of several types (stop when a satisfactory level of performance is achieved when the number of generations exceeds a certain threshold, *etc.*), according the required of the researcher.

Algorithm 1. Basic GP evolution algorithm for N generations.

- 1: Generate a initial population of individuals ("similarity trees")
 - 2: **for** N generations do
 - 3: Calculate the fitness of each individual
 - 4: Record the top N individuals and their fitness values
 - 5: Select the individuals to genetic operations
 - 6: Apply reproduction
 - 7: Apply crossover
 - 8: Apply mutation
 - 9: **end for**
 - 10: Apply the "best individual" recorded above on a query images set
-

According to the literature, everyone they agree that the genetic programming is more efficient in CBIR. For that, we have considered in our work. Despite its advantages, genetic programming suffers from the problem of complexity that affects the computation time. For this why, we have proposed in the next subsection an algorithm that reduces the computational time based on the multi-resolution BOF that we have presented in [7].

3.3 The proposed GP frameworks.

The present paper uses GP to combine simple metrics (descriptors). Each image is represented by a simple descriptor. For a given large image database and a given query image, the system retrieves a list of images from the database which are most "similar" to the query, according to a set of image features like shape, color, and/or texture.

The metrics (image descriptors) are combined using a mathematical expression uniquely represented as an expression tree, whose non-leaf nodes are numerical operators (see section 3.2.1) and the leaf node set is composed of the similarity values d_i , $i = 1, 2, \dots, k$. Each d_i represent a similarity value correspond with each descriptor type (color, shape and texture). Figure 3 shows a possible combination (obtained through the GP framework) of the similarity values d_1 and d_2 of two simple descriptors.

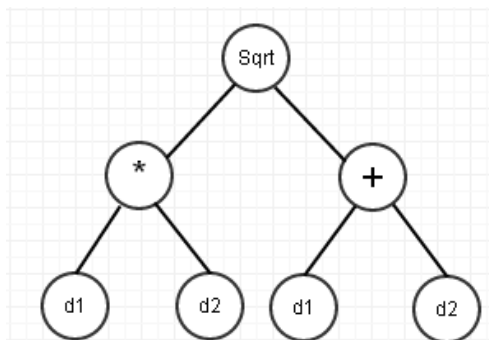


Figure 3. Example of GP based Similarity Function Represented in a Tree

In many areas, the system of problem resolution is based on a set of variables (data). Increasing the number of these variables (features) which model the problem introduces difficulties at many levels such as the complexity and the calculation time. For this reason, we propose in this paper a method to reduce the search time.

We adapt the technique proposed in [10] to our approach to improve the search time by using more than one descriptors. And for that, we proceed initially to use a signatures of smaller size to limit the set of candidate images, and each time we refine the results by using a more specific signatures that obviously became a signatures of big size: we recall that, each image is described by a pyramid of four histograms [7]; in the first time we use the histogram of the first level that are a small sizes, in the next time we refine the result by using the histograms correspond to the second level, we repeat this operation for the four histograms of each image signature. The algorithm of this approach is shown in the algorithm 2. Before this, we describe the definition of fitness function of GP adapted to the CBIR.

3.3.1 Fitness function

The fitness function plays a very important role in guiding GP to obtain the best solutions within a large search space. The fitness of an individual d_i is computed based on the similarity between the query pattern and all images from the training set. The fitness computation process is divided into two phases, described in the following. In the first phase, the training images are sorted according to their similarity with the query image Q , computed as follow:

Definition 1. (Combination multi-metric)

Let $M = \langle d_i \rangle$ be a vector of metric functions, with $|M| = m$ and d_i represents the L_2 distance between the query pattern and an image of the database according their i^{th} descriptor. $\forall i \in [1, m]$, the combination multi-metric (or combined metric functions) is a representation tree.

Example: In Figure 3, the combination is represented by: $d_1 * d_2 \sqrt{d_1 + d_2}$. Where d_1 and d_2 represent the L_2 distance between the query image and each image of the database according the color and shape descriptors. Once the ranking is obtained in the first phase, the second phase starts. The goal of this second phase is to evaluate the ranked list generated in Phase 1. This evaluation consists in assigning high values to ranked lists, in which relevant images present in the training set are ranked at the first positions. This evaluation is accomplished by applying an evaluation function that considers the rank position of the relevant images in the ranked list generated in phase 1. We recall, the aim of the fitness function is very important to guide GP to obtain the best combination and to find the best solutions. For this, the choice of the fitness function type is crucial, because a good fitness functions will help GP to explore the search space more effectively and efficiently. On the other hand, bad fitness functions can easily make GP get trapped in a local optimum solution and lose the discovery power.

In this paper we use the FFP1 function fitness proposed in [21].

The FFP₁ is defined as follow: $FFP_1 = \sum_{i=1}^{|M|} r(\hat{I}_i) \times k_1 \times \ln^{-1}(i + k_2)$, where

i is the image position after retrieval and \hat{I}_i is the image at position i . $r(\hat{I}_i) \in \{0, 1\}$ is the relevance score assigned to an image, being 1 if the image is relevant and 0 otherwise. $|N|$ is the total number of retrieved images. k_1 , k_2 are scaling factors. After exploratory analysis we set $k_1 = 10$ and $k_2 = 1.5$ in our experiments.

Another important parameter of GP that we are redefining is the stopping criterion that will be presented in the next paragraph.

3.3.2. The Stopping Criterion

The GP is terminated when all N training query images that are used, we have their majority similar images by an individual of the GP population or a predefined number

(M) of generations are completed. If the GP gives the high quality of all the N training query images, the best individual of the population gives the required optimum combination. Otherwise, when the GP is terminated after completion of M generations, the individual with the highest fitness measure is considered as the best individual from the population.

Table 1. Notations used in this Paper

Symbol	Definition
h_i	Histogram corresponds to i^{th} resolution (see section section 3).
P_{hi}	i^{th} histogram of object P.
$dbl_i(Q,P)$	i^{th} Lower bound distance: Euclidean distance between Q_{hi} and P_{hi}
$dist(Q_{hi}, P_{hi})$	Euclidean distance between Q_{hi} and P_{hi}
$dist(Q, P)$	Euclidean distance between Q_{h4} and P_{h4} (i.e. in this paper we have four resolutions, we notate also $P_{h4} = P_{hi}$).

The goal of our approach is to make the research by using several descriptors using genetic programming in high dimensional space and in a large database quickly.

The principal idea of our approach is to limit the number of images candidates that they have a big probability to be similar to the query image and ignores the images database that they have a small probability to be similar to the query image. This strategy is based on lower bound distances technique. Lemmas and describe this one.

Lemma 1: lower bound distance [10]

Given two objects (images) O1 and O2 represented by two Pyramid structures (pyramid of histograms) PO1 and PO2 respectively, with $PO1 = \{x1, y1\}$ and $PO2 = \{x2, y2\}$ where $y1$ and $y2$ are the lower resolutions of $x1$ and $x2$ respectively. A lower bound on the squared Euclidean distance between $y1$ and $y2$ denoted by $dist(y1,y2)$ can be represented as follows: $dist(x1, x2) \leq dist(y1, y2)$.

Proof:

Let $y1$ and $y2$ two n-dimensional vectors that represents the lower resolutions of two m-dimensional vectors $x1$ and $x2$ respectively, i.e.:

$$x_1[i] = \sum_{j=1}^m y_1[i][j] \quad \text{and} \quad x_2[i] = \sum_{j=1}^m y_2[i][j]$$

$$\begin{aligned} dist(x_1, x_2) &= \sum_{i=1}^n (x_1[i] - x_2[i])^2 \\ &= \sum_{i=1}^n \left(\sum_{j=1}^m y_1[i][j] - \sum_{j=1}^m y_2[i][j] \right)^2 \\ &\leq \sum_{i=1}^n \sum_{j=1}^m (y_1[i][j] - y_2[i][j])^2 \\ &= dist(y_1, y_2) \end{aligned}$$

So $dist(x1, x2)$ represents the lower bound of $dist(y1, y2)$.

Lemma 2. : Basic filtering

Let Q and P a two objects (images) represented by two pyramid structures [7], like $P = \{Ph1, Ph2, Ph3, Ph4\}$ and $Q = \{Qh1, Qh2, Qh3, Qh4\}$ where Ph_{i+1} and Qh_{i+1} are the lower resolutions for Ph and Qh respectively and R is a range search. If $dist(Qh_i, Ph_i) > R$ (with $i < 4$) then P automatically considered non relevant to Q and the distance $dist(Qh_{i+1}, Ph_{i+1})$ will not be calculated.

Proof:

By means of Lemma 1 we have:

$$\text{dist}(Qh_1, Ph_1) \leq \text{dist}(Qh_2, Ph_2) \leq \text{dist}(Qh_3, Ph_3) \leq \text{dist}(Qh_4, Ph_4).$$

Then if $\text{dist}(Qh_i, Ph_i) > R$ we have $\text{dist}(Qh_4, Ph_4) > R$.

From these two previous lemmas, we note that the images that are not found by using the lower histograms (small size) will not be found by using the histograms of the next resolution. However, in the first step we start the research by using the lower histograms. In the second step, we apply the histograms of the next level correspond to each pyramid of each image result obtained by the previous histograms to find from them those that are more similar to the query image; we repeat this operation for the four histograms of each pyramid of each image. We conclude that, by using this strategy we limit the region of research, hence the number of similarity measures become smaller. Therefore, by using our approach, we improve the search time without deteriorate the quality of result in term of precision. Figure 4 represents the schema of our approach.

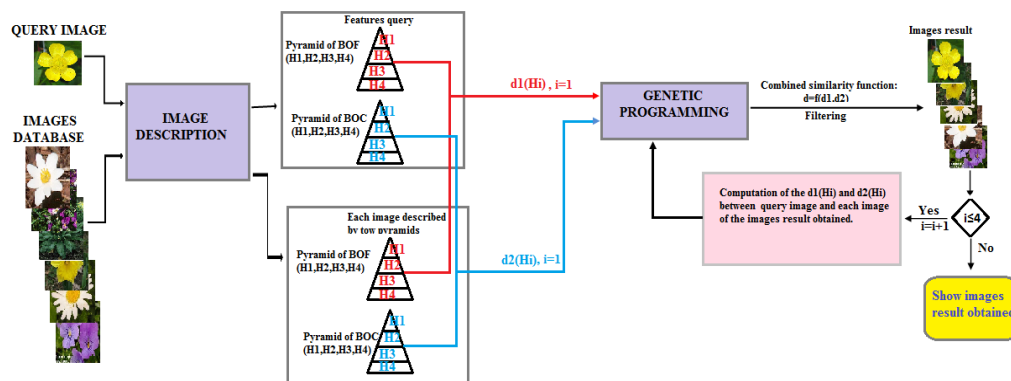


Figure 4. Schematic Diagram Representing our Approach

The pseudo-code in Algorithm 2 summarizes our nearest neighbor search algorithm using the multi-resolution GP.

Algorithm 2. The proposed GP-based process.

- 1: User indication of query image q that is described by 2 pyramid descriptors (pyramid of BOC and Pyramid of BOF, each one contain n histograms)
 - 2: $i=1$; Result=null; //initialize level by the first histogram in each pyramid
 - 3: **while** $i \leq n$ **do**
 - 4: Extract the i^{th} histogram for each pyramid for query and database images
 - 5: Apply GP to find the best individuals (similarity composition functions) - see Algorithm 1
 - 6: Rank the database images
 - 7: New database= the L most similar images
 - 8: $i=i+1$
 - 9: **end while**
 - 10: Result= the most similar images from new database.
-

4. Experiments

This section describes the experiments performed to validate our proposed frameworks.

4.1. Essential GP components

We chose, for genetic changes, the following parameters:

- Population: 200 programs;
- Maximum size of tree programs: 15 levels;
- Choice of operators +, -, *, /, sqrt;
- Crossover: 80% of genetic operations, tournament 5 individuals. The two strongest are "crusaders" and the two children will crushing the both weaker;
- Mutation (method 2: "whole sub-tree"): 25% of the genetic operations, tournament 5. The strongest is copied, the copy is mutated and replaces the lowest.
- Reproduction: copies the top rP trees in the current generation to the next. In our case, $r = 5\%$ and P is the population size.
- Stopping Criterion: obtaining more than 90% of accuracy on the test database, or after 25 generations.

4.2. Image Collections

To evaluate and validate the different approaches studied and developed in this work, three image databases are used, namely COLOMBIA database, WANG database and MirFlicker database. These bases form clear concepts, which each image can only belong to one class. The COLOMBIA database contains images of 3D objects taken from different positions. In this paper, the used collection of this database is COIL-100 [22]. It's consists of color images taken from 100 different objects, each category contains 72 images. The WANG database [23] contains 1000 images from the base COREL, that consists of exactly 1000 images classified in to 10 class, each calass contains 100 images. In the last data set, we use 50000 images from MirFlicker database [24], which consists of one million images that were extracted from the Flickr website (www.flickr.com). Three randomly chosen images were used as query images for each class of the both first databases and two for each class in the last database.

For each image query, a run of 25 generations is issued with our system. The average result is computed on the remaining runs for each category. Experiments are run five times and the average is presented. An image is considered relevant to the query if it is in the same category as the query.

4.3. Experimental Results

In this subsection, we present in the first part the experimental results of the proposed approach. We compare our method for image research by visual content (combination of several metrics) with the other of one metric in terms of accuracy, and in the second part we show the results of our approach in term of search time.

We set up our two pyramid descriptors BOC and BOF pyramid. The first obtained by quantifying colors thumbnails [25] and the second obtained by quantifying the SURF descriptors.

The implementation is done in Java. Performance evaluation is made in a quad core of an Intel processor running at 1.6 GHz PC with 10 GB memory, Windows 7 operating system and 400 GB of local disk.

In the research process, we set a query image and comparing its BOF and its BOC histograms with other BOF and BOC histograms of the images database, we obtained a set of sorted images according their similarity using the Euclidean distance as a similarity measure.

4.3.1. Experimental Results

We propose in this section to assess the research performance by the visual content based on our method. We assess the interest of the combination of different types of descriptors (metrics) of image content compared to use just one descriptor type.

The graphics that are drawn based on the average values obtained for the query images for each database. We use precision vs. recall figures, a standard evaluation technique for retrieval systems [54] for comparing the effectiveness of our algorithms.

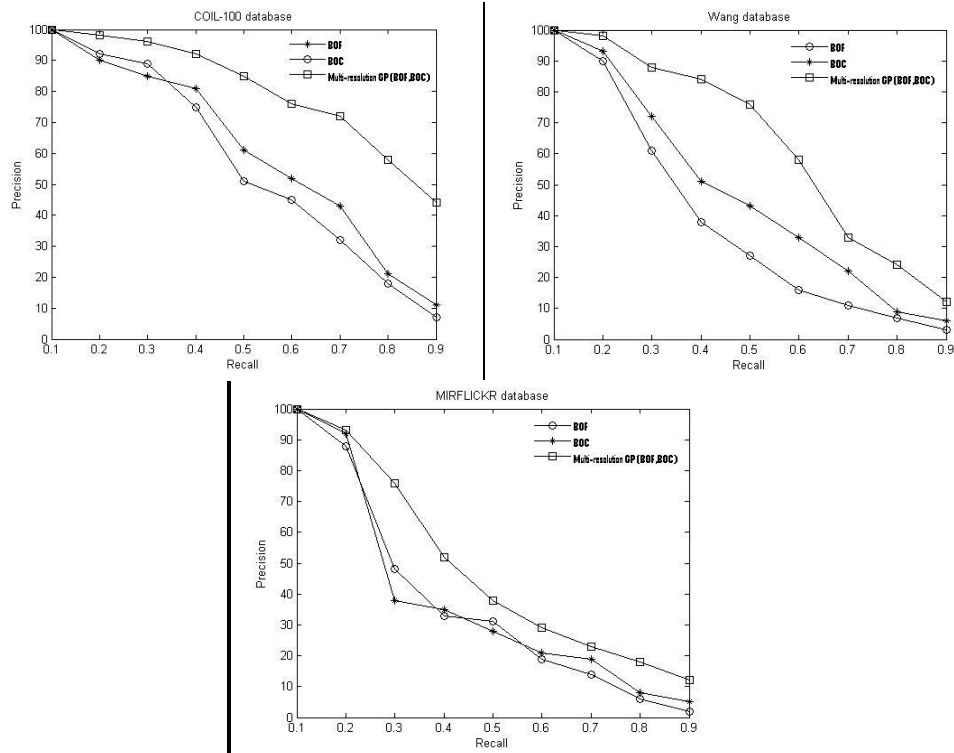


Figure 5. Precision versus recall curves of the BOF descriptor, BOC descriptor, and the GP (BOF and BOC)

Precision / recall curves that we are presented in Figures 5 are obtained for three image databases. From these curves, we noticed that the performance of the combination by genetic programming of the descriptors cited in the screen is always better compared to a single descriptor type.

The graphics of Figure 6 show a results example of similarity research of our implementation. The images results obtained correspond to the similar images of the query image. They are sorted and displayed from left to right and top to bottom according to their score of query image. The results of these examples show that, the research by using the combination (our approach) of two descriptors is more accurate to using a single descriptor at a time.

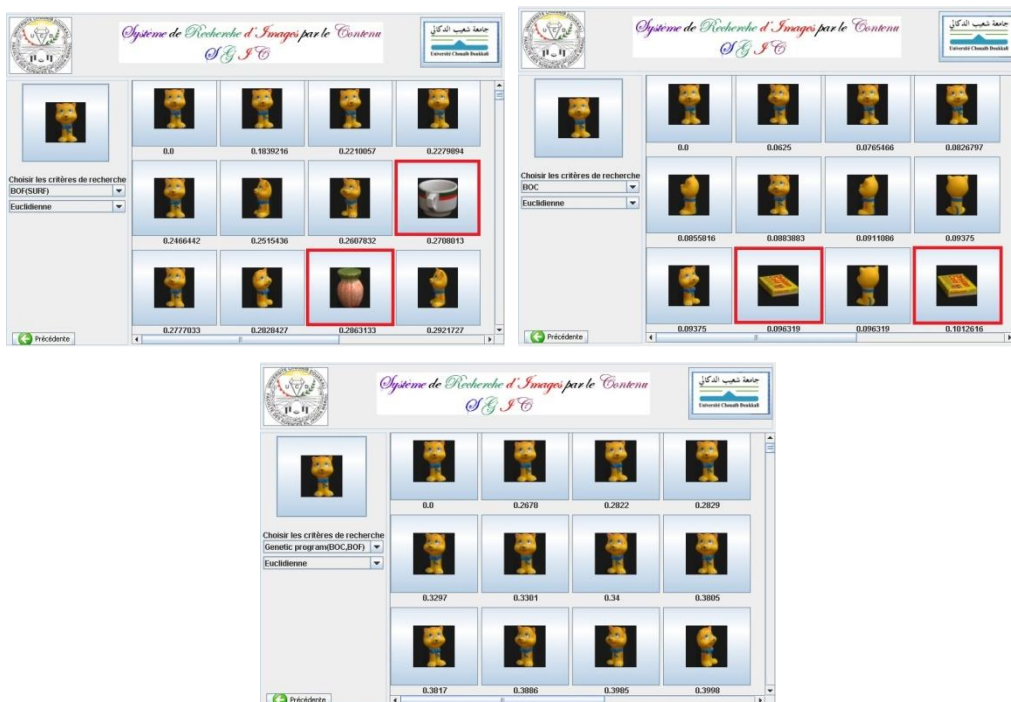


Figure 6. An Example of Image Research of our Implementation by using Color (BOC), Shape (BOF) Descriptors and Combination of them using Genetic Programming. Note that the Images Framed by Red Frames are not considered similar to the Query Image and below the Image Result Values Represent their Scores (here the Euclidean distance) to the Query Image

4.4. Search Time Evaluation

The goal here is to study the quality of our method in terms of search time. For this, we evaluate our method on a large database. We compare our method with classical GP method.

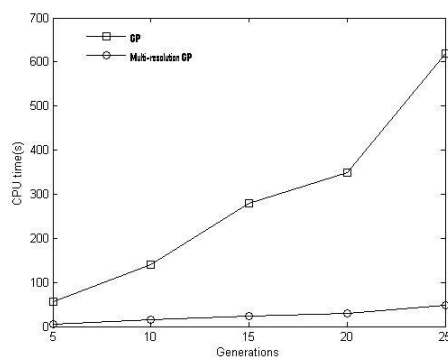


Figure 7. CPU Times at Five Different Generations (for a population size 200.)

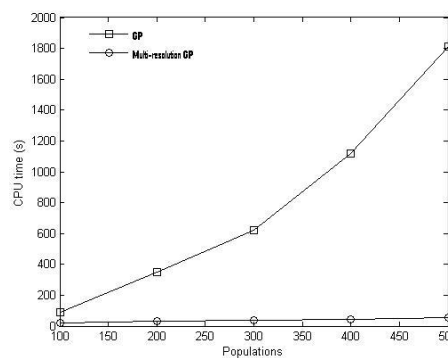


Figure 8. CPU Times at Nine Different Population Sizes (for 20 generations.)

The results are illustrated in Figures 7 and 8. These show that the response time of our approach slightly changes according to the number of generation as well as the population size in comparison to conventional GP for the same criteria. Thus, the response time obtained by this method for a population size 200 and 20 generations is 350s, whereas for

the same generation, we get 30s for classical GP, whether an acceleration factor of 11 compared to classical GP. This shows that our approach improves the response time compared to other methods.

5. Conclusion

The advantage of genetic programming is to be applicable to a wide range of problems such as CBIR, no special features. In addition, they are able to work on any area of research, continuous, discrete, mixed, tree space, etc. Against by, the calculation time of their functions can be cost a barrier to their use. We can conclude that the method that we have presented in this paper is feel fulfilled well task for decreasing the search time.

One possibility to improve more the search time is to combine our approach with the distribution of calculations on several processes, which is conceptually very easy. This technique we will study in the future work.

References

- [1] A. Sheppard, "Techniques in helical scanning, dynamic imaging and image segmentation for improved quantitative analysis with X-ray micro-CT", Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, vol. 324 (2014), pp. 49-56.
- [2] T. Peleg and M. Elad, "A statistical prediction model based on sparse representations for single image super-resolution", IEEE transactions on image processing: a publication of the IEEE Signal Processing Society, vol. 23, no. 6, (2014), pp. 2569-2582.
- [3] V. Gouet-Brunet, Chapitre de livre " Recherche par contenu visuel dans les grandes collections d'images ", Encyclopedie de l'Informatique et des systmes d'information, J. Akoka and I. Commyn-Wattiau (eds.), Vuibert, (2006) ,564-576.
- [4] H. Aanas, A. L. Dahl and K.S. Pedersen, "Interesting Interest Points: A Comparative Study of Interest Point Performance on a Unique Data Set", International Journal of Computer Vision, 97(1), (2012) pages 18-35.
- [5] F. Long, H. Zhang, D. Feng, "Multimedia Information Retrieval and Management. Technological Fundamentals and Applications". Springer-Verlag, Berlin, Heidelberg, New York. Ch. Fundamentals of content based image retrieval, (2003), pp. 1-26.
- [6] J. Sivic and A. Zisserman, "Video google : A text retrieval approach to object matching in videos". (2003), Pages 1470-1477, Nice, France.
- [7] A. Bahri, H. Zouaki, Y. Bourass, "Robust Method for Codebook Generation Using Embedding Algorithm for Large Image Collections". International Journal of Emerging Sciences, vol 5, issue 1, (2015), pp 37-48.
- [8] J. R. Koza, "Genetic programming : on the programming of computers by means of natural selection". MIT press, (1992).
- [9] C. Beecks, M. S. Uysal and T. Seidl, "Content-Based Image Retrieval with Gaussian Mixture Models", MultiMedia Modeling, Springer International Publishing, (2015).
- [10] A. Bahri, H. Zouaki, R. Oulad Haj Thami, "BLBTree : An Efficient Index Structure For Fast Search". The International Review on Computers and Software (IRECOS), vol 11 (10), (2016).
- [11] J. Law-To, V. Gouet-Brunet, O. Buisson and N. Boujemaa, "ViCop T : a robust system for content based video copy detection in large databases", ACM Multimedia Systems Journal, 15(6), (2009),337-353.
- [12] V. Gouet-Brunet and B. Lameyre, "Object recognition and segmentation in videos by connecting heterogeneous visual features", Computer Vision and Image Understanding Journal, 111(1) (2008), 86-109.
- [13] P. Gehler and S. Nowozin, "On Feature Combination for Multiclass Object Classification", Proceedings of the Twelfth IEEE International Conference on Computer Vision (ICCV'09), (2009), pages 221-228.
- [14] W. Fei, H. Yahong, T. Qi and Z. Yueting, (Multi-label boosting for image annotation by structural grouping sparsity), Proceedings of the international conference on Multimedia (MM'10), (2010), pages 15-24.
- [15] M. Fernandez, D. Vallet, P. Castells, "Probabilistic score normalization for rank aggregation". NCS/Advances in Information Retrieval Springer, Berlin, vol. 3936. (2006), pp. 553-556.
- [16] J. Knowles, D. Corne, "Approximating the nondominated front using the pareto archived evolution strategy". Evolutionary Computat. 8 (2), (2000), 149-172.
- [17] Q. Zhang, E. Izquierdo, "Optimizing metrics combining low-level visual descriptors for image annotation and retrieval". In: Proceeding of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP06), Toulouse, France, (2006), pp. 405-408.

- [18] M. Arevalillo-Herrez, J. Domingo and J.F. Francesc, "Combining similarity measures in content-based image retrieval". *Pattern Recognition Letters* 29 (2008), 2174-2181.
- [19] H. M. de Almeida, M. Gon, M. C. Acalves and P. Calado, "A Combined Component Approach for Finding Collection-adapted Ranking Functions based on Genetic Programming", in: *Proceedings of the ACM SIGIR*, (2007).
- [20] J. R. Koza, "Genetic programming II: automatic discovery of reusable programs", MIT press, Cambridge, Massachusetts; Mai (1994).
- [21] W. Fan, E.A. Fox, P. Pathak and H Wu. "The effects of fitness functions on genetic programming-based ranking discovery for web search", *JASIST* 55 (7) (2004), 628-636.
- [22] S. A. Nene, S. K. Nayar and H. Murase. "Columbia object image library (coil-100)", Technical Report CUCS-006-96, (1996).
- [23] J. Z. Wang, J. Li and G. Wiederhold, "Automatic linguistic indexing of pictures by a statistical modeling approach", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, (2001), 947-963.
- [24] M. J. Huiskes, B. Thomee and M. S. Lew, "New trends and ideas in visual concept detection: the MIR flickr retrieval evaluation initiative", in *MIR 10: Proceedings of the 2010 ACM International Conference on Multimedia Information Retrieval*, ACM Press, New York, (2010), pp. 527-536.
- [25] T. Hurtut, Y. Gousseau and F. Schmitt. "Adaptive image retrieval based on the spatial organization of colors", *Computer Vision and Image Understanding (CVIU)*, 111(2), (2008), pp. 101-113.

Authors



Abdelkhalak Bahri: He received his Ph.D. degree from the Department of Mathematics and Computer Science at the University Chouaib Doukkali, Faculty of Sciences, El Jadida, Morocco in 2014. My research interests focus on CBIR, image description, optimization and multimedia database.



Hamid Zouaki received his PhD. In Mathematics from the institute of Applied Mathematics of Grenoble, France, 1991. He is currently Professor at the department of Mathematics of the University Chouaib Doukkali, El Jadida (Morocco). His research interests are representation tools for shape description, convex geometry, image analysis and optimization.

