

# Color-based image retrieval using spatial-chromatic histograms

L. Cinque<sup>a,\*</sup>, G. Ciocca<sup>b</sup>, S. Levialdi<sup>a</sup>, A. Pellicanò<sup>a</sup>, R. Schettini<sup>b</sup>

<sup>a</sup>Dipartimento di Scienze dell'Informazione, Università La Sapienza di Roma Via Salaria 113, 00198 Roma, Italy

<sup>b</sup>Istituto Tecnologie Informatiche Multimediali, CNR Via Ampere 56, 20131 Milano, Italy

Received 3 August 2000; received in revised form 30 April 2001; accepted 26 May 2001

## Abstract

The paper describes a new indexing methodology for image databases integrating color and spatial information for content-based image retrieval. This methodology, called Spatial-Chromatic Histogram (SCH), synthesizing in few values information about the location of pixels having the same color and their arrangement within the image, can be more satisfactory than standard techniques when the user would like to retrieve from the database the images that actually *resemble* the query image selected in their color distribution characteristics. Experimental trials on a database of about 3000 images are reported and compared with more standard techniques, like Color Coherence Vectors, on the basis of human perceptual judgments. © 2001 Elsevier Science B.V. All rights reserved.

**Keywords:** Image retrieval; Image indexing; Color quantization; Query processing; Spatial-Chromatic histogram

## 1. Introduction

The dramatic improvements in hardware technology have made it possible in the last few years to process, store and retrieve huge amounts of data in multimedia format. First attempts to manage pictorial databases relied on textual description provided by a human operator. This time-consuming approach rarely captures the richness of visual content of the images cannot be easily expressed (either interactively or automatically) through words. For this reason research has been focused on the automatic extraction of the visual content of image to enable indexing and retrieval. A broad range of techniques [1,2] are now available. In particular, techniques exploiting low-level visual features have become a promising research issue [3,4,6,11,12,13,14]. General-purpose systems of this type are also now available e.g. [15,17].

These systems usually make it possible to extract image representations in terms of color, texture, shape and layout features from the images and define the relative search/matching functions that can be used to retrieve those of interest. However, notwithstanding the substantial progress made, the integrated management of the different features remains complex and application dependent [16,19].

We limit ourselves here to work with color and introduce

an efficient and effective method to characterize the amount and arrangement of the color pixels in an image. Such a feature may coexist with other features (shape, texture, size, distance, relative position, etc.) expressing different aspects of image content in the framework of a Visual Information Retrieval System, such as [18].

The problem here addressed has been defined by Mehtre et al. [16] as: “Assume that there are a large number of color images in the database. Given a query image, we would like to obtain a list of images from the database which are “most” similar in color to the query image”. More formally, given a query image  $Q$  and a database of images, find the images having chromatic content similar to  $Q$ . The chromatic similarity is to be computed using a function which evaluates the closeness of features extracted from images.

The image retrieval process can be divided in two steps: (1) indexing, for each image in a database a set or a vector of features summarizing its content properties is computed and stored; (2) retrieval, given a query image its features are extracted and compared to the others in the database. Database images are then ordered following a similarity criterion, as stated above.

In order to allow a software system to automatically perform image indexing, based on image content, we developed a simple technique to compute both color and spatial features. We believe that simple color information, such as those provided by a histogram, are not sufficient in many situations.

In this paper, we describe a novel, effective technique,

\* Corresponding author. Tel.: +39-06-4991-8431; fax: +39-06-884-1964.

E-mail address: cinque@dsi.uniroma1.it (L. Cinque).

called Spatial-Chromatic Histogram (SCH), to code spatial-chromatic features of the indexed image. SCH has been defined to answer the following questions:

1. How many meaningful colors are there in the image?
2. Where are the pixels with similar color approximately located?
3. How are these pixels spatially arranged?

The paper is organized as follows. We review the state-of-the-art in Section 2, while SCH are formally defined in Section 3. A suitable metrics for evaluating SCH similarity is defined and discussed in Section 4. Section 5 summarizes the experimental results, while we give our conclusions in Section 6.

## 2. Previous work

Color has been used for content-based image and video retrieval, probably because color features are quite easy to compute. In particular the color histograms of quantized images has been widely used to describe the image color content. The similarity between two color histograms is commonly measured with techniques based upon the Minkowski metric. Example of the use of histograms can be found in the work of Swain and Ballard [5], oriented towards pattern recognition for robot vision, and in the QBIC database system, [6] developed by IBM.

The above methods quantize the images over a fixed palette. Other authors [7,8,20] have investigated the feasibility of image-dependent color quantization/segmentation by clustering. The main issues in this case are the strategy adopted to predict the number of valid clusters/regions in the image, the computational cost, and the need of sophisticated measures for evaluating similarity. Rubner, Guibas and Tomasi have defined the distance between two color distributions as the minimum amount of work needed to transform one color distribution into the other [8]. Alternatively, Hausdorff distance, or modified Hausdorff distance can be applied.

In Ref. [9] the authors proposed a dependent scalar quantization (DSQ) algorithm in order to obtain a set of representative colors of the image. The DSQ approach partitions color space of an image in a dependent way in order to fully utilize the correlation of the color components of an image using the one-dimension moment-preserving technique. A dynamic matching method is then employed to compare the query images with the database images. Wan and Kuo [10] proposed to use color features obtained by hierarchical color clustering based on pruned octree data structure to achieve efficient image retrieval. The method proposed integrated into a single framework multiple color features such as dominant color number of distinct color, and the color histogram.

Unfortunately pixel histograms do not provide spatial

information about their arrangement, so very different images can have similar color distributions. Moreover, our experiments have shown that observers disagreed in evaluating color similarity, and that a set of similar images found by browsing the original images did not coincide with that obtained by browsing a randomized version of the database, in which the original image structure was changed, but not the color distribution [12].

To solve this problem many indexing techniques include a variable amount of spatial information. Stricker and Dimai, for example split an image into an oval central region and four corners. Their system evaluates and combines color features similarity for each of these sub-images, attributing more weight to the central region.

However, this is a strictly domain-dependent solution, which could be used for a stock photo house, but might not be acceptable in other applications. Smith and Chang [24] partition an image in regions using a sequential labeling algorithm based on the selection of a single color or a group of colors. After eliminating spurious artifacts using a thresholding method, for each region they compute a binary color set using histogram backprojection.

Pass and Zabih [26] described a split histogram called *Color Coherence Vector* (CCV). Each one of its buckets  $j$  contains pixels having a given color  $j$  and two classes based on the pixels spatial coherence. A pixel is coherent if the size of its connected component exceeds a threshold  $\tau$ ; otherwise, the pixel is incoherent. The feature is also extended by *successive refinement*, with buckets of a CCV further subdivided on the base of additional features.

Hsu et al. [22] proposed a three steps technique: first they select a set of representative colors. Then spatial knowledge of the selected pixels having a given color is obtained using a maximum entropy discretization that partitions the image in rectangular regions having a predominant color. The last step, the retrieval phase, is performed considering the degree of overlap between regions having the same color.

Huang et al. [23] described the use of color *correlograms* to integrate color and spatial information. They set a number  $n$  of inter-pixels distances and, given a pixel of color  $c_i$ , define a correlogram as a set of  $n$  matrices  $\gamma^{(k)}$ , where  $\gamma_{c_i, c_j}^{(k)}$  is the probability that a pixel at distance  $k$  away from the given pixel is of color  $c_j$ . A simplification of this feature is the autocorrelogram  $\alpha_c^{(k)}$ , which captures spatial correlation only between identical colors.

## 3. Image description index using SCH

### 3.1. Color quantization

The effective and efficient computation of the indices requires a drastic reduction in the number of colors used to represent the color contents of our 24-bit images.

Our quantization method [14] exploits the partition of the gamut of feasible colors in equivalence classes corresponding


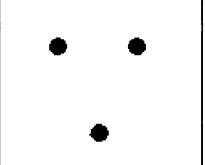


			
$\mathbf{b} = (0.495, 0.479)$	$\mathbf{b} = (0.479, 0.474)$	$\mathbf{b} = (0.0506, 0.061)$	$\mathbf{b} = (0.655, 0.741)$
$\sigma = 0.0369$	$\sigma = 0.325$	$\sigma = 0.25$	$\sigma = 0.25$

Fig. 1. Pixels spatial arrangement examples

to standardized linguistic tags. This method partitions the CIELAB color space into 256 subspaces (categories), in each of which the color remains perceptually the same, is labeled with a unique linguistic tag, and is distinctly different from that of neighboring subspaces. The color stimuli representing the color categories are derived from the ISCC-NBS color naming system proposed in 1955 by the Inter-Society Color Council and the National Bureau of Standards [31].

The ISCC-NBS color naming system partitions the Munsell color system into 267 blocks, each named in terms of its hue, lightness and saturation. Each block contains, on the average, some 40,000 discriminable color stimuli, and is represented, besides its name, by the Munsell coordinates of the centroid of the color block.

These centroids are determined by graphical interpolation and do not in general correspond to the center of mass of the blocks. As the blocks also have an irregular shape and vary greatly in size, mapping from the Munsell to ISCC-NBS must be performed by means of a look-up table. And since mapping from CIE coordinates and Munsell notations, and vice versa, also requires the use of three-dimensional interpolation techniques applied to a data table, conventional conversions for the transformation CIELAB  $\rightarrow$  Munsell  $\rightarrow$  ISCC-NBS would not be computationally convenient [31].

We have, therefore, defined a feed-forward neural network, trained by error back-propagation, which efficiently performs the transformation from the CIELAB color system to the ISCC-NBS color naming categories. The ISCC-NBS system, which classifies colors with words (in English) labeling the dimensions of hue (red, orange, yellow, green, etc.), saturation (grayish, moderate, strong and vivid) and lightness (very dark, dark, medium, light, and very light), can provide, the authors believe, a reliable basis in the future for the automatic creation of a detailed linguistic description of the indexed image. The images quantized by the proposed method are further clustered into a set of 11 equivalent classes representing basic color terms [31,32] (black, gray, white, red, orange, yellow, green, blue, purple, pink and brown) to produce coarse, but completely unsupervised image segmentation.

More formally, Let  $I$  be an  $n \times m$  image. Colors in  $I$  are reduced by a quantization method with respect to a palette of  $c$  colors. We define  $C$  a color space and  $P = \{c_0, c_1, \dots, c_{n-1} : c_i \in C\}$  a subset of  $C$ , having  $\|P\| \ll \|C\|$ .

The quantization process is represented by a quantizer  $Q_c$ , a function that maps each color in the color space  $C$  in an element of the  $P$  set. The quantizer is defined in the following way:

$$Q_c : C \rightarrow P \quad (1)$$

The application of the color quantizer produces a coarse, but completely unsupervised image segmentation. To reduce the quantization noise preserving edges and chromaticity contents (no new colors are added), a vector median filter is finally applied to the segmented image [25]. In formula, for each pixel in the image, the output of the filter is that pixel — indicated as  $x_{vm}$  — in a  $3 \times 3$  window which minimizes the sum of the distance from all the other pixels in the window ( $x_1, \dots, x_9 \in W$ ) :

$$x_{vm} = \arg \min_{x_j \in W} \sum_i \|x_i - x_j\| \quad (2)$$

### 3.2. Features extraction

Let  $I$  be our quantized image as described above, we define  $A_k^I := \{(x, y) \in I : I[x, y] = k\}$  as the set of pixels of  $I$  having the same color  $k$ . SCH are  $c$ -entries vectors where each entry  $j$  contains (a) the number of pixels having color  $j$ , (b) information about their spatial arrangement within the image. Before giving a definition of SCH we need to describe the features associated and extracted from the image. First we define  $h_I(k)$  as the ratio of pixels having color  $k$  in image  $I$ :

$$h_I(k) = \frac{|A_k^I|}{n \times m} \quad (3)$$












next, we compute the baricenter, given in relative coordinates, of the pixels in  $A_k^I$ ,  $\mathbf{b}_I(k) := (\bar{x}_k, \bar{y}_k)$  : where

$$\bar{x}_k = \frac{1}{n} \frac{1}{|A_k^I|} \sum_{(x,y) \in A_k^I} x \quad (4)$$

$$\bar{y}_k = \frac{1}{m} \frac{1}{|A_k^I|} \sum_{(x,y) \in A_k^I} y \quad (5)$$

The baricenter gives an idea of the position of the pixel having the same color, however it is still a rough approximation of the pixels' spatial properties, because different arrangements can have close baricenters. To refine spatial

Table 1  
Similarity examples

	Color	$h$	$\bar{x}$	$\bar{y}$	$\sigma$	
Query image						
	White	0.89	0.514	0.490	0.266	
	Red	0.11	0.338	0.535	0.276	
$Q$						
Image						
	SCH					Similarity
	White	0.89	0.474	0.490	0.266	1.937
	Red	0.11	0.662	0.169	0.276	
$T_1$						
	White	0.9	0.493	0.499	0.286	1.844
	Red	0.1	0.509	0.459	0.127	
$T_2$						
	White	0.933	0.494	0.499	0.284	1.792
	Red	0.067	0.506	0.452	0.731	
$T_3$						
	White	0.885	0.522	0.496	0.265	1.765
	Green	0.115	0.285	0.484	0.266	
$T_4$						

information about the regions we compute the standard deviation of pixels in  $A_k^I$ . Let  $\mathbf{p}$  be a generic pixel in  $A_k^I$  with relative coordinates  $(x_p, y_p)$  we will define such standard deviation as:

$$\sigma_I(k) = \sqrt{\frac{1}{|A_k^I|} \sum_{\mathbf{p} \in A_k^I} d(\mathbf{p}, \mathbf{b}_I(k))^2} \tag{6}$$

where  $d(\mathbf{p}, \mathbf{b}_I(k))$  is the euclidean distance between two pixels.

The standard deviation gives an idea of the pixels *spread* from the baricenter. Given  $I$  and the three previous features we can now define the SCH  $S_I$  as a vector of  $c$  elements (where  $c$  is the number of colors):

$$S_I(k) = (h_I(k), \mathbf{b}_I(k), \sigma_I(k)) \tag{7}$$

and  $1 \leq k \leq c$  identifying each single color.

### 3.3. Features properties

Fig. 1 shows examples of pixels arrangements: the first two (a single circular blob and three circular blobs respectively, all having constant, common color), have spatially close baricenters, with different standard deviations. Conversely, in the last pair of images we can observe two arrangements with the same  $\sigma$ , but with differently placed baricenters. Because the features are normalized they are insensitive to scale changes; a useful property in a heterogeneous collection of images. SCH is also a compact representation of the spatial-color image content and, moreover, can be computed rapidly.

These two properties are particularly desirable when the database size increase with time.

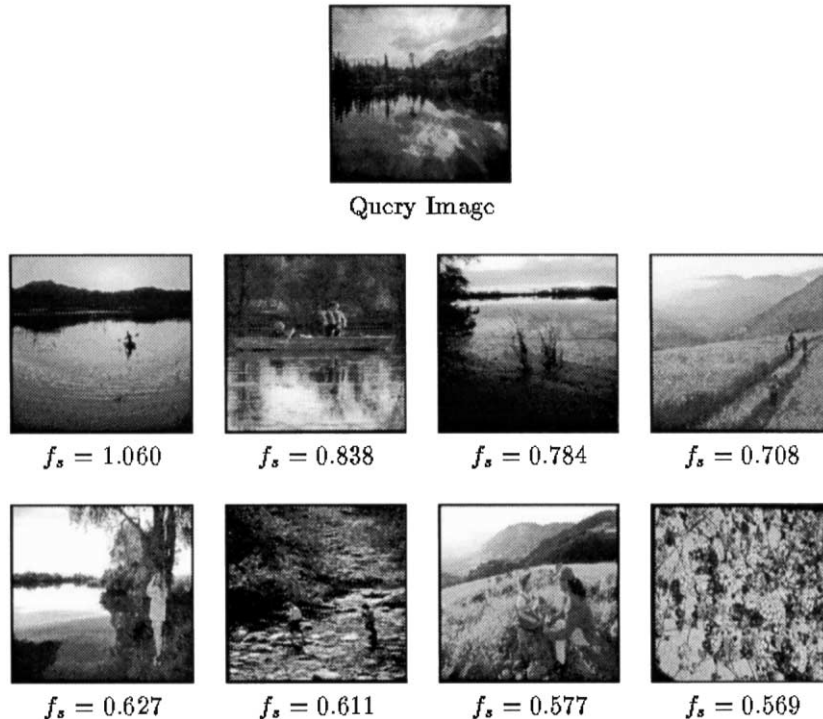


Fig. 2. Sunset query results and their similarity value

## 4. Evaluating color similarity

### 4.1. Similarity function

Feature vectors extracted from images are usually compared using  $L_1$  and  $L_2$  distance functions.<sup>1</sup>

There are, of course, examples of more sophisticated metrics like that in [6], where a quadratic form function takes accounts for similarity in any pair of colors.

For our purposes we designed a similarity function  $f_s$  that separately considers color and spatial information.

Given a query image  $Q$  and another image  $I$  in the database, we extract their SCH  $S_Q = (h_Q, \mathbf{b}_Q, \sigma_Q)$  and  $S_I = (h_I, \mathbf{b}_I, \sigma_I)$  and compare them by means of the following formula:

$$f_s(Q, I) = \sum_{i=1}^c \min(h_Q(i), h_I(i)) \times \left( \frac{\sqrt{2} - d(\mathbf{b}_Q(i), \mathbf{b}_I(i))}{\sqrt{2}} + \frac{\min(\sigma_Q(i), \sigma_I(i))}{\max(\sigma_Q(i), \sigma_I(i))} \right) \quad (8)$$

where  $c$  are the points of the discretized color space (see Section 5). The weighting function for  $f_s$  enhances the spatial component which has large populations of equally colored pixels; such spatial component increases whenever baricenters are closely located.

Table 1 are displayed examples of similarity results. The query image  $Q$  is compared with each of the other images and the similarity values are shown in the rightmost column. Image  $T_1$  has the highest similarity score because only one region is mirrored, but color percentage is the same of the query. Images  $T_2$  and  $T_3$  have different red regions arrangements are different white and red pixels percentage, so are evaluated consequentially. Image  $T_4$  has similar spatial arrangement as the query, but it has different color, red vs. green, so it has a lower similarity score.

In a preliminary work [27], we performed different tests to evaluate SCH-based indexing effectiveness on a database of 300 images of natural scenes which comes from the Electronic Library Project [28].

A query example using SCH is shown in Fig. 2, where a sunset picture used as query and the first eight most similar pictures retrieved, using a 64 colors quantized space, are presented. The captions contain the similarity score  $f_s$  with respect to the picture query. The pictures retrieved are plausible: the first four surely contain similar color and spatial properties, while the next four, even if with different subjects, have also some common colors. More specifically pictures with

a sunset sky in the upper region, mountains in the middle and water in the lower region are presented first, while other pictures, even with similar color content, have different spatial arrangements.

## 5. Experimental results

In order to test the performances of SCH-based indexing methodology, we compared it with the CCV-based one, performing different experiments.

### 5.1. Experimental setup and database

For our tests we used a database containing 3000 still images representing various subjects and scenes like flowers, animals, landscapes, people and so on.

A subset of 25 images, randomly chosen from the database, were used as a query image, and the system performs the similarity evaluation with respect to each query image has been assessed jointly by authors.

When a query is submitted, the system we implemented reorders the database images by similarity with respect to the query and then returns to the user the most similar images. We consider an image that is truly similar to the query correctly retrieved if it appears within the first set of displayed images, composed of 24 elements.

### 5.2. Results evaluation and discussion

In order to quantify our methodology's performances with respect to CCV we need a measure of *how much* the ordered list of images contained in the database resembles the human similarity order. Performance measures for text retrieval have been extensively studied [29], and some of these methods can be adapted to image content-based retrieval [30].

In our paper, we have evaluated the performance of the SCH and compared it with those of CCV using a measure called Effectiveness (Efficiency of Retrieval or Fill Ratio). This measure, proposed by Mehtre et al. [20], has also been applied recently in the comparison of shape similarity measures [16] and color similarity measures [12] in content-based image retrieval.

We let  $S$  be the number of relevant items the user wanted to retrieve when posing a query (short list)  $\mathbf{R}_q^I$ , is the set of relevant images, and  $\mathbf{R}_q^E$ , the set of relevant

Table 2  
Retrieval efficiency table

	Retrieval Efficiency		
	CCV64	SCH64	SCH11
Min	0.090909	0.181818	0.142857
Mean	0.319073	0.414057	0.472625
Max	0.52381	0.714286	0.761905

<sup>1</sup> Given a metric space  $D$ , a  $L_p$  distance function on  $D$  is defined as  $\sum_{k=1}^n |x_k - y_k|^p / p$ , for any  $p \geq 1$  and  $x_k, y_k \in D$ .  $L_1$  distance is the classical city-block and  $L_2$  is the Euclidean distance. For a more complete overview about similarity measures for image databases see Ref. [21].

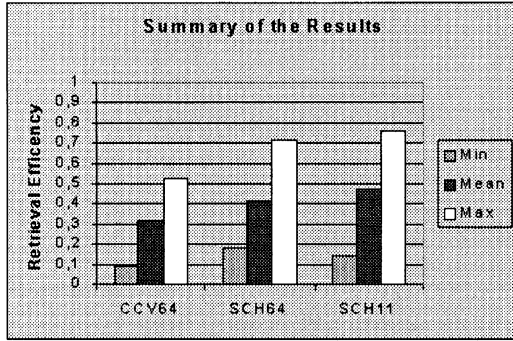


Fig. 3. Summary of the experimental results

images retrieved in the short list. The effectiveness measure is defined as:

$$\eta_S = \begin{cases} \frac{|R_q^I \cap R_q^E|}{|R_q^I|} & \text{if } |R_q^I| \leq S \\ \frac{|R_q^I \cap R_q^E|}{|R_q^E|} & \text{if } |R_q^I| > S \end{cases}$$

If  $|R_q^I| \leq S$ , the Effectiveness is reduced to the traditional Recall measure, while if  $|R_q^I| > S$ , the Effectiveness corresponds to Precision.

The query set used in evaluating the methods performance is composed by 25 randomly chosen images belonging to our database. Ground truth similarity has been assessed jointly by two paper’s authors — for all the queries, applying, in turn, the following retrieval methods:

- SCH11 Spatial-Chromatic Histogram, where the colors of the images are quantized into 11 colors according to our quantization method;

- SCH64 Spatial-Chromatic histogram where quantization is performed as follows: the RGB color cube is uniformly samples into  $16 \times 16 \times 16$  color bins and then clustered into 64 colors using a  $K$  — means clustering algorithm.
- CCV64. The Color Coherence Vectors (CCV) in the color space quantized in 64 colors as above as above. CCV buckets color pixels as coherent or incoherent according to whether or not they belong to regions whose size exceeds 1% of the image size are counted as coherent pixels). Before CCV computation the image is blurred by local averaging in a  $3 \times 3$  neighbor.

Table 2 and Fig. 3 summarized the experimental results. We reported Min, Mean and Max values for Retrieval Efficiency, each of them for any of the indexing technique we used. We can observe that the Mean value of  $\eta_S$  is higher using SCH than CCV; the best value is obtained with a palette of 11 colors, but also using 64 colors we still have better efficiency with respect to the CCV with equal number of colors. To have a more detailed comparison we need also to compare minimal values, because an isolated optimal results can remarkably improve a generally poor performance. As we can see in the same table minimal values of  $\eta_S$  still highlight a good behavior of SCH-based indexing compare with CCV-based one. The histogram based plot in Fig. 3 shows same results graphically.

In order to present a qualitative example of our experiments we show the results of an example query image having a centermost white flower; the three images show results using respectively SCH with 11 colors palette, SCH using 64 colors palette and CCV using 64 colors palette. We can observe that using SCH the short list contains more white flowers and objects than the one obtained by the use of CCV (Figs. 4–6).

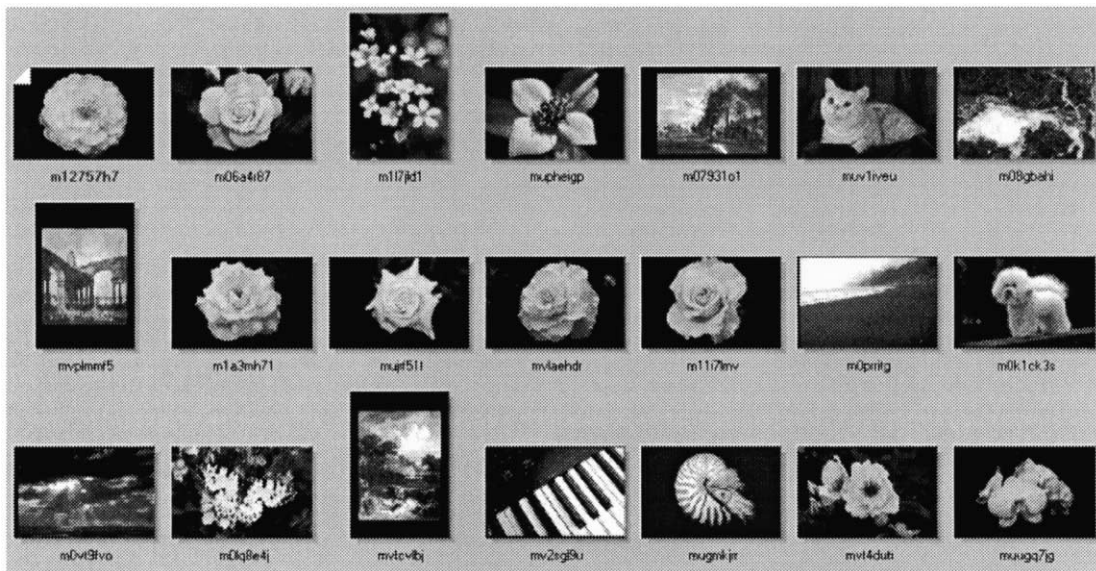


Fig. 4. Example query using SCH with 11 colors palette

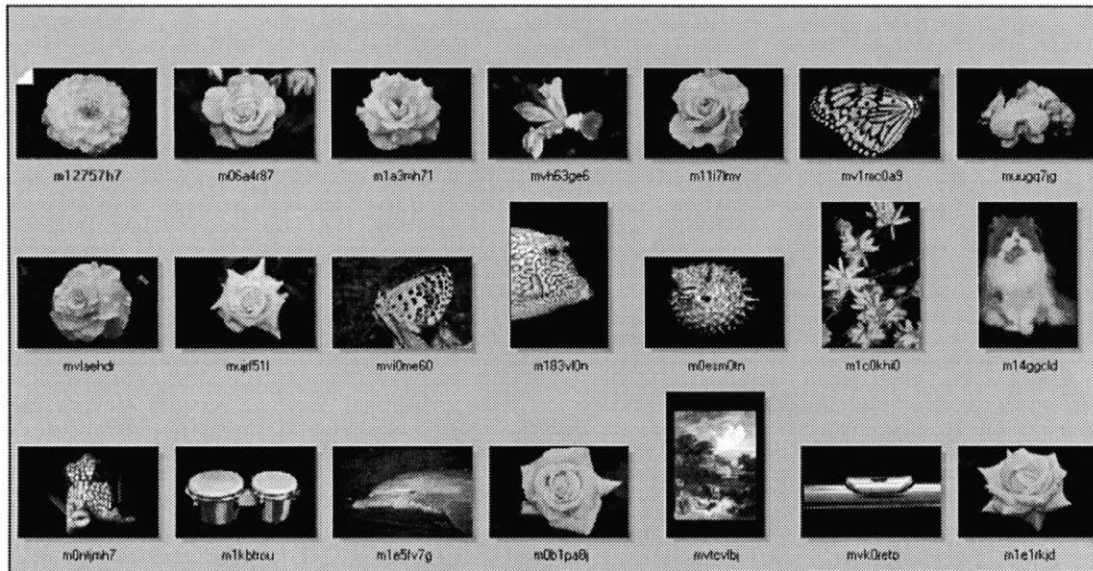


Fig. 5. Example query using SCH with 64 colors palette

**6. Conclusions and future work**

The research presented here focused on one pictorial feature, color, which, together with shape, texture, size, distance and relative position, characterizes scenes and pictures. We have introduced an image retrieval algorithm based on SCH that makes it possible to take into account spatial information in a flexible way without drastically increasing the computation cost.

The method proposed in this paper has several advantages if compared with other existing ones. Its main features are: (a) it exploits a coarse color quantization on a limited palette of salient colors that makes it

robust with respect to color image variations and image artifacts and noise, (b) it takes into account both chromatic and spatial features in an efficient and effective way, and (c) the similarity function designed makes it possible to perform partial image queries such as: search for images having a blue region on the top of it, and a green region on the bottom.

As future research we plan to extend the proposed metric to deal with image segmented by adaptive image quantization algorithms such as [9,10]. However, color alone cannot suffice to index large, heterogeneous image databases. The combination of color with other visual features is therefore an approach that merits further study [33].

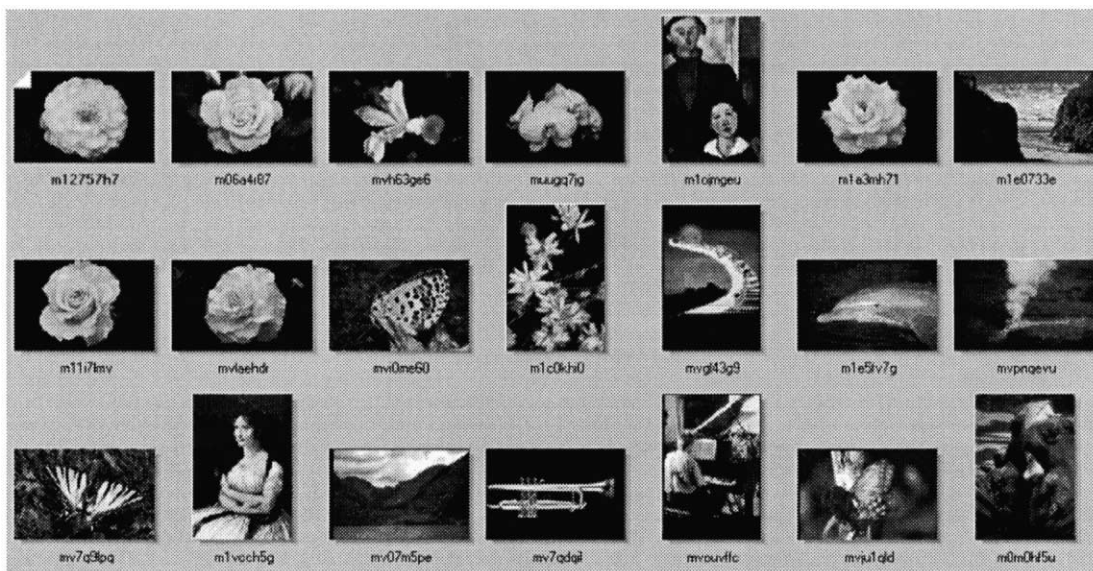


Fig. 6. Example query using CCV with 64 colors palette

## References

- [1] O.H. Aigrain, H. Zhang, D. Petkovic, Content-based representation and retrieval of visual media: a state-of-the-art review, *Multimedia Tools and Applications* 3 (1996) 179–182.
- [2] V.N. Gudivada, V.V. Rahavan, Modeling and retrieving images by content, *Information Processing Management* 33 (4) (1997) 427–452.
- [3] G.L. Gimel'Farb, A.K. Jain, On retrieving textured images from an image database, *Pattern Recognition* 29 (1996) 1461–1483.
- [4] R.W. Picard, T.P. Minka, Vision texture for annotation, *Multimedia Syst.* 3 (1995) 3–14.
- [5] M.J. Swain, D.H. Ballard, Color Indexing, *Int. J. Comput. Vision* 7 (1) (1991) 11–32.
- [6] J. Hafner, H.S. Sawhney, W. Equitz, M. Flickner, W. Nyblack, Efficient color histogram indexing for quadratic form distance functions, *IEEE Trans. PAMI* 17 (7) (1995) 729–736.
- [7] E. Binaghi, I. Gagliardi, R. Schettini, Image retrieval using fuzzy evaluation of color similarity, *Int. J. Pattern Recognition Artificial Intelligence* 8 (1994) 945–968.
- [8] Y. Rubner, C. Tomasi, L.J. Guibas, A metric for distributions with applications to image databases, *Proc. of the IEEE Int. Conf. on Computer Vision* pp. 59–66, (1998).
- [9] S.-C. Pie, C.M. Cheng, extracting color features and dynamic matching for image data-base retrieval, *IEEE Tran. Circuits Syst. Video Technol.* 9 (3) (1999) 501–512.
- [10] X. Wan, C.-C.J. Kuo, A new approach to image retrieval with hierarchical color clustering, *IEEE Tran. Circuits Syst. Video Technol.* 8 (5) (1998) 628–643.
- [11] A. Del Bimbo, P. Pala, Visual image retrieval by elastic matching of user sketches, *Tran. Pattern Anal. Machine Intelligence* 19 (1997) 121–132.
- [12] B. Barolo, I. Gagliardi, R. Schettini, An effective strategy for querying image databases by color distribution, *Comput. History Art J.* 7 (1) (1997) 3–14.
- [13] A. Della Ventura, I. Gagliardi, R. Schettini, Indexing color-texture image patterns, *Adv. Comput. Vision Machine Intelligence* (1998) 105–117.
- [14] I. Gagliardi, R. Schettini, A method for the automatic indexing of color images for effective image retrieval, *New Rev. Hypermedia Multimedia* 3 (1997) 201–224.
- [15] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petrovic, Efficient and effective querying by image content, *J. Intelligent Syst.* 3 (1994) 231–262.
- [16] B.M. Mehtre, M.S. Kankanhalli, Content based image retrieval using composite color shape approach, *Inform. Process. Management* 34 (1) (1998) 109–120.
- [17] J.R. Smith, S.-F. Chang, Visual SEEK: a fully automated content-based image query, *Fourth ACM Multimedia* (1996) pp 87–98.
- [18] G. Ciocca, I. Gagliardi, R. Schettini, Quick-look: a content-based retrieval system with learning capability, *Proc. IEEE Multimedia* (1999) 1028–1029.
- [19] T. Minka, R.W. Picard, Interactive learning with a Society of Models, *Pattern Recognition* 30 (4) 565–581 (1997).
- [20] B.M. Mehtre, M.S. Kankanhalli, A. Desai Narasimhalu, G.C. Man, Color matching for image retrieval, *Pattern Recognition Lett.* 16 (1995) 325–331.
- [21] R. Jain, S.N.J. Murthy, P.L.-J. Tran, S. Chatterjee, Similarity measures for image databases, *FUZZ-IEEE '95* (1995) Vol 3 pp 1247–1254.
- [22] W. Hsu, T.S. Chua, H.K. Pung, An integrated color-spatial approach to content-based image retrieval, *ACM Multimedia Conf.* (1995) 305–313.
- [23] J. Huang, S.R. Kumar, M. Mitra, W.-J. Zhu, R. Zabih, Image indexing using color correlograms, *IEEE Conf. Comput. Vision Pattern Recognition* (1997) 762–768.
- [24] J.R. Smith, F.S. Chang, Tools and Techniques for Color Image Retrieval, *Symposium on Electronic Imaging: Science and Technology — Storage and Retrieval for Image and Video Database IV*, vol. 2670 1996 pp. 426–237.
- [25] J. Astola, P. Haavisto, Y. Neuvo, Vector median filters, *Proc. IEEE* 78 (1990) 678–689.
- [26] G. Pass, R. Zabih, Histogram refinement for content-based image retrieval, *IEEE Workshop Applications of Comput. Vision* (1996) 96–102.
- [27] L. Cinque, S. Levialdi, A. Pellicano, K. Olsen, Color based image retrieval using SCH, *IEEE Multimedia Conf.* (1999) 969–973.
- [28] Electronic Library Project, <http://clib.cs.berkeley.edu>
- [29] M.J. McGill, G. Salton, *Introduction to Modern Information Retrieval*, McGraw-Hill, New York, 1983.
- [30] A. Desai Narasimhalu, M.S. Kankanhalli, Benchmarking multimedia databases, *Multimedia Tools Applications* 4 (1997) 333–356.
- [31] G.A. Agoston, *Color Theory and its Applications in Art and Design*, Springer, Berlin, 1987.
- [32] K.L. Kelly, D.B. Judd, *Color: Universal Language and Dictionary of Names*, NBS Special Publication 440, US Government Printing Office, Washington DC, 1976.
- [33] G. Ciocca, I. Gagliardi, R. Schettini, Quicklook2: An Integrated Multimedia system, *J. Visual Languages Comput.* 12 (1) (2001) 81–103.