# Spatialized Multi-Visual Features-Based Image Retrieval

Youssef CHAHIR[†] and Liming CHEN[‡]

[†] *Laboratoire HEUDIASYC UMR CNRS 6599*
*Université de Technologie de Compiègne*
*Centre de Recherches de Royallieu*
*BP 20529, 60205 Compiègne Cedex, France*
*E-mail: ychahir@hds.utc.fr*

[‡] *Département Mathématiques et Informatique*
*Ecole Centrale de Lyon*
*36, av. Guy de Collongue*
*BP 163 - 69131 ECULLY Cedex, France*
*E-mail : liming.chen@ec-lyon.fr*

## ABSTRACT

The content-based image indexing is a key technology for a large scale use of digital libraries. Within the French Information Highway program supported project "TransDoc"[†], we have been proposing solutions to this problem. In this paper, we investigate our methods for image retrieval based on spatialized multi-visual features, including color histogram, texture and some simple shape. We represent images according to a spatial hierarchical structure, namely quadtree structure, extracting visual features for each region of an image. Image splitting method and clustering technique are then applied to segment spatialized visual information such as texture, shape, and color. This information is used to retrieve complex queries which can be a combination of spatialized multi-features. We also discuss some experimental results.

## Keywords

color, content-based retrieval, feature-based indexing, quadtree, clustering

## 1. INTRODUCTION

Image and video is a major media for information conserving and dissemination in a information society under way. However, their full use are currently limited by their opaque characters in which yet there is no automatic content based indexing techniques, such as the ones for text based information. On the other side, a manual indexing of image is too costly to be applied to ever growing image documents. Consequently, building content based image retrieval systems is the focus of a large research community[1,2,3,4].

Image database systems go beyond the traditional capabilities of database systems as they contain image data. They require techniques that allow to classify images, filter relevant information, and retrieve images from a large set of stored images, based on their content. If the very first image database systems require to characterize images by a set of semantic attributes manually extracted[5,6,7], extracting automatically the full semantics of images to enable content-based querying is not conceivable. The current trend consists of using visual features and distance of similarities to enable content-based image querying[8]. The visual features usually refer to color histograms, complex scene textures such as sand, lawn, and object shapes. Content-based image retrieval systems find and retrieve a set of images from the database that satisfy the similarity criteria as compared to a given image query. However, most techniques proposed in the literature for the image content based retrieval are based on very simple image features, such as color histograms [9], efficient indexing structures[10], and sometimes make use of pre-filtering techniques[11]. A more sophisticated feature like color coherence vectors was also proposed for the indexing purpose[12]. However, these approaches have neglected two important visual features for similarity comparison : spatial information and spatial relationships.

By representing images symbolically [13,14], locations and spatial relationships of symbols have been used for the spatial queries evaluation. Unfortunately, these techniques cannot easily accommodate measures of similarity of the symbols such as visual features. Recently, there are some efforts trying to take into account the spatial information for the image retrieval purpose. For instance, VisualSEEK[15] project proposes the extraction of localized regions and features, and allows image querying by both global features and spatial information.

In this paper, we investigate our method for image retrieval based on spatialized multi-visual features, including color histogram, texture and some simple shape. We represent images according a spatial hierarchical structure, namely quadtree structure, extracting visual features for each region of a image. Image splitting method and clustering technique are then applied to segment spatialized visual information such as texture, shape, and color. This information is used to retrieve complex queries which can be a combination of spatialized multi-features. We also present discuss some experimental results.

The rest of this paper is organized as follows. Section 2 briefly presents the related work. We introduce in section 3 the architecture of our system. Section 4 detail the visual features that we extract from images and we

[†] TRANSDOC Project web adress: http://transdoc.ibp.fr

use for visual content-based retrieval. Section 5 describes the process of homogeneous regions identification. Section 6 briefly presents our search engine, which is followed by some experimental results. In conclusion, we summarize our work and give some directions of our further investigation.

## 2. RELATED WORK

Stricker and Dimai[16] proposed to divide each image into five fuzzy regions. They used color moment representation and allowed the user to assign weights to the five spatial regions. However, it is not possible to query for images by specifying either arbitrary regions or the spatial relationships among regions. Pass and al.[12] splits a global image histogram into coherent and scattered components to identify the connected colored regions. They get more precise color histogram information for query purpose, but their approach does not consider neither regions nor texture. X. Wan and C.-C. Jay Kuo investigate image retrieval based on JPEG compressed data[17]. They extract two level features in the DCT (Discrete Cosine Transform) domain. Then they use a tree structure to represent the DC coefficients of Y, $C_b$ and $C_r$ block arrays and the energy distribution of AC coefficients. However, their visual content is only limited to the color. J. Zhao and al.[18] present a compression method that allows in a flexible way the decompression quality of regions in an image. They use fuzzy technique to determine important regions in the same manner as human beings do. They describe the important region features. But their work is only limited to the compression and decompression purpose by means of several control parameters such as gray level threshold. F. Golshani and Y. Park present ImageRoadMap[19], a visual content-based image retrieval system. Their method use the DC values obtained from a Discrete Cosine Transform (DCT) to calculate a histogram. They apply split and merge algorithm on blocks with fixed threshold to extract homogeneous blocks. The drawback of their work is that these homogeneous blocks with compared to DCT are not necessary visually homogeneous regions.

The semi-automatic extraction approach is used in many systems, such as QBIC[20], which successfully used fully automatic unsupervised segmentation methods as it is able to to recognize regions on a background model in a specific class of images. However generally the user is assumed to be an expert of the application domain in order to semi-automatically extract the shape of relevant regions. In VisualSeek system[15], the user forms the queries by diagramming spatial arrangements of color regions. It is hybrid system in that it integrates feature-based image indexing with spatial query methods. However, their region segmentation is based on the color back projection, and it is very expensive.

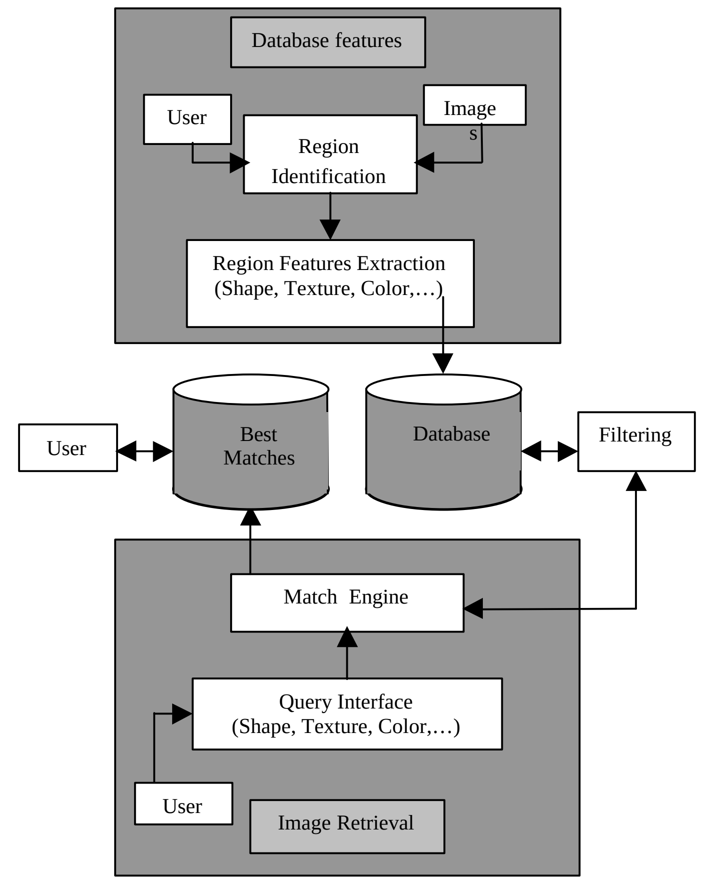



**Figure 1 – system's architecture**

## 3. THE ARCHITECTURE

The prototype we have been implementing has two main components : the component for features extraction (indexing) when image is inserted into the database, and the one for image retrieval (querying). However, a preprocessing with help of users may also be applied to an image during its insertion, including image decompression, image filtering, normalization, segmentation, and possibly object identification. The output of this stage is a set of significant regions and objects.

The component for features extraction, by means of quadtree representation and split and merge techniques, hierarchically decomposes an image into homogenous regions according to several criteria such as color histogram, textures, and basic shapes, thus also spatializes these features. Furthermore, these visual features, inserted into the database, are classified into primitives such as lines, circles, etc.

The querying component provides an interactive graphical browser and enables user querying on the image database. The user can either sketches a image query or chooses an image from the database as image query. For the retrieval purpose, several similarity distances are used, including quadratic distance, normalized histogram intersection distance, etc. When a user provides an image query by means of the browser, the visual features are extracted from this image query and they are compared to the features stored and previously extracted from the images in the database. When results are retrieved, the browser allows users to visually navigate among the images returned as answers. The figure 1 illustrates such an architecture.

Content-based retrieval :

*Preprocessing:* The image is first processed in order to extract the features, which describe its contents. The processing involves decompression, filtering, normalization, segmentation, and object identification. The output of this stage is a set of significant regions and objects.

*Feature extraction:* Features such as texture, color, etc. are used to describe the content of the image. Image features can be classified into primitive. We can extract features at various levels.

Organization : Efficient query processing necessitates the organization of image such that efficient search strategies can be used. Hashing and structures such R-tree (and variants), quadtree are used.

## 4. VISUAL FEATURES OF IMAGES DATABASE

For the database population purpose, image sources may be in different formats, such as GIF, JPEG, etc.. Before their insertion into the image collection, global visual features are first automatically extracted, and during this step users are also invited to annotate images. We then proceed to the extraction of further features such as dominant texture, the dominant color of the image, and the shape of relevant regions, in which user may identify objects and textually annotate them for the future high semantic level retrieval.

In the following, we introduce the three basic visual features that are extracted in our system : color, texture and shape.

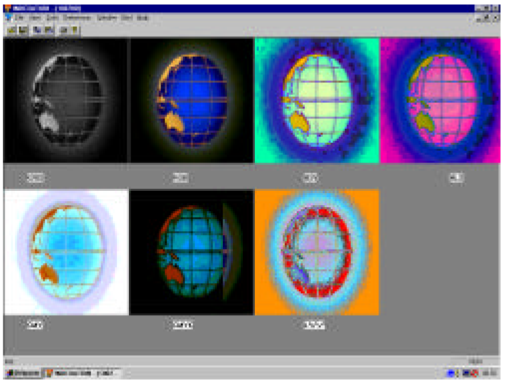


**Figure 2 - A same image in different color space**

### 4.1. Color

The color is the first basic visual feature. Thus, it constitutes a very important attribute for image retrieval. However there exists a number of color space representations. The RGB space has been widely used due to the availability of digitized images in the RGB format. Based on such a representation, the color may be transformed and represented in other models such as L*u*v*, HSV, HLS, YUV, YIQ, CMY etc . In figure 2, we represent an image in different color spaces. Left to right: Gray, RGB, HSV, HLS, CMY, CMYK and L*u*v*.

The color-based visual feature is captured by the image histogram which is the color distribution in an image (see figure 3). In a more precise way, a histogram is an N-dimensional vector where each bin i contains the number of pixels having color i. This feature is a global one, but it is invariant to image rotation and translation.

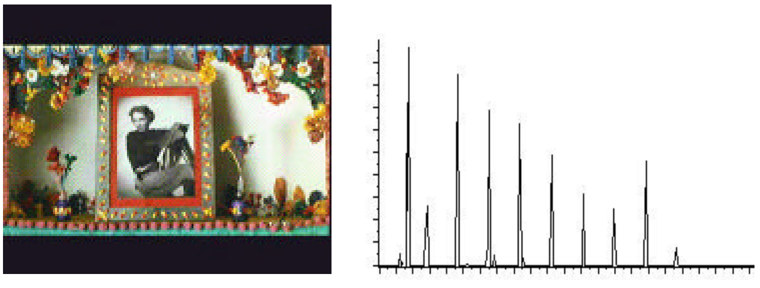


**Figure 3 – an image and its histogram**

The color histogram of an image can be represented either by a single 3-D histogram or three separate 1-D histograms. For example, a RGB color histogram which has been quantified into k bins for R, l bins for G and m bins for B can be represented as a single 3-D $H_{klm.}$ We can also store one histogram by image where each bin corresponds to each color, which is a combination of (R, G, B). Both representations are used in our experiments with respect to several different similarity measures. A normalized histogram is often used to compare images. It is defined by the following formula :

$$H^N(im_n, i) = \frac{H(im_n, i)}{\sum_i H(im_n, i)}$$

Where $H(im_n, i)$ is the histogram of an image $im_n$, where the index i represents a histogram bin.

Different similarity metrics were proposed to compare two histograms. They may be histogram difference, intersection, euclidean, normalized histogram intersection or quadratic. Below we briefly recall these distances.

Histogram difference:

$$d_1(H(im_m), H(im_n)) = \sum_{i=1}^{N} \left| H(im_m, i) - H(im_n, i) \right|$$

Euclidean distance:

$$d_2(H(im_m), H(im_n)) = \sqrt{\sum_{i=1}^{n} (H(im_m, i) - H(im_n, i))^2}$$

Histogram intersection distance:

$$d_3(H(im_m), H(im_n)) = \sum_{i=1}^{n} \min(H(im_m, i) - H(im_n, i))$$

Jain and A. Vailaya21 used three 1-D histograms in their experiments. The similarities used between the query image and a stored image in the database are given by the following equations:

Histogram intersection:

$$d_7(im_m, im_n) = \frac{\sum_r \min(H_R^N(im_m, r), H_R^N(im_m, r)) + \sum_g \min(H_G^N(im_m, g), H_G^N(im_m, g)) + \sum_b \min(H_B^N(im_m, b), H_B^N(im_m, b))}{3 * \min(\left|H(im_m)\right|, \left|H(im_n)\right|)}$$

Euclidean distance:

$$d_8(im_m, im_n) = 1 - \sqrt{\frac{\sum_r (H_R^N(im_m, r) - H_R^N(im_m, r))^2 + \sum_g (H_G^N(im_m, g) - H_G^N(im_m, g))^2 + \sum_b (H_B^N(im_m, b) - H_B^N(im_m, b))^2}{2*3}}$$

Note that the values of these distances lie in the interval [0,1]

Normalized histogram intersection distance:

In this metric, the intersection measure is incremented by the number of pixels, which are common between the

$$d_4(H(im_m), H(im_n)) = \frac{\sum_{i=1}^{n} \min(H(im_m, i) - H(im_n, i))}{\sum^{n} H(im_n, i)}$$

target image and the query image. The measure is finally divided by the total number of pixels in the query image as a normalization factor. This metric is insensitive to changes in image resolution, histogram size, occlusion, and depth. However, this metrics does not consider the perceptual similarity between the different bins. A metric which takes into account the similarity between the bins is the one based on a quadratic distance.

Quadratic distance:

$$d_5(H(im_m), H(im_n)) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} (H(im_m, i) - H(im_n, i))(H(im_m, j) - H(im_n, j))$$

Where the weight $a_{ij}$ represents the similarity coefficient between colors corresponding to bins i and j. This metric has a higher computational complexity than the histogram intersection. However it is closer to human judgment of color similarity.

To reduce the computational complexity in color indexing, we propose in our approach the use of the dominant features of a histogram, that is the dominant colors. We then use the first three moments of each color component to define a similarity distance :

Distance between moments:

$$d_6(H(im_m), H(im_n)) = \sum^{n} w_{i1} \left| \overline{H(im_m)} - \overline{H(im_n)} \right| + w_{i2} \left| \mathrm{var}(H(im_m)) - \mathrm{var}(H(im_n)) \right| + w_{i3} \left| s_m - s_n \right|$$

Where $\overline{H(im_m)}$ s the mean, Var(H($im_m$)) the variation and $\sigma_m$ the moment of degree 3 of H($im_m$)

This distance outperforms the previous distances as it reduces the number of bins in terms of storage requirements, retrieval speed and robustness.

Note that the previous color histogram based distances allow to compare images based on a global feature, without taking into account the location of the colors. We eliminate this problem by incorporating spatial information in the content representation. In our method, we can also use local histograms. Indeed, during the process of the quadtree decomposition of an image, histograms of the subimages are also calculated. Thereby we capture the locality of colors in an image.

## 4.2. Texture

The texture is another basic aspect of human visual perception, and it is the second visual feature that we propose to automatically extract from image regions. In our approach, the texture analysis methods are statistical or structural one. An example of textured image is illustrated in figure 4.

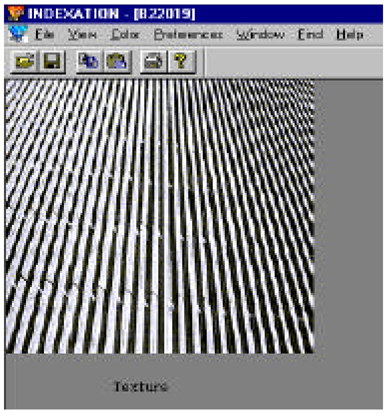



**Figure 4 : Textured image**

Concerning the statistical texture analysis, many methods are proposed in the literature, such as those based on the gray level (gray level difference, gray level run lengths end gray level profiles)[22], filter mask texture measures[23], co-occurrence features[24], auto-correlation features, relative extrema measures[25]. Structural methods proposes to describe texture by some primitives and placement rules[26]. For instance, on can measure several primitives features such as area, perimeter, compactness, eccentricity, and direction. The drawback of this approach is that it requires some preprocessing with respect to the chosen primitive feature and attention has to be paid to the influence of rotation and scale on the texture analysis.

On the statistical level we consider a texture as defined by a set of statistics extracted from a large set of local picture properties. Even simple statistics such as the gray level first-order statistics can be used to classify a limited set of textures. However, it is well known, as pointed out by Julesz [27] that Human beings are more sensitive to second-order statistics, that is why the second order moments are also considered in our system.

To summarize, the following simple texture measures are computed : the mean, the coarseness, the variance, the contrast, the entropy and the angular second moment which captures the directionality. The variance measures the dispersion of the difference of gray-level with a certain distance. The contrast measures the vividness of the texture and is a function of gray-level difference histogram. The directionality measures the "peakedness" of the distribution of gradient directions in the image. The Mean and the Variance provide an indication of how uniform or regular a region is. We also represent the texture of a region by statistic four first moments formulas.

Other possible texture measures on the histogram may be the minimum, the maximum, the skewness, the kurtosis. While the Skewness is an indication of symmetry, the Kurtosis measures the degree of peakedness.

The major advantage of using the texture attributes is their simplicity. Combined with other visual features, the texture measures are a powerful tool for retrieval process. The following formulas give the different statistic measures of texture.

C is the coarseness , it is defined as:

$$C = 1 - \frac{1}{1 + S_D}$$

$$S_D = \sum_{i=0}^{N-1} (i - S_M)^2 \; h[i]$$

$S_D$ is the dispersion of the image

$S_M$ is the mean of the histogram h of the image

Let I(i,j) be the value of the pixel (gray level) at i^th^ line and j^th^ column. The two-dimensional (p+q)^th^ order moments are defined as:

$$m_{pq} = \sum_i \sum_j i^p j^q I(i, j) \qquad p, q = 0,1,2,...$$

$(\bar{x}, \bar{y})$ is the gravity center, or the mean of the pattern.

$$m_{pq} = \sum_i \sum_j (i - \bar{x})^p (j - \bar{y})^q I(i, j) = \sum_{r=0}^{p} \sum_{s=0}^{q} C_r^p C_s^q (\overline{-x})(\overline{-y}) m_{p-r,q-s}$$

Let $m_{00}$= A be the area of a shape. The central moments $\mu_{pq}$ are invariant under translation and are defined as:

where $$C_r^p = \frac{p!}{r!\,(p-r)!}$$

Hu[28] proposed seven low-order invariant moments $M_1$, …$M_7$, which are functions of the second-and third-order central moments. More higher is the order of the moment, more result is independent of size, and orientation as well as position.

$$Variance = s^2 = m_2 = \frac{1}{N} \sum_i n_i (x_i - \bar{x})^2$$

$$Entropy = -\sum_j p_i \log(p_i)$$

With $p_i$ = occurrence probability of pixel i

The fundamental problem with the computation of textures is the choice of an appropriate window size[29]. When textures features are measured on small subimages, they are unreliable ; but if we use large subimages, it is rather hard to find out regions that are uniformly textured. To overcome these difficulties, we use a pyramidal approach. The "pyramidal" approach consists of successively reduced-resolution versions of the given image. Fathers remain intact if their son nodes are sufficiently similar, otherwise the block is split and the process is repeated for the son quadrants. When this process is complete, each block that remains unsplit is now contained in a homogeneously textured region. This process is accomplished by the quadtree decomposition[30]. We calculate many attributes on squared local areas. This decomposition process is then followed by a grouping algorithm in order to cluster squared areas into homogenous regions, and these homogenous regions are then in their turn clustered into homogenous super-regions, using a feature similarity threshold. More precisely, a depth-first search in the region adjacency graph is performed, followed by a small region elimination step in which regions are grouped according to their mean merge criterion (gray level, central moment…etc). Thus we build a multi-resolution structure which is very suitable to color and textural based and image retrieval.

### 4.3. Shape

Shapes constitute the third important visual feature that can be extracted from an image or a region. They represent the borders of relevant regions. In generally, relevant regions are significant and well defined in the application domain.

In our system, shapes are represented by a matrix of pixels, which represent the borders in the region. The matrix of pixels consists of elements equal to 1 if the pixel is a part of the region, and 0 if not. While this technique facilitates the display of a region border, but it requires more memory places. Thus we also consider another method in our system because of its compactness, that is the Freeman Codes used to represent the border of the region. In such a representation, a region shape is composed of a starting point (x, y) and a list of element $a_i$, where $a_i$ is an element of {1,2,3,4,5,6,7,8} indicating one of eight directions as illustrated by the next figure 5**.** The next table gives the Freeman code list of the region U.

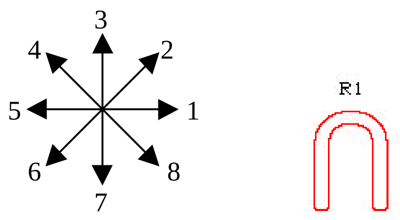



**Figure 5 - Freeman code and a region U**

Freeman code of region R1: 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 5 3 3 3 3
3 5 3 3 5 3 3 5 3 5 3 5 3 5 3 5 3 5 3 5 5 3 5 5 3 5 5 5 5 3 5 5
5 5 5 5 5 5 5 5 5 7 5 5 5 5 7 5 5 7 5 5 7 5 7 5 7 5 7 5 7 5 7 5
7 7 5 7 7 5 7 7 7 7 7 5 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 1 7 1 1 1 1 1 1 1
3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 1 3 3 1 3 1 3 1 1 3 1 1 1 3
1 1 1 1 1 1 1 1 1 1 7 1 1 7 1 1 7 1 7 1 7 7 1 7 7 7 1 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 1 7 1 1 1 1 1 1 1 3

The Freeman code representation also facilitates the deduction of Fourier descriptors, which give a more compact approximation of the shape of a region as compared to the Freeman code representation. The descriptors of Fourier are defined as a set of attributes ($a_n$,$b_n$) where :

$$a_n = \frac{T}{2n^2p^2} \sum \frac{\Delta x_p}{\Delta t_p} [\cos(\frac{2pnt_p}{T}) - \cos(\frac{2pnt_{p-1}}{T})] \qquad b_n = \frac{T}{2n^2p^2} \sum \frac{\Delta x_p}{\Delta t_p} [\sin(\frac{2pnt_p}{T}) - \sin(\frac{2pnt_{p-1}}{T})]$$

Each index "n" of the attributes $a_n$ and $b_n$ represents a step of the contour approximation. For n=0, the descriptors of Fourier represent the barycenter of the region. For n=1, the descriptors of Fourier represent an ellipse. While n increases, the shape obtained approaches the original shape.

The first advantage of the contour representation by Fourier descriptors is the compactness as they generally reduce the number of values used to describe the region contour. They also allow to find out if necessary particular points in the contour such as the contour extreme points. Besides, they mathematically represent the shape independently of the region orientation, and thus enable a certain tolerance of elastic deformations between structures.

In our system, as each shape representation has advantages and drawbacks, we have implemented three shape description models : binary matrix, Freeman code compressed with Run Length Method, and Fourier descriptors

## 5. Characterizing Regions

A region in a image is a homogeneous area according to either color or texture. Their characterization is one of the most important tasks for content-based image retrieval, as they are often used to define semantic objects within an image. In our system, after an image quadtree-based decomposition, we first proceed to reconstitute relevant regions according to close colors, then we also cluster homogeneous neighbor regions in function of their distance. In the following, we first describe homogeneous regions constitution process, followed by an overview of different characteristics extracted from each region, then we also describe spatial-relationship which defines topological relationships between regions.

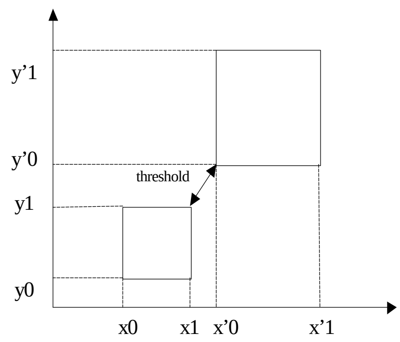


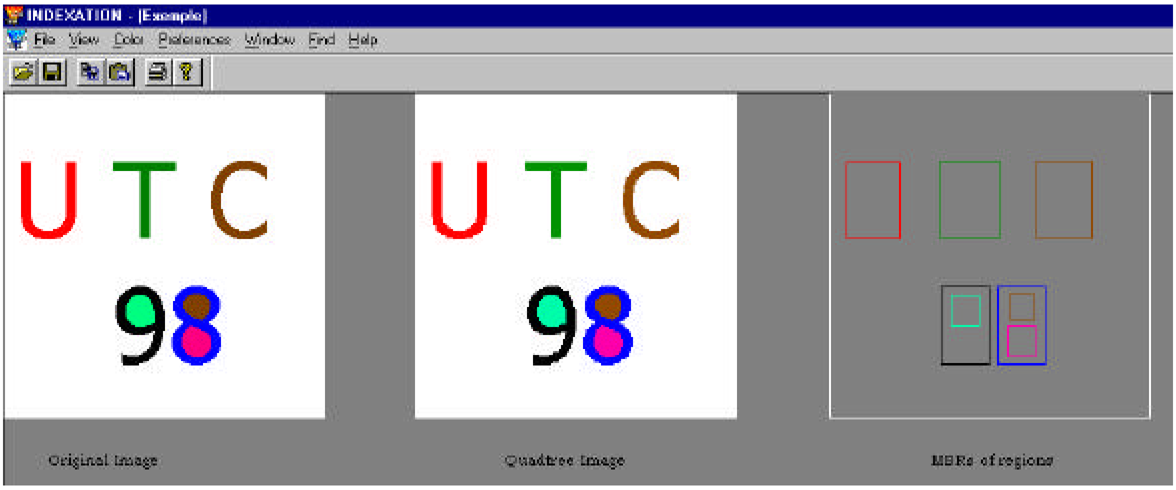

**Figure 6 - merging two close homogeneous regions**

## 5.1. Region Constitution:

The advantage of quadtree[31] based approach lies in the possibility to approximate an image by its quadtree representation with a level largely inferior than the final decomposition. The quadtree representation of an image allows the use of split and merge techniques for the connected component identification. As illustrated by figure 6, two regions are merged into one unique region if their colors are close, and their distance is under a fixed threshold (in pixels).

Thus at the end of such a merging process, each region represents all neighboring quadrants with a close color characteristics. These regions are then framed by a minimum boundary rectangle (MBR) localized by a Peano key[11]. This Peano key gives the spatial position of the center of gravity associated with the MBR. The MBRs of the regions are indexed using an R-tree[32], which provides a dynamic structure for rectangles indexing. Generally, the MBRs overlap.

In figure 7, an image, its quadtree image is shown and the image regions are extracted and then enclosed by minimum bounding rectangles.



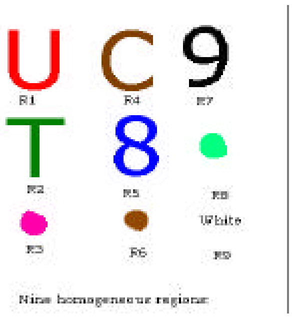




**Figure 7: Homogeneous regions extracted**

5.2. Region Description:

For the further content-based retrieval purpose, each region is also characterized by a set of significant attributes that we present below :

- Region Color: it gives the mean value of gray levels or dominant color components of a region
- Area Ratio: it defines the percentage that a region takes over the whole image area. Thus it gives the relative size of a region. It is defined by the following formula :

$$Arearatio_m = \frac{\sum pixel_m}{Width \times Height}$$

Where $\sum pixel_m$ is the total of pixels belonging to region m.

- Position: it describes how far the center of gravity of the region m is from the center of the image. It is defined by

$$Position_m = \sqrt{(m_x - C_x)^2 + (m_y - C_y)^2}$$

Where $(C_x, C_y)$ is the coordinate of the image center, and $(m_x, m_y)$ are the coordinates of the center of gravity of the region m.

- Compactness: it shows how compact the region $_m$ is, and is defined by

$$Compact_m = \frac{4p \times (area_m)}{(perimeter_m)^2}$$

Where $area_m$ and $perimeter_m$ respectively are the area and perimeter of region m.

- Border Connection : it identifies whether a region is part of the background in an image. It is defined by

$$Border_m = \frac{\sum connect_m}{2 \times (Width + Height)}$$

Where $\sum connect_m$ is the number of pixels which are part of the boundary of region m and connected with the background of the image.

Other measures that we use in our system enable us to compare regions with respect to their geometrical properties. They include for instance :

- Shape of the region
- Area
- Perimeter
- MBR coordinates = Peano key

We also use the following concepts to synthesize some particular regions, such as horizontal and vertical lines, simple shapes for instance circle, square, isolated point denoting a region formed by an isolated point, arcs when the region is formed by more than one point and its area covered by the region is 0. The figure 8 gives the data structure giving the different characteristics attributes which we extract from a region.

| Region ID | Image ID | Dominant color | Area | Perimeter | Center of Gravity | ……. | Area Ratio |
|---|---|---|---|---|---|---|---|

**Figure 8 : The attribute table for region**

### 5.3. Region spatial-relationship identification

Region spatial-relationship identification aims at identifying spatial arrangements relationships among regions. As example of such spatial relationships, we say for instance that a region is *below, in the right, or left* of another region. Including such spatial information in the image representation will not only improve the retrieval quality but also enables users to query an image database based on spatial arrangement of sub-images.

| | | | |
|---|---|---|---|
| | | | d |
| | a | | |
| b | | c | |
| | f | | e |

**Figure 9: 2-D string**

There exist two major approaches for describing spatial relationships between objects. The first approach proposes to use a single content descriptor such as 2D-string and its variants[33]. 2D-string consists of two components : the first one is obtained by projecting the domain objects on the x-axis and the other one by projecting the same objects on the y-axis. 2-D strings consider objects as point objects situated at their gravity center, and they were introduced to characterize spatial relationships among objects[34]. The major drawback with 2D-string-based representations is the difficulty to recognize rotational variants of an image as similar images.

Let (u,v) be the 2-D string over V ={a,b,c,d,e,f} corresponding to the picture in figure 9 is:

$$(u,v) = ( b < f\,a < c < e\,d\ ,\ f\,e < bc < a < d )$$

The second approach typically builds a complete weighted graph to identify spatial orientation between any couple of objects[35]. Thus this approach makes use of several content descriptors, that is pair-wise spatial relationships[13]. However, this approach does not give a unique representation of spatial relationship among a couple of objects. Besides, it leads to a lot of redundancy in the representation. In our system, we have preferred 2-D string representation for its simplicity.

## 6. Visual Content-based querying

Users are allowed to browse and search into a collection of images, say B, using both visual content-based and text-based methods. In content-based searching , the output of a query is a new ordered list of images, say C, with $C \subseteq B$ such that C is ordered by similarity distance to the image query provided by the user. A search operation may be driven over a particular collection of images or the entire database.

Currently, our system provides three methods for visual content-based searching : color histograms based, object spatial locations based or color regions' arrangements based. However, the visual content-based methods that we have developed so far for indexing, searching, and navigation can also be applied to other types of features, such as texture and shape.

During the database populating process, images are clustered and indexed on the basis of feature similarities for the retrieval efficiency consideration. Then they are already ranked in the database by their degree of similarities. Retrieving image data is accomplished by comparing the spatial layout, relationships among objects in the image query, and its visual features such as color and texture, with those of images in the database.

In our prototype, image data and its visual features are stored and represented by several object classes. Each object class schema consists of a header and a data section. In the header, the feature names and some metadata on the file are defined. The data section contains a list of vectors that store values or distributions of visual features. Care must be taken so that the visual features of images are represented at the same scale. Indeed, as image sources may be in very different formats, we are led to normalize these visual features for the similarity-based comparison purpose.

## 7. Experimental results

Since there does not exist a no standard image corpus for the purpose of evaluating image retrieval methods, a test set of 1,500 images was generated for our experiences. The figure 10 illustrates some pictures of our image database.

In our query examples, the query image is always the first image at the top on the left. The retrieved images are ranked by their degree of similarity with each specify query image.

The quality of our query engine is assessed according to two well known measures in information retrieval field : recall and efficiency. For a given query, the images to be retrieved can be categorized into two groups, that is relevant or no-relevant. For a given query response, let A represent the number of images which are correctly retrieved and let C be the number of relevant images which are not returned by the querying process. Note that A along with C gives the total number of relevant images. Let B be the number of images which are falsely retrieved. Then, A along with B gives all the images which are retrieved as the answer of the query. The recall and the efficiency measures can be defined as follows:

**Figure 10 - some pictures of our image testbed**

*Recall* = A/(A+C), and *efficiency* = A/(A+B).

The first experience that we have conducted[36] shows that the use of a color space also has impact on the precision of query results. Generally, the precision of results decrease in the following order: HSV,HLS, L*u*v*,CMY... and at the end RGB. Thus, we have chosen the HSV space for the assessement of the following requests.

In the following, we give the results of queries belonging to three different categories performed by the prototype over a database composed of 1500 images. All the three queries make use of the color as a basic search criterion, but also combined with other visual features.

### 7.1. Querying on color and macro-texture

The first query Q1 is to retrieve textured images of nature based on color and macro-texture. The figure 11 illustrates some results of retrieved images. The image query is the one on the top and left corner. The retrieved images are sorted by the column. 40 images were designated to be relevant. The next table gives a comparison of results with respect to recall and efficiency when different color metrics are used. We can see that the performance of metrics D1,D2, and D5 is very close in this case.

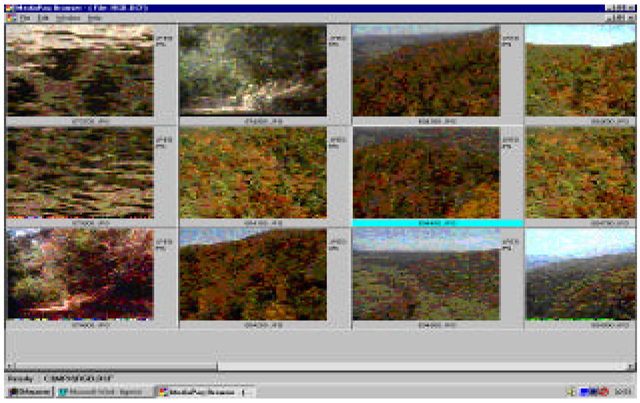


**Figure 11 – color and macro-texture based query results sorted by the similarity**

| 40 nature images | D1 | D2 | D4 | D5 |
|---|---|---|---|---|
| | | | | |
| **Recall** | **0,827** | **0,825** | **0,53** | **0,83** |
| | | | | |
| **Efficiency** | **0,555** | **0,441** | **0,7** | **0,409** |

**Table 1 - Query 1: Nature. Color and Texture**

### 7.2. Querying on color and shape

The second query retrieves images of globes based on color and shape. The figure 12 illustrates some of the query result. 20 images are considered to be relevant. Each of these images is used in turn to query the collection, and answers are evaluated according to each of the following four distance metrics. The result is reported by the next table.

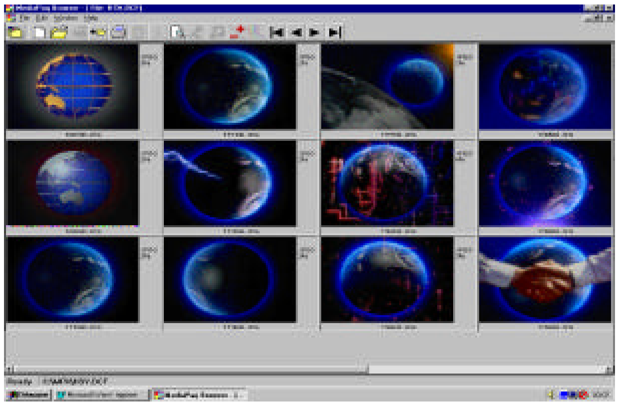


**Figure 12 – Query 2 : Globes. Color and Shape**

| 20 globes images | D1 | D2 | D4 | D5 |
|---|---|---|---|---|
| | | | | |
| **Recall** | **0,871** | **0,770** | **0,769** | **0,857** |
| | | | | |
| **Efficiency** | **0,423** | **0,495** | **0,51** | **0,482** |

**Table 2 - Query 2 : Globes. Color and Shape**

We can see that D5, quadratic metric, is better than D4 with respect to recall and efficiency. The quadratic color histogram improves performance than the Euclidean distance. However, the retrieval effectiveness when D1is used is slightly better than all the other distances in this case.

### 7.3. Querying on Colors and their Spatial relationship (Spatial arrangement)

The third query is to retrieve images of bars according to their colors in the HSV space and their spatial arrangement. The figure 13 shows some results returned by our research engine.

| 20 bars images | D1 | D2 | D4 | D5 |
|---|---|---|---|---|
| | | | | |
| Recall | **0,755** | **0,756** | **0,754** | **0,761** |
| | | | | |
| Efficiency | **0,629** | **0,532** | **0,562** | **0,4** |

**Table 1: Query 3 Bars . Color & Spatial arrangement**

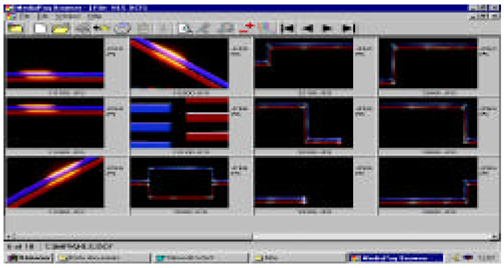


**Figure 13 - Query 3 : bars. Colors and their spatial arrangement**

The goal of query 3 is to retrieve images of red and blue bars where red is bellow blue. The performance of metrics D1, D2, D4 is similar. They provide good image retrieval results, returning, on average, 7 good images among the first ten retrieved.

### 7.4. Lessons from the experiences

In summary, we can say that search methods based on color histogram provide consistently good performance, when they are combined with other visual features. Note also that the use of quadratic metric provide a better result as compared to other metrics, at the cost to have a higher computational complexity.

So far as recall and efficiency rates are concerned, we can see that while recall increases, efficiency decreases. Note that images in a collection are never homogeneous in their nature, this implies that the responses to a query also have a certain degree of inconsistency as compared to the image query.

## 8. CONCLUSIONS

In this paper, we have presented our methods for image retrieval based on spatialized multi-visual features, including color histogram, texture and some simple shape. We represent images according to a spatial hierarchical structure, namely quadtree structure, extracting visual features for each region of a image. Image splitting method and clustering technique are then applied to segment spatialized visual information such as texture, shape, and color. This information is used to retrieve complex queries which can be a combination of spatialized multi-features. We also discuss some experimental results.

We plan to in our future work to tackle the problem of high computational complexity of current color histogram similarity-based methods. We would like to rely on a hierarchical indexing scheme which involves features of lower computational complexity to subset of an image, and to take advantage of wavelet based decomposition of histograms which enables matching based on the indices and magnitude signs of the top few wavelet coefficients for each image.

## 9. REFERENCES


[1] : V.N. Gudivada, V.V. Raghavan, Guest editors, Special issue on Content-Based Image Retrieval Systems, Computer, Vol. 28(9), Sept. 1995

[2] : T. Chiuch, "Content-Based Image Indexing", Proc. Of the 20$^{th}$ VLDB Conf., Santiago, Chile, 1994, pp.582-593

[3] : A.D. Narasimhalu, Guest Editor, Special Issue on Content-Based Retrieval , ACM Multimedia Systems, Vol.3(1), Feb.1995

[4]: A. Ralescu, R. Jain, Guest Editors, Special Issue on Advances in Visual Information Management Systems, J. Intelligent Information Systems, Vol .3(3), July 1994

[5] : N. Roussoupoulos, C. Faloutsos and T. Sellis "An efficient pictorial database system for PSQL" IEEE Transaction on software enginnering, vol. 14, no 5, May 1988

[6] : T. Joseph and A.F. Cardenas "PICQUERY: A high level query language for pictorial database management" IEEE transaction on software engineering, vol. 14, no 5, May 1988

[7] : M. Choc, A. F. Cardenas and A. Klinger "Manipulating data structures in pictorial information systems" Computer, vol. 25, no 11, pp. 43-50, Nov 1981

[8] : P. Aigrain, H.J. Zhang, D. Petkovic, "Content-based Representation and Retrieval of Visual Media: A State of the art Review", Multimedia Tools and Applications special issue on Representation and Retrieval of Visual Media

[9] : M. J. Swain and D.H. Ballard .Color indexing. International Journal of Computer Vision, 1991

[10] : E.G.M. Petrakis and C. Faloutsos. Similarity searching in large image databases. Technical report 3388, Department of Computer Science, University of Maryland, 1995

[11] : M. Flickner and al. Efficient color histogram indexing for quadratic form distance functions. IEEE Trans. Pattern Anal. Machine Intell., July 1995

[12] : G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In Proc. ACM Multimedia 1996

[13] : Gennaro Costaglio and al., Representing and Retrieving Symbolic Pictures by Spatial Relations; Visual Database Systems, II 1992 IFIP

[14] : A. Soffer and H. Samet. Retrieval by content in symbolic-image databases. In Symposium on Electronic Imaging: Science and Technology – Storage and Retrieval for Image and Video Databases IV, IS&T/SPIE 1996

[15] : John R.Smith and Shih-Fu Chang, VisualSEEK: a fully automated content-based image query system, ACM Multimedia '96 Procedings 1996

[16] : M. Stricker and A. Dimai. Color indexing with weak spatial constraints . In Symposium on Electronic Imaging: Science and Technology - Storage & Retrieval for Image and Video Databases IV, pages 29-41. IS&T/SPIE, 1996

[17] : Xia Wan and C. Jay Kuo, IMAGE Retrieval Based on JPEG Compressed Data , Proc. SPIE Multimedia Storage and Archiving Systems , p104-115; Vol 2946, 18-19 Nov. 96 Boston

[18] J. Zhao et Al , "A JPEG Codec Adaptive to Region Importance", The Fourth ACM International Multimedia Conference , 96,

[19] : Content-based image indexing and retrieval in ImageRoadMap, Multimedia Storage and Archiving Systems II, Volume 3229 p: 194-205, 1997, 3-4 Dallas

[20] : Myron Flicker and al., Query By Image and Video Content: the QBIC system, Computer vol 28, Number 9, Sep 1995.

[21] : Anil K. Jain et A. Vailaya, "Image Retrieval using Color and Shape", May 15, 1995

[22] : Martin D. Levine Vision in man and machine, McGraw-Hill Book Company

[23] : E.J. Carton, J. Weszka, and A. Rosenfeld, Some Basic Texture Analysis Techniques, TR-288 Computer Vision Laboratory, Computer Science Center, University of Maryland

[24] : P.C. Chen and T. Pavlidis, Segmentation by texture using a co-occurrence matrix, Computer. Graphics Image Process. 10, 1979, 172-182

[25] : L. Van Gool and al. Texture Analysis Anoo 1983, Computer vision, graphics, and image processing 29, 336-357 (1985)

[26] : T. Matsuyama, and al. A structural description of regularity arranged textures, in Proc. 5th Int. Conf. on Pattern Recognition, Miami, Florida , pp. 1115-1118

[27] : B. Julesz, Experiments in the visual perception of texture, Sci. Amer. 232, No.4. 2-11

[28] : Hu, M.-K ,"Visual Pattern Recognition by moments invariant ", IRE Transactions on Information Theory, vol. IT-8 no.2, pp. 179-187

[29] : D. Barba and J. Ronsin, New method in texture analysis in the context of image segmentation, Proc. 2nd European Signal Processing Conference, Erlangen, W. Germany, Sept. 12-16, 1983, pp 283-286

[30]: Y. Chahir et L. Chen, "Peano key rediscovery for content based retrieval of images", SPIE , Multimedia Storage and Archiving Systems II, Volume 3229 p: 172-181, 1997, 3-4 Dallas

[31] : Hanan Samet. Applications of Spatial Data Structures Computer Graphics Image Processing, and GIS, Chap. 3 and chap. 4

[32] : A. Guttman. R-trees: A dynamic index structure for spatial searching. In ACM Proc. Int. Conf. Manag. Data (SIGMOD), pages 47-57, June 1984

[33] : S.K. Chang and al.. Iconic Indexing by 2D Strings. IEEE Transaction on PatternAnalysis and Machine Intelligence, p(3):413-428, 1987

[34] : S-Y Lee and F.–J.Hsu . Spatial reasoning and similarity retrieval of images using 2D C String knowledge representation. Pattern Recognition, 25(3):305-318,1992

[35] : V. Gudivada and V. Raghavan. Design and evaluation of algorithms for image retrieval by spatial similarity . ACM Transactions on Information Systems, 13(1): 115-144, April 1995

[36] : Y.Chahir, L.Chen, “Influence de l’espace couleur sur la précision de recherche par le contenu basée sur la couleur. Rapport interne de recherche au laboratoire HEUDIASYC: Mars 1998