Color Coarse Segmentation and Regions Selection For Similar Images Retrieval

Jérôme Da Rugna and Hubert Konik
{darugna,konik}@vision.univ-st-etienne.fr
Equipe Ingénierie de la Vision - Laboratoire LIGIV EA 3070
Université Jean Monnet de Saint-Etienne
3, rue Javelin Pagnon – BP 505 42007 Saint-Etienne Cédex 01, France

Introduction

With the growth of large image databases, content-based image retrieval systems are actually a highly challenging problem. The common approach is to extract a signature for every image based on different features (texture, color, shape analysis …) and to minimize a distance for retrieving similar images to a request one. Then, features extraction becomes the most important theme objectively, a large panel of systems[4,12] and methods exist, based on statistical features,[3] visual parameters, color histograms,[10] region-based search[8]… The main attention must be paid to develop insensitive features to intensity variation, scaling, rotations or else compression effects.

Finally, we will develop the solution* to extract some numerical features for every image before achieving with the presentation of our content-based retrieval system called ***i***COBRA.** The efficiency of this method will be illustrated on a large classical color images database, composed notably by goodshoot©images, containing very diverses images with a high rate of jpeg compression.

Color Segmentation

In order to identify automatically regions of interest in an image, two approaches are generally useful in image indexation: first to extract "points of interest",[9,5,7] secondly to achieve a segmentation into homogeneous regions. Objectively, the most popular way to obtain a description of each region color and texture characteristics is produced with unsupervised segmentation methods.[2] In fact, it seems always more difficult to estimate these local informations around different points of interest, notably because there are principally corners, where the homogeneity is then badly at fault.

Expected Results

But, it is well-known that segmentation algorithms are not robust and more or less always adapted to not universal problem. Objectively, it is absolutely illusive to apply one method, with a priori parameters as thresholds or else homogeneity predicates, to a large domain and in different contexts. However, retrieving images from a large and varied collections using image content as a key does not suppose to necessary develop a method that describes each region perfectly, but only a soft one, where the major objects are coarsely determined. All the more as our approach is to combine this first step with a post selection one.

From now on, let then assume that the segmentation results we are looking for have to be generally "good" but can involve some "errors" from a perceptual point of view. In fact, even if the results seem to be not perfect, the cost and the risk to merge different regions together wrongfully is not necessary integrating the post-processing. For example, our idea is not to extract the flower presented in figure 1 as only one region but to create different homogeneous regions from a color point of view as the semantic information will be set up again during the last step of our method. Moreover, even if the "quite-red" regions have to be merged in this case, it is perhaps not always the case considering large and extremely varied collections…

Segmentation Algorithm

Then, our work is first to propose a new low cost and basic segmentation, based on a color gaussian pyramid, particularly well-adapted in an image indexation context, as it simulates the human vision in its attention focusing,[11] through an individual and contextual analysis of each region. The basic idea of the pyramid structure is to produce a stack of interrelated images with progressively reduced resolution. Taking into account both spatial and color information, we construct the pyramid in a gamma-corrected RGB space, where the mixing is additive.[6] More precisely, our method simulates the human visual system miming the focus-of-attention principle, assuming that there is an optimal resolution for the problem. The main principle of our segmentation is a bottom-up process linking the different pyramid levels, while the lower resolutions provide a global view of the image, and the higher provide local information that are necessary in the human visual process of seeing an image. Although most of pyramid algorithms have been reported to be successful in a large number of fields, let assume that some classical problems are inherent in such a tool but can be minimized in our context.[1]

Let notice that the post selection process will suppose the data of N by the user, as the number of retained regions. So, the top of the pyramid is computed as the last level where at least N elements appear. In order to implement a fast and without any a priori predicates algorithm, we perform the simple following steps:

1. Assign a different label to each element of the top level. Moreover, if two connected "seeds" are similar enough, the same label is affected. More precisely, each pyramid pixel has a three-dimensional color description in the L*a*b* color space, turning the fact to good account that this color space is approximately perceptually uniform; thus distances in it are more meaningful.
2. Force label choice of each pixel in the level immediately below to its nearest parent. For example, in an overlapping 4 x 4 structure, each "child" choose between four potential fathers. With this top-down process, both spatial and color similarity are used.
3. Repeat the same process until the first level (the original image) of the linked pyramid is reached.

Finally, a coarse segmentation is obtained. Nevertheless, this method does not extract objectively regions or objects like we can expect from a classical segmentation algorithm as the number of regions seems to be too important. Figure 1 illustrates these steps for different sample images. Objectively, our purpose was not to create a new "perfect" unsupervised method but to obtain a coarse result relevant enough and sufficiently enough to be used and adapted for general color image indexation tools. Moreover, it can be simply modified to integrate other classical features as texture or shape criteria.

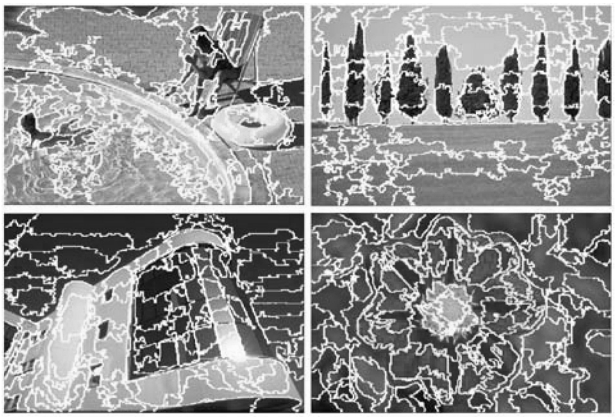


Figure 1. Color Segmentation

Regions Selection

At this point, we obtain a region set, covering the entire image as noted before. This coarse segmentation creates a "large" number of regions and, in order to describe the image, our goal now is to select the most representative regions. We will restrict this problem in extracting exactly *N* regions, *N* set by the user. Setting the number of regions before is in fact the more simple solution to best evaluate the distance between two images. The selected regions have to describe the visual information contained in the image. More precisely, we present in this paper the color similarity selection. These selected regions are called "Regions of interest", in comparison with interest points detector. Each element of the region set is described following by parameters:

- {R,G,B} colorspace values and average on each region.
- {L*,a*,b*} colorspace values and the ΔE dispersion computed on each region.

As we exclude for the moment to use another parameter describing the regions, like spatial information for example, the selection uses only color information. Our first idea was to separate the distribution of region color average (in RGB color space) in *N* classes and to extract *N* regions representative of each class. But RGB space has three dimensions, so computing classes (with statistical methods or genetic algorithms) requires a lot of resources. Then, the problem is more simple if we are able to restrict the problem of classes separation on a 1D space. Two different ways, answering differently to a query, were chosen. First, to select region by using the measure of color dispersion: the ΔE information on each region. The regions are separated from homogeneous to very disparate (in color) one. The second solution is to compute color description in the *HSV* (hue saturation value) color space, and to use hue for selection. Computing *HSV* is a linear calculus (and we compute it only from RGB average values), and, so, requires really few resources. Let us now describe more precisely this second way explored.

This choice is not, of course, without any problem and interrogation. Nevertheless, our goal is not to make the best regions selection but to make a hotchpotch between fast computing and quality of results.

For each region, we compute then the hue *H* from the *RGB* average value. Then, we obtain a one dimension space for describing regions. We have to note that this space is linear but, circular too. Indeed, the construction of the hue from 0 to 360 degree makes a circular vision of the hue : Red hue is as much 1 degree as 359. Then, with a circular distance and a classical genetic algorithm (k-average) we obtain *N* classes with a good distribution on the circular value *H*. Figure 2 illustrates this discrimination.

Figure, we obtain *N* classes where we select two regions in each one, one maximizing - and the second minimizing - the value *V*. Indeed, a class represents a kind of hue (it is not logical to select another time on hue). The value in *HSV* color space model indicates the luminosity of the color and, visually, the focalisation is probably the higher value of *V*. Finally we take, for same reason of visually attract, the lower value of *V*, the darkest.

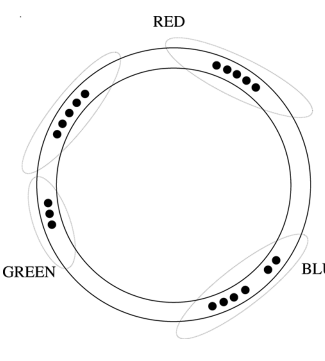


Figure 2. Classes discrimination using H

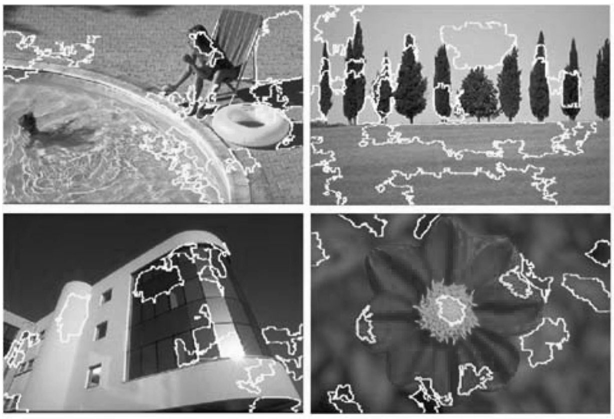


Figure 3. Regions selected by H and V

Information Retrieval Problem

We have now computed a new method for regions selection. This extraction provides regions not too large, homogeneous and, normally, representative enough of the visual information contained in the image. If we show different regions set to a human user, he will be able to answer the question: from which "regions set" this one is the nearest? In fact, we have restricted the problem on *N* regions but the problem of indexation stay: how comparing regions?

Region Descriptors

First of all, it needs to compute descriptors for each region. We retain for one region a vector, which will serve for measuring distance. Each region are, by segmentation and selection method, describe by *RGB* average values, ΔE dispersion and also the *HSV* values. We will retain for the first tests the *RGB* values average.

Of course this choice is really restrictive and we can not expect the most quality for information retrieval tests. Anyway our goal is first to introduce a new method for region selection and, secondly to show how it can be used in a content based retrieval application. Others descriptors may be used, in order to best color discrimination for example the ΔE dispersion computed on each region.

Distances "Blob to Blob"

The notion of distance between 2 images is the key of all information retrieval tools. Of course, we need now to establish a distance between 2 regions set. Regions are described by a vector (*RGB* values for example) and we suppose for each descriptor we have an implicit distance. In the case of *RGB* a classical one is Euclidean distance d_e.

Of course, using a classical distance between the *N* vectors of the 2 images is not logical and do not give interesting results. It needs a distance specific to a blob comparison! We purpose different distances, each one has its proper characteristics and own kind of similarity.

A naive idea is to use mathematics models and to use well-known distances as:

$$\sum_{0 \le i < N; 0 \le j < N} d_e(A_i, B_j)$$

For measuring similarity between 2 sets of regions, our idea is to use a "blob to blob" adaptative distance. We search first the couple which minimize the Euclidean distance d_e. We extract them respectively from the set A and B and we iterate the same method until all couples have been extracted. Finally the distance is the sum of all distance between each couple. In fact it assimilates (the best possible with Euclidean distance) one region with another.

First Results

(a) Image query

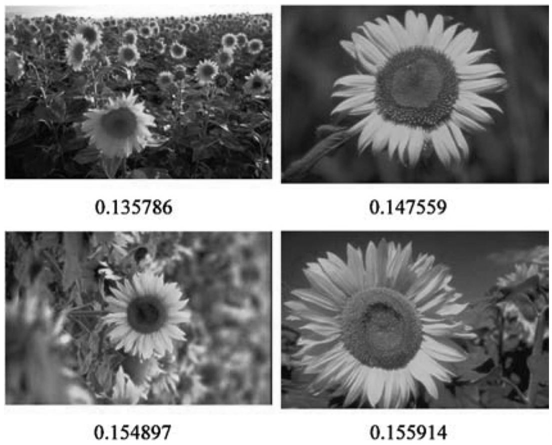
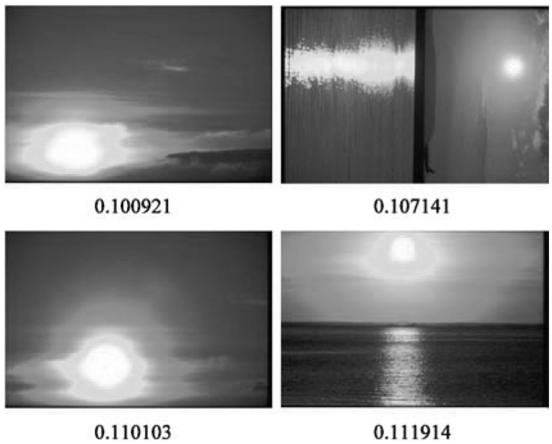


Figure 4. Example 1

(a) Image query



(b) Similar images

Figure 5. Example 2

Figures 4 and 5 present the more similar images to the image query from left to right, top to bottom.

Perspectives

We have presented a new approach for image indexation, using a coarse segmentation and a region selection to measure specific distances between images. Pyramid choice for segmentation is a powerful way to create a fine segmentation for a region selection: fast, insensitive to scale, compression and good stability to light or color enhancement. Color region selection based on hue from *HSV* space permits to extract *N* interesting regions, summarizing the visual information contained in the image. Then, with a "blob to blob" distance we have shown that this method is a previous step to explore for image indexation.

Nevertheless, in this work, we only experimented with the idea to retrieve similar images according to a simple color point of view. So, several limitations are still clearly viewable on some requests and give quite insufficient results. Objectively, we do not incorporate as far as possible the human perception on seeing an image. Actually, we just merge regions during the request step that are color similar without combining other major informations as texture, shape criteria or else spatial relationships. In fact, in the example presented figure 1, we expect to describe the flower as a yellow blob surrounded with a larger red one. Future research work will then be divided into two major steps: first to improve the segmentation method and secondly to incorporate a "semantic" aspect as much in the region selection as in the distance.[12] Such achievement will be certainly reached using neighboring or inclusion graphs.

References

* Work supported by the region Rhône-Alpes grant ACTIV2

** http://www.ligiv.org/icobra/

1. M. Bister, J. Cornelis, and A. Rosenfeld. A critical view of pyramid segmentation algorithms. *Pattern Recognition Letters*, 11:605–617, Sept 1990.
2. Chad Carson, Megan Thomas, Serge Belongie, Joseph M. Hellerstein, and Jitendra Malik. Blobworld: A system for region-based image indexing and retrieval. *Visual information systems*, 1999.
3. Alberto Del Dimbo. *Visual Information Retrieval*. Morgan Kaufmann publishers, inc, 1999.
4. M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Janker. Query by image and video content: the qbic system. *IEEE Computer*, 28(9):310–315, 1995.
5. C. Harris and M. Stephens. A combined corner and edge detector. *4th Alvey Vision Conf. Manchester*, pages 189+, 1988.
6. H. Konik, V. Lozano, and B. Laget. Color pyramids for image processing. *Journal of imaging science and technology*, 40(6):535–542, November 1996.
7. Jérôme Da Rugna and Hubert Konik. Color interest points detector for visual information retrieval. Electronic Imaging, SPIE, 2002.
8. Arnold Smeulders, MarcelWorring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Image databases at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(1), January 2001.
9. S.M. Smith and J.M. Brady. SUSAN - a new approach to low level image processing. *Int. Journal of Computer Vision*, 23(1):45–78, May 1997.
10. M. J. Swain and D. H. Ballard. Indexing via color histograms. In *IEEE Proceedings*, pages 390–393, 1990.
11. Brian A. Wandell. *Fundations of Vision*. Sinauer Associates, Massachusetts, 1995.
12. James Z. Wang, Jia Li, and Gio Wiederhold. Simplicity: Semantics-sensitive integrated matching for picture libraries. *IEEE transactions on pattern analysis and machine intelligence*, 23(9):947–963, 2001.

Biography

Hubert Konik is a Professor Assistant at the Université Jean Monnet de Saint-Etienne, France since 1996. His research interest are multi-resolution and image retrieval by contact applications.