Queries by visual content using fuzzy similarity measures on regional descriptions

Jean-François Omhover and Marcin Detyniecki
LIP6 - Université P. et M. Curie & CNRS
8 rue du Capitaine Scott - 75015 Paris France
Phone: +33-1-44278803, Fax: +33-1-44277000
email:{FirstName.LastName}@lip6.fr

ABSTRACT: This paper introduces a new CBIR system based on segmented images and fuzzy similarity measures. It gives the possibility to the user to specify a set of regions, and to retrieve images containing these regions in an approximate spatial configuration. It also supports competitive queries, and allows the expert-user to manipulate the parameters of the functions involved. In this manuscript we focus in the architectural aspects of the system.

KEYWORDS: Image Retrieval, Segmentation, Fuzzy Similarity Measures, Aggregation.

INTRODUCTION

Content-based image retrieval (CBIR) aims to fulfil the need of finding a specific content in an image database. A CBIR system should be able to return images to some user based on the expression of his semantic interest. Unfortunately, this simple goal has not yet been achieved [6].

To avoid the use of textual annotations, which are subjective and highly consuming in terms of human work charge, the community has focused on the automation of the extraction and the comparison of image features. The query is then expressed not as some textual description, but as an image, chosen by the user, and then is compared to the entries of the image database. Such a comparison is based on a given set of features extracted from the images, called signatures. Two signatures extracted from two different images can then be compared by a similarity function (or distance) of this feature space. Many different feature spaces have been developed to compute the similarities involved in those queries and the corresponding systems developed (among the most famous we have [12],[13]).

At the same time, in the last few years, segmentation has been used as a tool of image processing. Images can be segmented in regions, roughly corresponding to objects in the image, thus reflecting the objective content of an image. For the moment, only a few image retrieval systems follow this approach. Blobworld [1] proposes its segmentation tool to the user, and it is him who specifies the request by selecting a set of regions of interest. The retrieval then takes place by comparing pairs of regions. The returned images are the one for which regions obtained the best scores in one-to-one region similarities computation. Another system called Simplicity [2][7] extends this approach to image-to-image comparison. It compares images globally by aggregating region-to-region similarities. Its scheme does not include the spatial configuration of the regions, though it tries to match regions of the request to regions of each image.

Here we introduce a new CBIR system based on fuzzy similarity measures, which can be aggregated to support composite regional queries. It gives the possibility to the user to specify a set of regions, and to retrieve images containing these regions in an approximate spatial configuration. It supports competitive queries, and lets the expert-user manipulate the parameters of the functions involved.

In this paper, although we present an overview of our approach, we focus on the architecture of the system. The software has been designed so that it is open (for instance to new similarity measures), and highly parametrable (see section on Architecture); its interface is also a practical tool for the formulation of complex queries using several regions of interest (see section on Queries Examples). These complex queries are evaluated using different kind of visual similarity measures: one classical global measure as reference, and a similarity that compares regions one-to-one (see section on Segmented-Image Signature). This last similarity is based upon three measures based on the colour, shape and position of the regions, which we briefly describe. We finish with an illustration of our approach with some simple queries in the evaluation section.

If the reader is interested in particular in the features used and their manipulation (queries, aggregation) please refer to [14] and to [15] if he is rather interested in the use of fuzzy similarities measures and its consequence in the final result. The reader can also visit the following webpage in order to test the system: http://strict.lip6.fr/

STRICT IMAGE RETRIEVAL PLATFORM

We developed a complete image retrieval system from scratch. This system has been designed to let an experimented user handle different methods of the CBIR field. The underlying scheme of our architecture lets an expert build complex queries (see section on query example to see the range of possibilities). In this particular section, we focus on the architecture of our system, the information contained in its queries, and we finish by pointing out some types of meaningful requests.

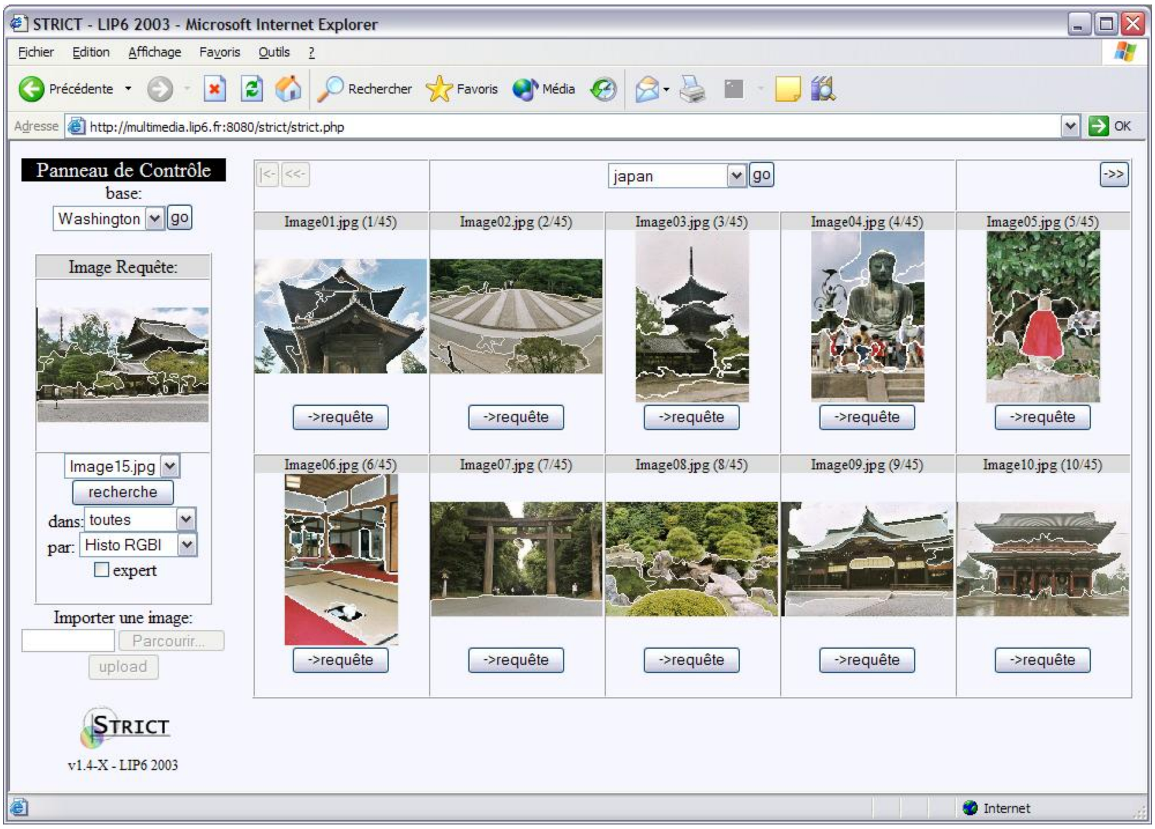

Figure 1: STRICT interface.

ARCHITECTURE

Our architecture is composed of two distinct systems. First, we have our own indexation server that keeps the signature database and computes similarities on demand. Second, there is an HTTP server that runs the interface, serves the images to the user and presents the results of his requests.

The indexation server remains invisible to the user. It maintains a database of vectors extracted from the set of images. These vectors are the signatures of each image. The server also proposes a register of similarity measures. These measures are based on different types of signatures. The measures can have parameters; therefore our server keeps a list with the type, the possible values, and a description of each parameter. And all this can modify on-the-fly during the request.

The indexation server receives the queries from the interface. One query contains all the information needed by the server to compute the result. A request can be composed of an image, the target of the comparison (the whole image database or a part of it), and the names of the comparison measures choose by the user, with their optional parameters. The system returns a list of the images that are found most similar to the request in the sense of the measures specified. The syntax of this information is roughly explained in the section on the query examples.

The visible part of the system is its web interface. Its dynamic format enables the user to browse the image database, formulate his queries and compare the results using different similarity measures (see Figure 1). When a user opens a new connection to the interface, the description of each measure is sent from the indexation server to the web server. Based on this list, the web server generates a form that allows the user to modify the parameters. Thus the interface has not to be modified to cope with the modifications of the comparison measures or the addition of new parameters within the invisible server.

From the interface, the user can then compose the request using several measures with different parameters. Each measure of the request, when received by the indexation server, is then run against every image of the database. When the resulting list of images is displayed, each image is marked by the intensity of each measure specified. This feature enables the user to compare the effects of its parameters on the retrieval.

The labels of the measures involved are not the only information sent in the query. The user can formulate an aggregation operator using basic mathematical operations (see the next section). And then the system will aggregate the results on-the-fly. Using the minimum, the maximum, the addition and the product, he can define his own score post-processing operator. He specifies the entries using the available similarity measures, and then tests this newly created similarity measure against the images. The basic operations programmed in this application let us build most of the known aggregation operators [4]. The utility of this scheme is illustrated in the section on the query examples.

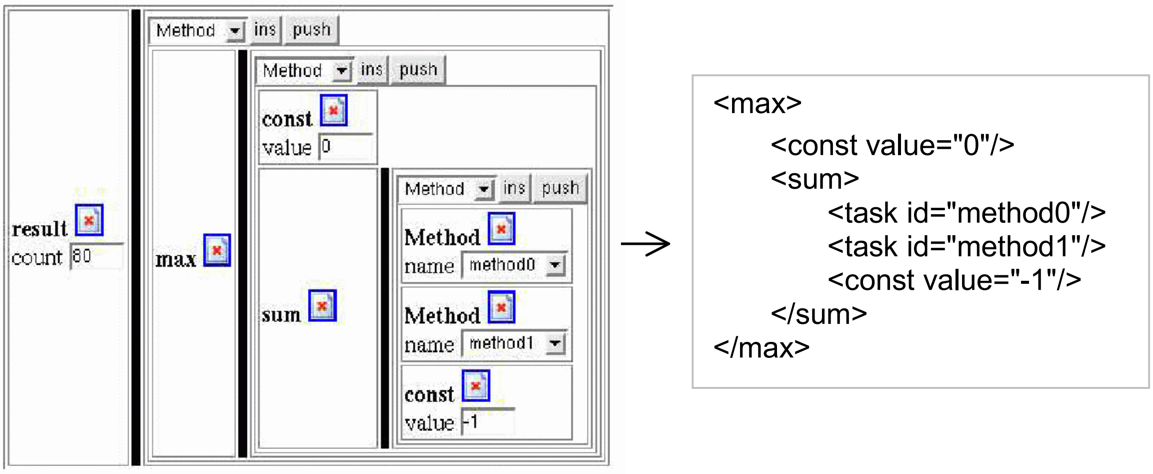

Figure 2: Lukasiewicz t-norm [$(x,y) \rightarrow \max(x+y-1,0)$] built by simple clicks on STRICT web interface.

QUERIES TO THE INDEXATION SERVER

The indexation server of STRICT receives its queries from the web interface in XML format. All the requests contain the same information. By specifying the images of interest, the parameters of the comparison measures, and some aggregation operators, the user can formulate many different types of visual queries. Here, we describe the different parts of a request:

Image inputs: The user specifies the inputs of his query by selecting list of images. He can also choose to make his interest more explicit by specifying a set of regions for each image. We developed region-matching similarity measures (i.e. not global image comparisons), for which the specification of a region is mandatory.

Target database: The query can be run on the whole database, or just on a specific part of it.

Comparison measures: The list of the comparison measures follows. Among the available measures listed in the register of the indexation server, the user can choose several measures. The user can choose to simply get the result of each comparison measure, or to aggregate their results. He can also modify some of their parameters. This high interaction allows us to test different aspects of the visual similarity, like global versus regional matching to evaluate their respective efficiency, or to set different values of parameters for a same measure to find the best configuration of this measure.

Score post-processing: STRICT can operate modifications or aggregations to the similarity values. For example if the user wants to use two different measures, one focused on colour and the other on texture, he can use an averaging operator to build a mixed score representing both colour and texture similarity. He can also insert weights to balance the importance of one or the other. Practically speaking, for each measure (which has been specified in the comparison measures' list within the query), the system returns a score from the comparison of the input with each entry of the image database. Then a post-processing operator can be applied to aggregate these evaluated scores. This operator is specified within the query, received and computed by the invisible server. The value returned by this operator is then returned to the user.

Building such an operator is an easy task using the interface. It is based on the following basic operators: minimum, maximum, sum, product, inversion, fuzzy negation ($x \rightarrow 1-x$), or averaging. A trivial operator has been added to return a constant value. From these operations, one can build a large variety of fusion operators. In Figure 2 we show the Lukasiewicz t-norm built using our interface.

In the queries, our operators are expressed as an XML tree: the nodes are the labels of the basic mathematical operations; the leaves are the names of the measures specified above (or constants).

INTERESTING QUERY EXAMPLES

STRICT can compute global and regional similarities. For the moment, the system is based on one-to-one comparison functions. For instance for a single (global) image request R, a similarity measure S is run against every entry I of our image database. The value $S(R,I)$ of the comparison is computed and returned for post-processing.

In the case of a region request, a regional similarity measure S_{reg} is used to find in every I a region that is similar to a single region $R(i)$ of image R. The returned score is noted $S_{reg}(R(i),I)$, which corresponds to the truth value of the proposition *" there exists a region like $R(i)$ in I "*.

Those scores, corresponding to similarity measures, can be aggregated on-the-fly under users' command. An operator, noted *Agg*, is used to produce a single value $f(I)$ from the n different scores involved in the query. This feature enables the user to build different kinds of requests. In the following, we present four interesting query types:

- **Single Image, multiple criteria:** Using a single image request R, the user can compare the image entries using multiple features (colour, texture...). STRICT then aggregates the scores resulting from the different similarity measures. The final score $f(I)$ of the image I is then expressed as:

$$f(I) = Agg(S_1(R,I), ..., S_n(R,I)) \tag{1}$$

where *Agg* can be the averaging operator, the minimum, the maximum, or any constructible operator, depending on the purpose of the request.

- **Multiple Images:** The user can also specify a multiple image requests. The idea is to retrieve the images, which are similar to all of the requests (some of them might be more important than others). With $R_1,...,R_n$ a set of request images, STRICT will compute the global similarity between each image I of the database and each R_i. It will then aggregate these similarities to render a final score $f(I)$ for each I.

$$f(I) = Agg(S(R_1, I), ..., S(R_n, I)) \tag{2}$$

The aggregation used here can also reflect the importance given by the user to each image request, by an optional weighting.

- **Multiple Region from one Image:** Based on a single image request R, the user can select different regions $R(i),...,R(j)$ that he would like to find in one (or several) images of the database. The final score $f(I)$ is then expressed by:

$$f(I) = Agg(S_{reg.}(R(i), I), ..., S_{reg.}(R(j), I)) \tag{3}$$

Attention must be paid to the fact that each score $S_{reg}(R(i),I)$ is a global one: the aggregated score does not indicate if the set of regions in the image I is formed of different regions, they just individually look like *one* of the regions $R(i)$.

- **Multiple Region from multiple Images:** Obviously, the previous scheme can also be applied on a request composed of regions extracted from different images. $f(I)$ then takes the following form:

$$f(I) = Agg(S_{reg.}(R_1(i_1), I), ..., S_{reg.}(R_n(i_n), I)) \tag{4}$$

The dynamic interface of our system is a practical tool to test all of these four schemes. With simple actions, an expert user can simulate one of these four requests, specify its parameters, and compose an aggregator to fulfil its needs. No other current CBIR system offers all of these features.

SEGMENTED IMAGE SIGNATURES AND SIMILARITIES

We have designed descriptors and their corresponding measures to form a homogeneous set of measures. All the measures programmed in our platform return values in [0,1], so that they can be aggregated and weighted in a scheme independent of the descriptors themselves. In this part, we shortly present the available features and similarities for more details please refer to [14].

As we already said we work with segmented images. We use a proprietary algorithm that gives good results in a very short computation time [5]. Basically, from an image we extract regions complying to a given homogeneity criteria. This criterion is usually based on colours: the algorithm tends to isolate regions of connected pixels, which present similar colours. Those regions roughly correspond to the objects present in each image. Now, our hypothesis is that a similarity measure based on those regions can reflect an objective similarity of the content. Therefore, we developed a similarity measure for regions based on the similarities of colour, shape, and position.

FUZZY SIMILARITY MEASURES

As expressed in [8], the similarity between two fuzzy sets can be calculated by a function S, with three variables:

- $M(A \cap B)$ the area of the fuzzy intersection of A and B, which measures the common features of fuzzy sets A and B.
- $M(A\text{-}B)$, the area of the fuzzy difference of A by B, which measures the features that are only present in A.
- $M(B\text{-}A)$, the area of the fuzzy difference of B by A, which measures the features that are only present in B.

Applied to global image histograms, these areas would correspond to the following:

$$\begin{aligned} M(A \cap B) &= \textstyle\sum_{C_i} min(H_A(C_i), H_B(C_i)), \\ M(A - B) &= \textstyle\sum_{C_i} max(H_A(C_i) - H_B(C_i), 0), \\ M(B - A) &= \textstyle\sum_{C_i} max(H_B(C_i) - H_A(C_i), 0) \end{aligned} \tag{5}$$

This expression of fuzzy similarity measures falls within Twersky's contrast model [16], a psychological framework for similarity measures. As shown in [10] these measures provide an intuitive measurement of similarity. They are also independent on the scale of the fuzzy sets. Based on this general framework, different particular measures were derived. We chose to implement only four of them: Jaccard, Dice, Ochiai and Fermi-Dirac similarity measures.

REGION COLOUR, SHAPE AND POSITION

The resemblance between regions is more elaborated than the global one. We have to consider several aspects of comparison, that are colour, shape and position. For each of these comparisons, a region is represented as a vector. Because these three vectors belong to different spaces, and cannot be related one to the other, the similarity between pairs of region is based on three different measures: one measure for each of the vector pairs. For two regions $R(i)$ and $I(j)$, extracted respectively from the image request R and from an image database entry I, the region similarity S_{reg} can be written as an aggregation of these three "sub-measures".

Fuzzy Region Colour Similarity

The region colour similarity $S_{reg|color}$ is based on the histograms of the regions. We extract the colour distribution of each region from each image. To compare the regional colour histograms, we use similar approach as for global case (the same measure adapted to the region, see equation (5)).

Fuzzy Proximity Measure

The proximity measure $D_{reg|position}$ is not a resemblance measure [8]. It simply evaluates the proximity of the centre of two regions. We define a fuzzy set D_{near}, which represents the set of distance values considered as "near".

$D_{reg|position}(R(i),I(j))$ simply returns the membership to the fuzzy set near. Notice that the coordinates of each centre are normalised by the width and height of the image they belong to.

Fuzzy Shape Similarity

The similarity of the shapes of two regions relies on the comparison of their minimum bounding rectangles (MBR). These are rough representation of shape, but they are easy to implement and compare. The MBR of a region is the rectangle that minimally covers the region and which borders are parallel to the borders of the image.

To compare the two shapes of regions $R(i)$, $I(j)$, we centre their minimum bounding rectangles on a common point, and compute the common surface and their different surfaces. A resemblance measure is then directly applied. At the time we prepared this article, the MBR comparison was the only shape description we implemented. It is a trivial scheme with reduced of accuracy. We are currently working on other implementations of shape representation, in particular by shape invariants.

PRELIMINARY EVALUATION

In order to illustrate the performance of our system, we present here five visual requests (see Figure 3): each time, 2 or 3 regions of interest were selected, then used as a query, using the default parameters. We compared the efficiency of this aggregated method to the efficiency of the classical global histogram approach.

image1 image2 image3 image4 image5

Figure 3: Images used as requests to evaluate the approach.

The system retrieves the 80 best results for both methods amongst a database of 7700 images. This operation takes about 1.2 seconds. We evaluate the effectiveness of both methods by measuring two values:

- the rank R of the first non pertinent image in the list.
- the proportion P of pertinent images in the first 10 results (extended to 20 or 30 when the first non-pertinent image was not present in the 10 first).

For a method to be efficient, it has to maximise the rank of the first non-pertinent image, and also maximise the proportion of pertinent images in the first results.

On table 1, we observe that our method based on region matching gets better results than the global histogram comparison, though a larger benchmarking should be run in order to be able to estimate correctly the efficiency.

	Segmented		**Global Histo.**	
	P	R	P	R
image1	19/20	19	7/20	5
image2	22/30	19	24/30	21
image3	4/10	5	4/10	3
image4	5/10	3	1/10	2
image5	4/10	4	4/10	2
average results				
	0.59	10	0.41	6.6

Table I:. Evaluation Results

CONCLUSION

The architecture of our image retrieval has been designed to handle composite regional queries. Our system uses a regional representation of images, built upon a segmentation algorithm. This solution, currently investigated in the community, has been implemented in STRICT to let the user specify his visual interest in his request. For the expert, our system proposes features that enable him to modify parameters, run complex requests, and compare their results in an dynamic and easy to use interface.

The efficiency of regional representation has been tested against the classical global representation, and proven to be more accurate. Although it still needs development, in the extraction of different image features, our system proposes a scheme relying on fuzzy resemblance measures and aggregation operators that reflects the similarity between visual objects, and is open to intuitive parameterisation.

REFERENCES

[1] C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. In Third Int. Conf. on Visual Information Systems, Amsterdam, 1999.

[2] Y. Chen and J.Z. Wang. A region-based fuzzy feature matching approach to content-based image indexing and retrieval. IEEE Trans. on Pattern Analysis and Machine Intelligence, 24(9), September 2002.

[3] H.D. Cheng, X.H. Jiang, Y. Sun, and J. Wang. Color image segmentation: advances and prospects. Pattern Recognition, 34:2259-2281, 2001.

[4] M. Detyniecki. Mathematical Aggregation Operators and their Application to Video Querying. PhD thesis, Université Pierre et Marie Curie, 2000.

[5] G. Durand and P. Faudemay. A fast region-merging segmentation algorithm for video analysis and indexing. In CIR 98 Symposium and Workshop on Image Retrieval, Newcastle, 1998.

[6] J.P. Eakins. Towards intelligent image retrieval. Pattern Recognition, 35:3-14, 2002.

[7] J. Li, J.Z. Wang, and G. Wiederhold. Irm: Integrated region matching for image retrieval. In 8th ACM Int. Conf. on Multimedia, pages 147-156, 2000.

[8] M. Rifqi. Mesures de comparaison, typicalité et classification d'objets flous : théorie et pratique. PhD thesis, Université Paris 6, December 1996.

[9] M. Rifqi, M. Detyniecki, and B. Bouchon-Meunier. Discrimination power of measures of resemblance. In IFSA'03, Istanbul, Turkey, 2003.

[10] S. Santini and R. Jain. Similarity measures. IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(9) : 871-883, 1999.

[11] M.J. Swain and D.H. Ballard. Color indexing. International Journal of Computer Vision, 7(1) : 11-32, 1991.

[12] Flickner, M.D. and Sawhney, H. and Niblack, W. and Ashley, J. and Huang, Q. and Dom, B. and Gorkani, M. and Hafner, J. and Lee, D. and Petkovic, D. and Steele, D. and Yanker, P. Query by Image and Video Content: The QBIC System. Computer 28 (9) : 23-32, 1995.

[13] J.R. Smith and S.F. Chang. VisualSEEk: a fully automated content-based image query system. In ACM Multimedia '96. November, 1996

[14] J.F. Omhover, M. Detyniecki and B. Bouchon-Meunier. A Region-Similarity-Based Image Retrieval System. In IPMU'04, Perugia, Italy, July 2004

[15] J.F. Omhover, M. Detyniecki and M. Rifqi. Image Retrieval using Fuzzy Similarity: Measure Equivalence based on Invariance in Ranking. In IEEE International Conference on Fuzzy Systems Fuzz-IEEE, Budapest, Hungary, July 2004

[16] A. Tversky. Features of similarity. Psychological Review, 84: 327-352, 1977.