A SURVEY OF METHODS FOR COLOUR IMAGE INDEXING AND RETRIEVAL IN IMAGE DATABASES

Raimondo SCHETTINI, Gianluigi CIOCCA, Silvia ZUFFI

Istituto Tecnologie Infomatiche Multimediali
Consilio Nazionale delle Ricerche
Via Ampere 56, 20131 Milano
ITALY
[centaura, ciocca, zuffi]@itim.mi.cnr.it

ABSTRACT

Color is a feature of the great majority of content-based image retrieval systems. However the robustness, effectiveness, and efficiency of its use in image indexing are still open issues. This paper provides a comprehensive survey of the methods for color image indexing and retrieval described in the literature. In particular, image preprocessing, the features used to represent color information, and the measures adopted to compute the similarity between the features of two images are critically analyzed.

Keywords: *Color Indexing, Image Retrieval, Color Similarity, Image Database, Color Features.*

1 INTRODUCTION

Color has been widely used for content-based image and video retrieval in multimedia databases. Much research has been devoted in recent years to the definition of effective and efficient tools for specifying visual queries and implementing retrieval strategies that satisfy some criteria of matching or pictorial similarity. The use of color has been extensively experimented in

1. color matching, to find images containing specified colors in assigned proportions, e.g. [8];
2. similarity searches, to find a ranked list of images "similar" to an image provided or hand sketched by the user, e.g. [3] [33];
3. region searches, to find images containing regions of color as specified in a query , e.g. [39], [69];
4. target searches, to find a list of the images in which an object specified by the user appears[58][71];
5. semantic categorization, to group images in meaningful categories, such as graphics as opposed to photos, or indoor as opposed to outdoor pictures, e.g. [79];
6. retrieving images with certain color-induced effects, e.g. [15].

All these tasks depend upon the definition of robust and efficient color features that can represent image contents. Unfortunately, there is no single "best" representation of color, but only multiple representations which characterize the color feature from different perspectives. In any given context, however, the selected features will ideally present the following basic properties:

1. perceptual similarity: the feature distance between two images is large only if the images are not "similar";
2. efficiency: they can be rapidly computed;
3. economy: their dimensions are small in order not to affect retrieval efficiency;

4. scalability: the performance of the system is not influenced by the size of the database;
5. robustness: changes in the imaging conditions of the database images do not affect retrieval.

In matching images, a feature similarity/dissimilarity function must be coupled with the color features. A distance d is defined as $d : \Im \times \Im \rightarrow \Re^{+}$ and must satisfy the following four properties for all images I, J and K in $\Im$:

$$
\begin{array}{ll}
P_1 : d(I,I) = d(J,J) & \text{self} - \text{similarity} \\
P_2 : d(I,J) \geq d(I,I) & \text{minimality} \\
P_3 : d(I,J) = d(J,I) & \text{symmetry} \\
P_4 : d(I,K) + d(K,J) \geq d(I,J) & \text{triangular inequality}
\end{array}
$$

Any function satisfying P_1, P_2 and P_4 is a metric. Any function satisfying P_1, P_2 and P_3 is a similarity measure.

A schematic description of the activities of a visual information retrieval system (VIR) is shown in Figure 1. During input, images are processed to compute the features selected to represent the image contents. This process, called *indexation* or *indexing,* assigns to each image a set of identifying descriptors, or indices, which will be used by the system in the matching phase to retrieve relevant images and reject extraneous ones. The indices are stored in the database, ideally are designed for efficient retrieval. Different features (color, shape, texture, size, distance, relative position, etc.) express different aspects of image contents, and may, of course, coexist [14]. Only color-based features are considered here.

When an image query is posed, its color features are extracted from the database, or computed using the same procedures applied to input images. Image retrieval is then performed by a matching engine, which compares the features of the query image with those of the stored images. The matching mechanism implements the retrieval according to the selected metric, or similarity measure. The images of the database are ranked according to their similarity/match with the query, for evaluation by the user according to his information needs.

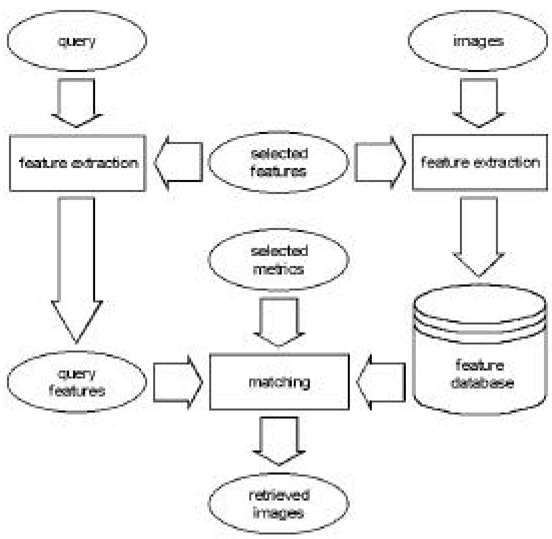

Figure 1. Schematic description of the activities of a VIR system.

With a very few exceptions, the effective and efficient computation of color indices requires a drastic reduction in the number of colors used to represent the color contents of an image. The algorithms employed for this are reviewed in Section 2. Once the number of colors has been sufficiently reduced, many different strategies for representing and comparing color distributions can be used. These strategies are described in Section 3. Image retrieval based purely on color distribution tends to include too many false positives when the database is large. The several possible extensions of global color features to code local spatial information are described in Sections 4, 5, and 6. In Section 7 the integration of color-based features with other features is discussed. Our conclusions are presented in Section 9, together with an indication of what we see as the road map for future research.

2 COLOR DISCRETIZATION

The drastic reduction in the number of colors used to represent the color image content that effective and efficient computation of color indices usually demands is in general achieved by color space quantization, using a predefined color palette (static quantization), or by clustering and/or spatial segmentation (dynamic quantization). Formally, we let C be a color space, and $P = \{c_1, c_2, ..., c_i, ..., c_n \mid c_i \in C, n << \|C\|\}$, a subset of C called the quantization space. A function Q that maps each color in C to an element in P is called a *quantizer,* and is defined as:

$$Q : C \rightarrow P$$

A schematic representation of the different quantization methods is given in Table 1.

Color Discretization				
Static				Dynamic
Most Significant Bits	Color Space Partition	Color Space Clustering	Reference Colors	Image Segmentation

Table 1: Quantization methods.

Several authors have used only a few (generally two) of the most significant bits of each of the R, G, and B color channels to severely reduce the number of image colors [55][59]. But the lack of perceptual rules for color mapping may cause considerable shifts in color.

Smith and Chang [69] have partitioned the HSV color space into 166 bins, placing more importance on hue (18 levels) than on value and saturation (three levels each). Before computing the image index, a median filter is applied on each HSV color component to eliminate outliers and emphasize prominent color regions.

The QBIC system [33] makes it possible to compute a k element color histogram, where k can be set by the user (the default value is 64). Each R, G, and B color axis is initially quantized in 16 levels, obtaining an initial partition of the RGB color space into 4096 cells. The coordinates of the center of each cell in a Modified Munsell color space [54] are then computed, and a standard, greedy minimum sum of square clustering is performed to obtain k "super-cells". A similar partition has been applied in the HSV color space by Jain et al.[54][79].

Ciocca et al. have quantized the device color space in two steps. First, a random sampling of few million colors is generated in the RGB color space. These samples, assuming that they are coded in sRGB terms, are mapped into the CIELAB color space, where a competitive cluster algorithm [78] is then applied to find the 64 most significant colors. The colors of the images to be indexed are mapped in the CIELAB color space, and assigned to the nearest of the 64 centroids [14].

Syeda-Mahmood [70] has proposed a quantization method that partitions the RGB color space into about 220 subspaces (categories) in which the color remains perceptually the same and distinctly different from that of neighboring subspaces. This partition was obtained by a "rather informal but extensive psychophysical experiment", which systematically examined the device-dependent HSV color space. A look-up table was used for the mapping between the RGB values and the color categories.

Gagliardi and Schettini [29] have proposed the use of multiple descriptions of color in order to deal with the intrinsic variability in human evaluation of color similarity. Their quantization method partitions the gamut of feasible colors into equivalence classes corresponding to standardized linguistic tags. The CIELAB color space is divided into 256 subspaces (categories), in each of which the color remains perceptually the same, is labeled with its own 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 The images quantized by this method are further clustered in a set of 13 equivalent classes representing basic color terms (black, gray, white, red, orange, yellow, green, blue, violet, purple, pink, brown, and olive) to produce a coarse, but completely unsupervised image segmentation.

In their ImageRover system [67] Sclaroff et al. have first transformed the RGB values into the CIELUV space, after which each color axis is split into 4 equal size bins for a total number of 64 bins (theoretical, since very dark and very light colors can not have a high chroma).

In order to achieve a more compact representation of color images, Mehhre et al [50] have heuristically defined a small set of RGB reference colors in which all the colors in the application (trademark indexing) are approximately covered. Each pixel of the color image is simply assigned to the nearest color in table. This type of discretization must be performed very carefully, and requires a high level of interaction. When a large subset of the image database is available, the color space can, instead, be partitioned so that each discrete color appears with approximately equal likelihood in order to maximize the information conveyed [37].

Gong et al. [32] have roughly partitioned the Modified Munsell color space (the same used by QBIC) into eleven color zones defined and validated empirically by different groups of examiners. A similar partition has been proposed by Cox et al. in the PicHunter system in HSV [16], and by Ciocca et al. in the Quicklook system [14].

Other authors [8][50][64] 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. The resulting image signature may represent the color image content very well. However, as is well known, unsupervised image segmentation is a ill-posed problem: when little is known about the images to be indexed and the clustering-segmentation parameters can not be tuned accordingly, the risk of serious errors should not be underestimated.

Liu and Yang have proposed a function that does not require any user-set parameter or threshold values for the quantitative evaluation of the performance of algorithms for the segmentation of color images [43]. The function has been designed to incorporate, directly or indirectly, three out of the four heuristic criteria suggested by Haralick and Shapiro [34] for evaluating segmentation results without having to set threshold values for any of the subjective properties of region size, shape, or homogeneity. The incorporated criteria are: 1) the regions must be uniform and homogeneous, 2) the interior of a region must be simple, without too many small holes, and 3) adjacent regions must present significantly different values for uniform characteristics. The authors, commenting on their experimental results, suggest that their function also takes into account, although indirectly, the smoothness of the boundaries (part of the Haralick and Shapiro's fourth criterion, which includes boundary accuracy). The evaluation function is empirically defined as:

$$F(I) = \frac{1}{1000 \cdot (N \times M)} \cdot \sqrt{R} \cdot \sum_{i=1}^{R} \frac{e_i^2}{\sqrt{A_i}}$$

where I is the segmented image, $N ´M$ the image size, and R the number of regions of the segmented image, while A_i and e_i are, respectively, the area and the average color error of the i-th region; e_i is defined as the sum of the Euclidean distances between the RGB color vectors of the pixels of region i and the color vector attributed to region i in the segmented image. The smaller the value of $F(I)$, the better the segmentation result should be.

The equation is composed of three terms: the first is a normalization factor which takes into account the size of the image; the second, $\sqrt{R}$, penalizes segmentations that form too many regions; the third, the sum, penalizes segmentations with non-homogeneous regions. Since the average color error e_i of the region is significantly higher for large regions than for small ones, e_i has been scaled by the factor $\sqrt{A_i}$. The authors, who report a good match between function values and visual evaluation of the corresponding image segmentations, have used function F to automatically select the best segmentation, varying the color space and the dissimilarity measure employed in their algorithm. Since this function tends to evaluate very noisy segmentations favorably, Borsotti et al. [10] have proposed an enhanced function that corresponds more closely to visual judgment:

$$Q(I) = \frac{1}{10000 \cdot (N \times M)} \cdot \sqrt{R} \cdot \sum_{i=1}^{R} \left[\frac{e_i^2}{1 + \log A_i} + \left(\frac{R(A_i)}{A_i} \right)^2 \right]$$

where all the entities are as previously defined for F, while $R(A_i)$ represents the number of regions having an area equal to A_i. The body of the sum is composed of two terms: the first is high only for non-homogeneous regions (typically, large ones), while the second is high only for regions with an area A_i equal to the area of many other regions in the segmented image (typically, small ones). The design of this new term takes into account the fact that the number of regions of area A_i in a given image will probably be small if area A_i has an high value, in which case $R(A_i)/A_i$ contributes little to the sum. On the other hand, the number of regions of area A_i *may* be large, if the area A_i has a low value; here $R(A_i)/A_i$ contributes strongly to the sum. Heuristically it can be said that $R(A_i)$ is most always 1 for large regions, and may be much larger than 1 for small regions. In any case, the denominator A_i drastically forces the term $R(A_i)/A_i$ to near zero for large regions, and lets it grow for small regions.

A different approach has been proposed by Del Bimbo [19]. Image colors are first quantized on a palette of 128 reference colors by a competitive learning algorithm. Region segmentation is then obtained by iteratively aggregating uniform color patches, so as to minimize the following measures:

$$F = \sum_i \left(a \frac{1}{A_{R_i}} + bD_{R_i} + g \sum_J \frac{1}{D_{R_i - R_J}}\right)$$

where A_{R_i} is the area of the region R_i, D_{R_i} is a measure of color uniformity of region R_i, $D_{R_i - R_J}$ is a measure of the difference in color between R_i and its adjacent regions R_j, and a, b, g are control parameters. Color regions are iteratively aggregated until a minimum of F is found; the values of a, b, g are then adjusted, and the procedure repeated. The image segmentations produced by this process are then organized in a pyramidal schema.

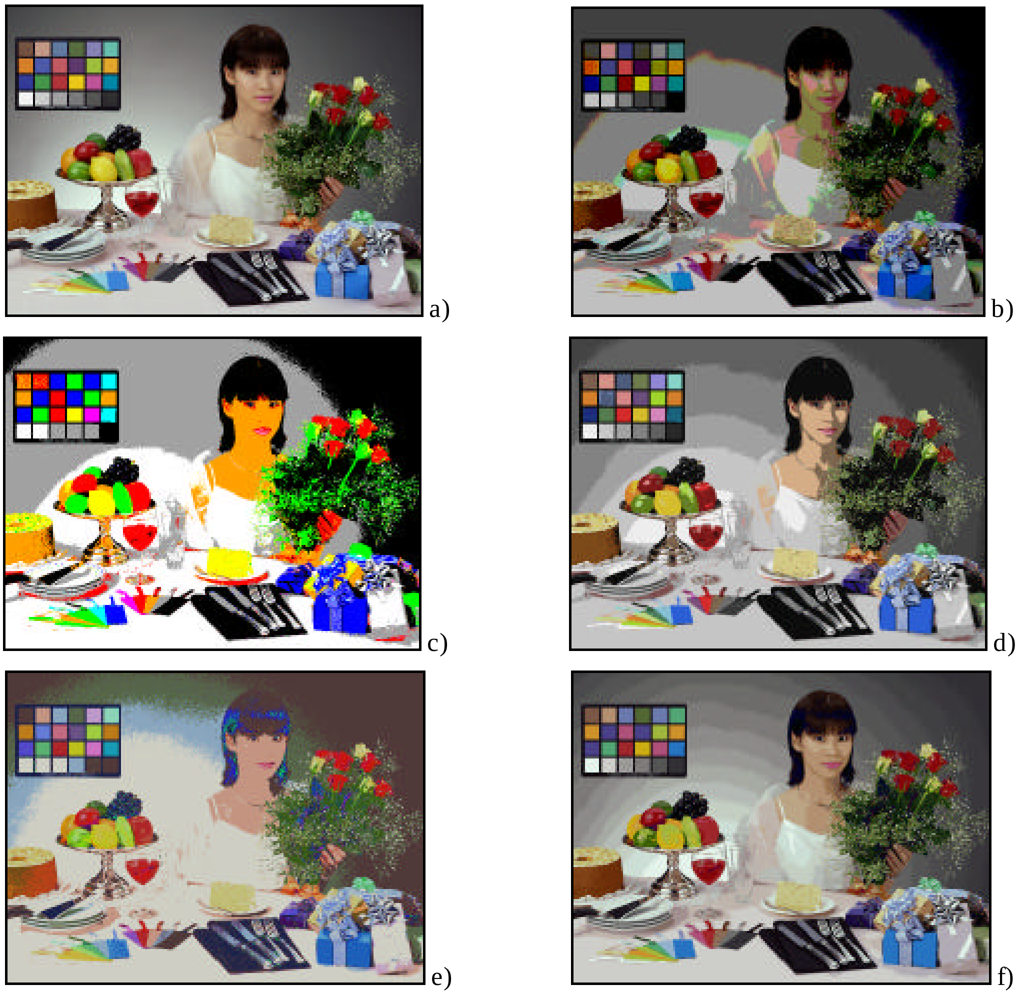

Figure 2. a) original image, b) image quantized using 2 bits per color channel, c) image quantized using eleven color classes corresponding to basic color names d) image quantization by partitioning the HSV color space into 64 bins, e) image quantized in 64 colors using color space clustering, f) image quantized in 64 colors using a segmentation algorithm.

3 COLOR INDICES

Once the number of color images has been significantly reduced, they can be coded in many different ways in order to represent the image content. Very often color is represented by histograms; in this case it is the matching strategy that most distinguishes the different retrieval methods. A color histogram, H, is a vector $[h_1, \ldots h_n]$ in which each bin h_j contains the number of pixels having the color j in the image and can be considered the probability density function (*pdf*) of the color values. If the images to be compared are of different sizes, but have been quantized on a common palette, their histograms can be compared as follows [75]:

$$D(H, H') = \frac{\sum_i min(h_i, h'_i)}{\sum_i h'_i}$$

If the images are of the same size (or the histograms have been scaled to the same size) and quantized on a common palette, their similarity is commonly measured using the sum of the squared differences (L_2 metric), or the sum of the absolute values of differences (L_1 metric). These metrics usually perform poorly, even for the simplest types of query (Figure 10).

To render the L-metrics more stable with respect to quantization (a slight change in lighting conditions may result in a corresponding shift in the color histogram, causing these metrics to misjudge similarity completely, as shown in Figure 3), Stricker and Orengo [72] have proposed the use of cumulative histograms. But to use these histograms the colors must first be ranked in the color space.

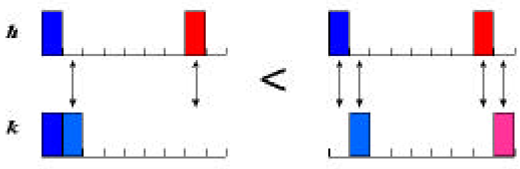

Figure 3. Misjudged similarity caused by color shift

Hafner et Al. [33] have suggested using a weighted distance between histograms that takes into account the "cross-talk" between colors. However, these authors were mainly concerned with the efficiency of the solution, and only marginally with the necessity of coding the perceptual similarity between colors. According to Hafner the distance between histograms H and H' is defined as:

$$d_H(H, H') = \sqrt{\sum_i \sum_j a_{ij} (h_i - h'_j)(h_i - h'_j)}$$

where a_{ij}, coding the similarity between the color i and j, is expressed as:

$$a_{ij} = 1 - \frac{d_{ij}}{\max_{ij}(d_{ij})}$$

and d_{ij} represents the Euclidean distance between the colors as defined in a variant of the Munsell color space. They have also proposed an alternative a_{ij}:

$$a_{ij} = \exp\left(-\sigma\left(d_{ij}/d_{max}\right)^2\right)$$

for some positive constant value of σ.

In earlier studies Binaghi et al. have exploited perceptual correlates of the psychological dimensions of Lightness, Chroma, and Hue, defined in the CIELAB-LUV color space to address color range and image indexing [8]. Using a structured interview technique these authors elicited three primary fuzzy sets corresponding to similarity in lightness (L*), hue (h*) and chroma (C*) color dimensions.

Each membership function defined assigned a degree of similarity between two colors i and j as a function of the difference in a given color feature, while the product of these three membership functions was used to define the global degree of similarity between the two colors considered. Extending these studies, the authors have defined the similarity between two histograms as:

$$d_F(H, H') = \sqrt{\sum_i \sum_j \mu_{h^*L^*C^*}(i, j)(h_i - h_j)(h'_i - h'_j)}$$

Weighted versions of the L_2 metric may underestimate distances because they tend to accentuate the similarity between color histograms presenting many non-empty bins. (see Figure 4) and are also computationally expensive. Hafner et al. [33] have proposed a simpler low-dimensional distance measure, called the average color distance, which can be used to perform a sort of prefiltering before applying histogram matching. An alternative strategy for increasing retrieval efficiency is described in [6].

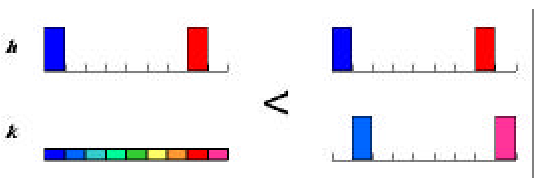

Figure 4. Overestimated similarity with histograms lacking well defined modes.

Stricker has proposed two other approaches more efficient than those based on color histograms, as they do not require color quantization, and produce more compact indices [73][74]. In the first, instead of storing the complete 3D color histogram, only the first three moments of the histograms of each color channel are computed and used as an index; in the second, the image is represented only by the average and covariance matrices of its color distribution.

The features of mean, variance and skewness can be computed for the i-th color channel as follows:

$$E_i(\tilde{I}) = \frac{1}{N}\sum_{x,y} \tilde{I}_i(x,y) \quad \text{that is, the average color channel values;}$$

$$ó_i(\tilde{I}) = \sqrt{\frac{1}{N}\sum_{x,y}\left(\tilde{I}_i(x,y) - E_i(\tilde{I})\right)^2} \quad \text{that is, the standard deviation;}$$

$$s_i(\tilde{I}) = \sqrt[3]{\frac{1}{N}\sum_{x,y}\left(\tilde{I}_i(x,y) - E_i(\tilde{I})\right)^3} \quad \text{that is, the third root of skewness.}$$

The similarity functions used in these approaches for retrieval is a weighted sum of the absolute differences between the features computed. Each feature entry is weighted by a value, $w_{i1}, w_{i2}, w_{i3} \geq 0$, selected by the user, depending upon the specific application (figure 11).

$$d_{mom}(\tilde{I}_1, \tilde{I}_2) = \sum_i \left(w_{i1}\left|E_i(\tilde{I}_1) - E_i(\tilde{I}_2)\right| + w_{i2}\left|ó_i(\tilde{I}_1) - ó_i(\tilde{I}_2)\right| + w_{i3}\left|s_i(\tilde{I}_1) - s_i(\tilde{I}_2)\right| \right)$$

If the images to be compared have been quantized on different color palettes, none of the above metrics can be applied.

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 [64]. Alternatively, Hausdorf distance, o modified Hausdorf distance can be applied. This provides the degree of mismatch between two color signatures (A, B) as the maximum distance between the colors of A and those of B:

$$H(A,B) = \max\left(\max_{a \in A} \min_{b \in B} d(a,b), \max_{b \in B} \min_{a \in A} d(a,b) \right)$$

Androutsos et Al. [2] have designed a combinatorial distance measure, based on the vector angle between two vectors. It allows queries based on single and multiple colors and, in addition the exclusion of certain colors.

All these methods neglect the spatial relationships of color pixels; consequently, images that are actually different in appearance may be judged similar simply because they have a similar color distribution. To further complicate things, observers disagree in their evaluations of color similarity, and a set of similar images found by browsing the original images may not coincide with that obtained by browsing the database version in which the color distribution has been preserved but the original image structure has been changed at random [3].

4 COLOR SPATIAL INDICES

The main weakness of all the indexing methods described above is the lack of spatial information in the indices. For example, all the patterns shown in Figure 5 have the same color proportions, but different spatial distributions. Their appearance is clearly quite different, so obviously we can not assume that color distribution always suffices to represent the pictorial content of an image.

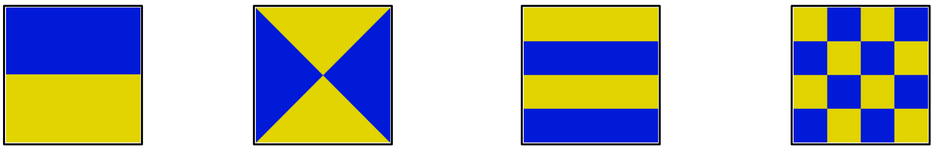

Figure 5. Some patterns having the same color proportions, but different spatial distributions.

Since in most applications complete segmentation implies a great deal of user interaction during database acquisition, this approach is not feasible for large image databases. The simplest way to provide spatial information is to divide the image into sub-images, and then index each of these. Gong et al.[32], for example, model the color-spatial information of an image by splitting it into nine equal sub-images, and representing each sub-area by a color histogram. However, although conceptually simple, this approach may still not provide accurate color-spatial information, and is expensive both to compute and to store.

To facilitate the search of large-scale image collections Smith and Chang have used color sets to approximate histograms. Color sets correspond to salient image regions, and are represented by binary vectors to allow a more rapid search [69].

Stricker and Dimai [74] have split the image into a oval central region and four corners. Their system evaluates and combines the color feature similarity of each of these sub-images, attributing more weight to the central region. This is, however, a strictly domain-dependent solution: while it could be effective for an archive of photographs, it might not work well in other applications. Some splitting strategies are shown in Figure 6.

Figure 6. Examples of possible sub-image indexation.

Stricker [71] has used boundary histograms to encode the lengths of the boundaries between different discrete colors, in order to take into account geometric information in color image indexing. But this method may yield a huge feature space (for a discrete color space of 256 elements, a boundary histogram of 32,768 bins), and is not robust enough to deal with textured color images. Gagliardi et. al. [29] have investigated the use and integration of different color information descriptions and similarity measurements to enhance the system's effectiveness. In their method both query and database images are described in the CIELAB color space with two limited palettes of perceptual significance, of 256 and 13 colors respectively. A histogram of the finer color quantization and another of the boundary lengths between two discrete colors of the coarser quantization are used as indices of the image. While the former contains absolutely no spatial information, but describes only the color content of the image, the latter provides a concise description of the spatial arrangement of the basic colors in the image. The similarity between two boundary histograms BQ and BD is computed as follows:

$$S_b(BQ, BD) = 1 - \frac{\sum_{i \in \{\text{all bins}\}} |BQ_i - BD_i|}{s(BQ) + s(BD)}$$

where s( ) is the size of the histogram that is the sum of the edge lengths in the image. Normalization is necessary as the size of the boundary histogram is not constant, but depends upon the image segmentation. Since the images are coarsely segmented, the boundary histogram is little influenced by minor image details and noise.

Suitable procedures for measuring the similarity between histograms are then adopted, and the measures combined to model the perceptual similarity between the query and target images.

Pass, Zabih and Miller [59] also present a histogram-based method for comparing images that incorporates spatial information. They classify each pixel of the quantized image as either coherent or incoherent, depending upon whether or not it is a part of a “large” similarly-colored region (a region is classified as large if its size exceeds a fixed user-set value). By counting coherent and incoherent pixels separately the method offers a finer distinction between images than color histograms alone can provide (Figure 12)

For each color c_i the number of coherent pixels, α_{c_i}, and the number of non-coherent pixels, β_{c_i} are computed; each entry in the CCV is thus a pair $(\alpha_{c_i}, \beta_{c_i})$, called a *coherence pair.* The whole coherence vector is defined as:

$$CCV(\tilde{I}) = \langle (\acute{a}_{c_1}, \hat{a}_{c_1}), \ldots, (\acute{a}_{c_i}, \hat{a}_{c_i}), \ldots, (\acute{a}_{c_n}, \hat{a}_{c_n}) \rangle$$

The sum $\alpha_{c_i} + \beta_{c_i}$ is clearly the number of pixels of color c_i present in the image; the set of the sums for i=1..n represents the color histogram. The L1 distance can be used to compare two CCV:

$$\ddot{A}CCV(\tilde{I}, \tilde{I}') = \sum_{i=1}^{n} \left(\left| \acute{a}_{c_i} - \acute{a}'_{c_i} \right| + \left| \hat{a}_{c_i} - \hat{a}'_{c_i} \right| \right)$$

Spatial Chromatic Histograms (SCH) [12] combine information about the location of pixels of similar color and their arrangement within the image with that provided by the classical color histogram. For every color in the quantized image, the percentage of pixels having the same color is calculated, and the spatial information summarized in the relative coordinate of the baricenter of their spatial distribution (**b**) and the corresponding standard deviation from the bariceter (σ). Combining histogram and spatial information requires a new distance function. Given two Spatial Chromatic Histograms H and H' having c bins, the distance is computed as follows:

$$D(H,H') = \sum_{i=1}^{c} \min(h_H(i) - h_{H'}(i)) \left(\frac{\sqrt{2} - d(b_H(i), b_{H'}(i))}{\sqrt{2}} + \frac{\min(\acute{o}_H(i), \acute{o}_{H'}(i))}{\max(\acute{o}_H(i), \acute{o}_{H'}(i))} \right)$$

where h(i) is the ratio of pixels having color i.

Mitra et al [55] have proposed new color features for image indexing called color correlograms. Color correlograms include the spatial correlation of colors, and can be used to describe the global distribution of the local correlations. These features appear to tolerate even large changes in the appearance of the same scene caused by changes in viewing position and background, partial occlusions, and camera zooms. The color correlogram is a set of values $\gamma^{(k)}_{c_i,c_j}$ that gives the probability that a pixel of color c_i lies at distance k from a pixel of color c_j:

$$\gamma^{(k)}_{c_i,c_j} = \Pr_{p_1 \in I_{c_i}, p_2 \in I} [p_2 \in I_{c_j} \,||\, p_1 - p_2 |= k]$$

where p_1 and p_2 are two pixels, I is the image, and I_{c_i}, I_{c_j}, the set of pixels of colors c_i and c_j respectively. Computing the correlogram for a set of m distances k and given c colors of quantization, the whole feature size is c^2m, which is rather high if the quantization is not sparse. To overcome this problem, another feature, called the banded correlogram, is employed. This feature merges the results of the correlograms computed on different k:

$$\bar{\gamma}^{(k)}_{c_i,c_j} = \sum_{k'=kb}^{(k+1)b-1} \gamma^{(k')}_{c_i,c_j} \quad \text{for } 1 \le k \le b$$

The banded correlogram requires c^2m/b values. If only local information is needed a small set of distances suffices to capture the spatial correlation of pixels, further reducing the dimensions of the feature.

Brambilla et al. [11] have used multiresolution wavelet transform in a modified CIELUV color space to compute image signatures for use in content-based image retrieval applications, i.e. target search and similarity retrieval. The multiresolution wavelet decomposition applied is based on Haar wavelet filtering applied consecutively along horizontal and vertical directions (Figure 7). The major features of the images are coded in signatures of predefined lengths, which are compared in the retrieval phase by applying a similarity measure the system has pre-learned from a learning set of "very similar", "rather-similar", "not-very-similar", and "different" pairs of images, using a regression model for ordinal responses. This method is related to Jacob et al.'s work [35], in which a similar approach has been employed to target image search, i.e. to seek a specific image in the database when only a rough painted sketch or low-resolution version of it is available.

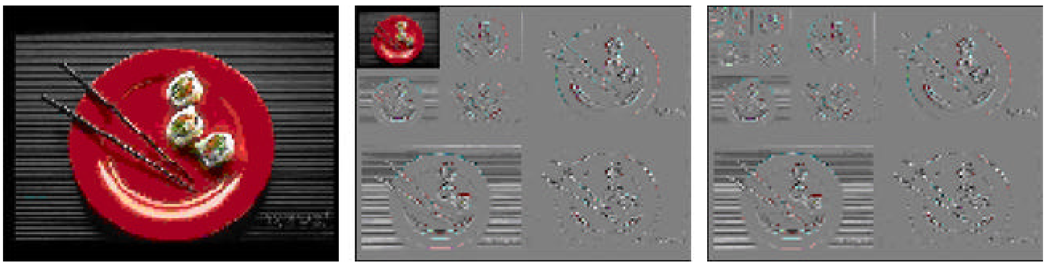

Figure 7. a) original image, b) two-step multiresolution wavelet transform, c) multiresolution wavelet transform.

Figure 8. Examples of 64 and 128 coefficient wavelet-based retrieval.

5 COMPARING SEGMENTED IMAGES

Segmented images represented in the spatial domain are usually described by Region Adjacency Graphs [58], or by multidimensional indices representing the spatial distribution of the color clusters [69].

Binaghi et al. [8] have described the segmented image in terms of color, spatial distribution, and coverage. During retrieval feature differences are mapped to a reduced set of qualitative linguistic modeling the uncertainty in the color description.

Smith and Chang [69] have developed a method for automatically extracting and indexing significant color regions from images. These methods utilize efficient indexing techniques for color information, region sizes, and absolute and relative spatial locations to allow a wide variety of complex joint color-spatial queries. However, they do not provide a general procedure for measuring the similarity between two images.

Kankanhalli et al (1999) [39] have combined color and spatial clustering for image retrieval. Color clustering is applied first, to obtain an initial segmentation; then a component labeling algorithm is applied to obtain spatial clustering. The spatial color cluster is identified by the mean color value, the fraction of the image the cluster constitutes, and the coordinates of the centroid. The similarity between two images is based on five heuristically weighted elements: the similarity between color clusters, the relative frequency of pixels in corresponding clusters, the spatial distance between color clusters, the relative frequency of pixels of the corresponding spatial color clusters, and the spatial distance between spatial color clusters.

The description of segmented images using 2D strings, or their variant is brittle in the sense that minor changes in region location may greatly affect the comparison of two images. In [17][7], the authors have introduced an original modeling technique for the quantitative representation and comparison of the mutual positioning of a pair of extended regions, which can account for the overall distribution of relationships among the individual pixels belonging to the two regions. In this approach, the relationship between two regions is represented by a finite set of equivalence classes ("walkthroughs") among the dense sets of possible paths leading from any pixel in first region to any pixel in the second. Each equivalence class is associated with a weight which provides an integral measure of the set of pixel pairs connected by a path belonging to the class, thus accounting for the degree of truth with which the individual class represents the actual relationship between the two regions.

6 ILLUMINANT INVARIANT COLOR IMAGE INDEXING

We may assume that all the images stored in the database are described in terms of sRGB color coordinates, but we can not always assume constant imaging conditions during data acquisition (particularly when the database contains images collected from many different sources, such as material taken from the WEB). In many cases the user may still be able to recognize the colors in the scene, but we can only guess to what extent an image search engine can perform the same task, and at what cost when the retrieval is based only on color.

Previous studies in the field of object recognition have drawn attention to how changes in illuminant conditions modify the colors of the image acquired, and affect the results of color-based algorithms. Swain and Ballard [75] have suggested the application of a color constancy algorithm to cope with this problem. Their work has been followed by several attempts to provide for the effects of illuminant changes through the definition and use in indexing of illuminant-independent color features.

6.1 Illuminant-invariance achieved with descriptors

A method for deriving illuminant-independent color descriptors has been developed by Funt and Finlayson [28]. Starting from the assumption of the local constancy of illumination, they have indexed the derivative of the logarithm of the image, which is, in effect, the ratio of neighboring colors. The histogram of the derivatives of the logarithm of an image codes the lengths of the boundaries between colors, providing an illumination-independent description of the image. These authors' experiments on small sets of synthetic and real images have demonstrated that their *color constant color index algorithm* performs slightly worse than Swain's under fixed illumination, but substantially better than Swain's under illumination that varies both spectrally and spatially.

In a subsequent publication, Finlayson et al. [22] have proposed the use of color angular indexing to describe image similarity. In the angular indexing paradigm, objects are represented by the three color angles. To these the authors have added three color texture angles, calculated after a linear filtering of the image. Using these six features they have obtained illuminant-invariance in indexing, while accelerating matching process by drastically reducing the parameters to account for, from the 4064 of Swain and Ballard’s histogram to 6. In their experiments they have found that this method performs better than color indexing [75] and color constant color indexing [28] in the presence of changes in illumination.

Gevers and Smoulders have proposed various color models and checked their invariance under changing imaging conditions, including the presence of shadows and highlights [30][31]. These tests were performed on a database of about 500 images. One of the color model proposed assumes dichromatic reflection and white illumination, and is independent of the viewpoint, surface orientation, illumination direction and intensity and highlights [31]. An instantiation of this model is:

$$l_1(R,G,B) = \frac{(R-G)^2}{(R-G)^2 + (R-B)^2 + (G-B)^2}$$

$$l_2(R,G,B) = \frac{(R-B)^2}{(R-G)^2 + (R-B)^2 + (G-B)^2}$$

$$l_3(R,G,B) = \frac{(G-B)^2}{(R-G)^2 + (R-B)^2 + (G-B)^2}$$

A further model which was demonstrated to be invariant for illuminant color as well is

$$m_1 = \frac{R^{x_1} G^{x_2}}{R^{x_2} G^{x_1}}, m_2 = \frac{R^{x_1} B^{x_2}}{R^{x_2} B^{x_1}}, m_3 = \frac{G^{x_1} B^{x_2}}{G^{x_2} B^{x_1}}$$

where R, G, B are the color values in the RGB color space and x_1 and x_2 are the locations in the image of two neighboring pixels.

Berens and Finlayson [5] have proposed a method that addresses simultaneously the problem of illuminant-invariance in indexing and the dimension of color histograms. Each (R,G,B) coordinate is transformed into a log-chromaticity space defined by:

$$(\ln(\frac{R}{G}), \ln(\frac{RG}{B^2})) = (\ln R - \ln G, \ln R + \ln G - 2\ln B)$$

In this color space, illuminant changes result in a translation of coordinates; consequently illuminant-invariance can be accomplished by mean subtraction. To reduce the dimension of the matching problem, the authors have coded the log-chromaticity distribution obtained by projecting it onto a PCA base. Using Funt et al.'s dataset of 11 objects photographed under 4 different illuminants they have demonstrated how removing the mean from the log-chromaticity coordinates removes the effect of illuminant variance on the color distribution.

In [46] Mandal et al. have presented a method to define illuminant invariant moments. Assuming that the change in illumination is uniform, and the illumination does not produce a nonlinear effect on the image, the histograms of an image with varying lighting conditions can be approximated as a translated and scaled version of each other. This translation and scaling effect can be countered by using as image indexing, a set of moments of a translation and scale invariant (TSI) histogram. If we assume that the change in illumination has dilated the *pdf* of an image by a factor a, it can be demonstrated that the k-th central moments and normalized central moments of the *pdf* of the changed image and those of the original image are related as follow:

$$ì'_k = á^k ì_k \quad \text{and} \quad â'_k = óâ_k$$

From the normalized central moments, a set of translation and scale invariant (TSI) moments can be defined as:

$$\varsigma_k = \frac{\hat{a}_k}{\hat{a}_2} \quad k > 2, k \in Z$$

Since b_1 is equal to zero, the h_k's can be considered moments of a *pdf* with the first moment set at zero, and the second set at one, whatever the value of a that transformed the image's illumination. In [47] Mandal et al have defined a procedure to convert TSI moments to the corresponding Legendre moments as it has been shown that these perform better than central moments.

Mandal et al [47] have proposed a method of image indexing using illumination-compensated standard deviations and shape parameters of highpass wavelet subimages. They have demonstrated that the standard deviations of the coefficients of the wavelet subbands are related to each other by a linear function of the illuminant scale factor a:

$$s'_{F_k} = a s_{F_k}$$

The histograms of highpass wavelet coefficients can be described using a generalized Gaussian density function which depends on a parameter called *shape* (γ) [9]. For the shape parameter the following relationship holds:

$$g'_{F_k} = g_{F_k} + c(1-a) \text{ with } c \text{ a positive constant}$$

The standard deviation and the shape parameter are used as indexing values: first, the scale factor a is estimated by comparing the b_2 moments of the histograms corresponding to the query and target image (see the translation and scale invariant moments), then the new values are calculated from the above equations, and used to evaluate the distance between the two images.

6.2 Illuminant-invariance achieved by image pre-processing

All the previously mentioned studies address the definition of illuminant-invariant color features. Funt and Barnard [26] have, instead, compared five preprocessing algorithms for normalizing images to a standard neutral illuminant and removing the effects of varying illumination conditions in the acquired scenes. The algorithms tested were: White Patch Retinex, Greyworld, 2D Gamut-constraint and 3D Gamut-constraint and Neural Networks. They were run on 110 images of 11 objects viewed under 5 different illuminants and in 2 different positions. Apart from the obvious conclusion that color constancy improved color indexing when the target image was produced under an illuminant different from that used to acquire the corresponding image in the database, the results achieved by color constancy algorithms were disappointing. The best performer was successful only 67% of the time, compared with the 92% when the illuminant was known.

Finlayson et al. [23] have presented a comprehensive image normalization which removes image dependency on lighting geometry and illumination color. Their approach alternates normalization for illumination intensity and a normalization for illumination color, iterating this process until a stable state is reached. If an RGB image of N pixels is represented with a Nx3 matrix, the normalization operators are defined as:

$$R(I)_{i,j} = \frac{I_{i,j}}{\sum_{k=1}^{3} I_{i,k}}$$ normalization for illumination intensity

$$C(I)_{i,j} = \frac{\frac{N}{3} I_{i,j}}{\sum_{k=1}^{N} I_{k,j}}$$ normalization for illumination color

where i,j indicates the ijth element of the matrix. The algorithm can be outlined as follows:

1. $I_0=I$ Initialization
2. $I_{i+1}=C(R(I_i))$ Iteration
3. $I_{i+1}=I_i$ Termination condition.

The experiments were run out on a database of about 100 images.

Retinex was originally proposed by Land and McCann [41] in order to understand and emulate human color perception. Many variations of the method have been developed in the last thirty years [48][27]. Ciocca et al. [13] have recently examined the performance of various color-based retrieval strategies when coupled with a pre-filtering based on Brownian Look Up Table Retinex algorithm [48] to see whether, and to what degree, Retinex improved the effectiveness of the retrieval, regardless of the strategy adopted.
The retrieval strategies implemented have included color and spatial-chromatic histogram matching, color coherence vector matching, and the weighted sum of the absolute differences between the first three moments of each color channel (see Section 7 for a description of the retrieval algorithms). The experiments were performed on two databases, containing 310 paintings and 387 ceramic objects respectively: 15 images were randomly selected from each database, simulating for each of these a change in imaging conditions, using one of eight illuminants. These images were then used to query the corresponding database, employing one strategy at a time, first without any pre-filtering, and then applying the Retinex algorithm to both the query and the database images. The conclusion was that Retinex pre-filtering improved the retrieval effectiveness of all the retrieval strategies implemented. A simple comparison of the improvements in performance suggests that better results are obtained when Retinex is coupled with color quantization.

Figure 9 shows images resulting from the application of Marini et al.'s version of Retinex [48] and of that defined by Funt et al. [27]. The algorithms have been applied to images of the same scene under different illuminants.

Figure 9. a) Original image. b) Simulation of a warm illuminant. a1) b1) Retinex proposed by Marini et al. on image a) and b). a2) b2) Retinex proposed by Funt et al. on image a) and b).

7 INTEGRATION WITH OTHER FEATURES

Most content-based image retrieval systems do not exploit only color and spatial-color features for characterizing image content. All the low-level features that can be computed automatically, i.e. without human assistance, could be computed and associated with the color-based features described in the previous Sections to index an image database. In many systems color-based features have already been coupled with texture and edge/shape features.

Texture has been widely studied in psychophysics, as well as in image analysis and computer vision. However, our understanding of it is still very limited, compared with our knowledge of other visual features, such as color and shape. Most of the computational methods available for describing texture provide for the supervised or unsupervised classification of image regions and pixels. Within these contexts gray level textures have been processed using various approaches, such as Fourier transform, co-occurrence statistics, directional filter masks, fractal dimension and Markov random fields (for a review of the various methods, see [20][77]).

Rao and Lohse have designed an experiment to identify the high level features of texture perception [62][63]. Their results suggest that in human vision three perceptual features (“repetitiveness”, “directionality”, and “granularity and complexity”) concur to describe texture appearance. Consequently, the computational model applied in image indexing should compute features that reflect these perceptual ones. To do so, the IBM QBIC system uses a modified version of the features of “coarseness”, “contrast”, and “directionality” proposed by Tamura for image indexing [76][21]. Amadusun and King have proposed another feature set that corresponds to the visual properties of texture: “coarseness”, “contrast”, “busyness”, “complexity”, and “texture strength” [1]. Picard and Liu, extending the work described in [24][25], have proposed an indexing scheme based on Word Decomposition of the luminance field [44][61] in terms of "periodicity", "directionality", and "randomness”. Wavelet transform representations, also described in Section 4, have been proved effective for texture annotation [45].

Shape features may provide the highest level of abstraction in describing image content, but they can not always used as they generally require that regions of interest in the images be segmented a-priori, and, as we have said in Section 2, "good" unsupervised segmentations are not always possible. Various approaches have been proposed in the literature for shape-based image retrieval. According to Del Bimbo [19] these can be distinguished by whether they employ feature based representations (both boundary based and region based) or follows shape transformation techniques. Depending on the applications some shape representations must be invariant to translation, rotation, and scaling, while others need not. Jain and Vailayata [36] have defined a very simple and effective shape matching technique based on histograms of the direction of significant edges. This features is invariant to translation and can be made invariant for scaling and rotation as well.

Mehtre et al. [52] and Scasselati et al. [68] have reviewed and reported experimental comparisons of different methods of image retrieval based on shape similarity.

Although several general-purpose systems have been developed in the last few years, the integrated management of the various image features remains complex and application dependent [57][60]. Several factors may intervene when choosing the aggregation operator to integrate the results of a query based on many features [8]: different tasks in the same context deal with similarity at different levels of precision; similarity depends greatly on the nature of the objects to which it is applied, and on the features selected for their description; different users from different backgrounds may interpret image content differently, and the objectives of their queries may also differ. All these factors, which are interrelated and consequently influence each other, make it quite impossible to determine in advance the most suitable aggregation operator for the different similarity measures, e.g. [51]. This leaves to the users the burden of formulating their information needs, which may be rather difficult (and tiresome) to express as a weighted combination of the features that are actually employed for retrieval [65].

It is obvious that user feedback is a key element in the successful retrieval of multimedia information. The potentials of relevance feedback in textual information retrieval have been widely studied. In image retrieval, it has been employed by Minka and Picard [53] and by Cox et al. [16] for target search, and by Rui et al. [66], La Cascia et al. [40] for similarity retrieval. Ciocca and Schettini [14] have designed an algorithm that, through the statistical analysis of the image feature distributions of the retrieved images the user has judged relevant or not relevant, identifies the features the user has taken into account in formulating this judgement. It then modifies accordingly the contribution of the different features to the overall evaluation of image similarity (see figure 13 and 14).

8 PERFORMANCE EVALUATION OF IMAGE RETRIEVAL STRATEGIES

Although there are a host of measures that can be used for evaluating retrieval strategies, in the great majority of papers, when there was more than one relevant image in the database with respect to a generic query, the effectiveness of the retrieval methods has been quantified in terms of recall vs. precision graphs [42]. Recall, which is defined as the ratio between the number of relevant images retrieved and the number of all relevant images in the database, quantifies the ability of the system to retrieve useful images. Precision, which is defined as the ratio between the number of relevant images retrieved and the number of retrieved images, measures the ability to reject useless ones [56]. A measure called Effectiveness (Efficiency of Retrieval, or Fill Ratio), has been proposed by Mehtre et al. [50]and often applied [51][52][29] for evaluating image retrieval methods. Let S be the number of images retrieved in the short list when posing a query; $\mathbf{R}_{\mathbf{q}}^{\mathbf{I}}$, the set of relevant images in the database; and $\mathbf{R}_{\mathbf{q}}^{\mathbf{E}}$, the set of images retrieved in the short list (considered "relevant" by the system). The effectiveness measure is defined as:

$$
\varsigma_{S} = \begin{cases} \dfrac{\left|\mathbf{R}_{\mathbf{q}}^{\mathbf{I}} \cap \mathbf{R}_{\mathbf{q}}^{\mathbf{E}}\right|}{\left|\mathbf{R}_{\mathbf{q}}^{\mathbf{I}}\right|} & \text{if } \left|\mathbf{R}_{\mathbf{q}}^{\mathbf{I}}\right| \le S \\[2ex] \dfrac{\left|\mathbf{R}_{\mathbf{q}}^{\mathbf{I}} \cap \mathbf{R}_{\mathbf{q}}^{\mathbf{E}}\right|}{\left|\mathbf{R}_{\mathbf{q}}^{\mathbf{E}}\right|} & \text{if } \left|\mathbf{R}_{\mathbf{q}}^{\mathbf{I}}\right| > S \end{cases}
$$

If $\left|\mathbf{R}_{\mathbf{q}}^{\mathbf{I}}\right| \le S$, the effectiveness is reduced to the traditional recall measure, while if $\left|\mathbf{R}_{\mathbf{q}}^{\mathbf{I}}\right| > S$, the effectiveness corresponds to precision. This measure, however, does not take into account the number of relevant images in the database. To overcome this drawback, the authors evaluate the Effectiveness for different lengths of the short list S, producing an "Effectiveness vector", the significance of which is not immediately clear.

In some application domains some images may be "more relevant than others", therefore, the ranking of relevant images must be taken into account in the design of the performance metrics [29]. As even all expert may find it difficult and tiresome to exactly rank the images on the basis of similarity, a few equivalent classes of similarity can be provided (e.g. *very similar, rather similar, not-very similar* and *different*) [29].

To quantify the performance of a retrieval strategy with a global score when there is only one relevant image (target) in the database with respect to the query, the Success of Target Search index (STS) can be used. This score is defined as:

$$
STS = \left(1 - \frac{Rank - 1}{N - 1}\right)
$$

where *Rank* is the retrieval position of the target image and ranges from 1 to *N*, *N* being the number of images in the database.

Since image retrieval is often an interactive process, other performance measures can be applied [42][56]: these may include the average number of processing stages required to achieve satisfactory results, the computational complexity, and other, domain-specific performance indices [19].

9 CONCLUSIONS

Color has been widely used for content-based image and video retrieval. Since the introduction of color distributions as descriptors of image content, various research projects have addressed the problems of color spaces, illumination invariance, color quantization, and color similarity functions. Many different methods have been developed to enhance the limited descriptive capacity of color distributions. We have presented here the state of the art of color-based methods that can be used to index and retrieve color images.

The most surprising element that emerges from our study of color indexing and retrieval is that most of the methods analyzed do not explore the problem of how to deal with color in a device-independent way. Very seldom are details given, or references made to image acquisition and management in terms of standard color coordinates, although it is reasonable to assume that the image database contains images acquired from many sources, and subjected to a number of processing steps before indexing and display. Quantization and segmentation reduce acquisition noise, and may to some extent cope with changes in imaging conditions, but a more rigorous approach to color imaging is surely desirable.

Several of the algorithms proposed have been designed to implement machine color constancy, but their application in real world conditions is still under investigation. The very definition of machine color constancy is still a matter for future research, rather than an effective tool that can be employed in current content-based image retrieval engines.

Most color based retrieval algorithms make it possible to perform searches for the presence of specific colors, but not, with very few exceptions [2], for their absence. Consequently, it would be desirable to modify retrieval methods so that users can specify which colors should be excluded from the image retrieval query.

The definition of feature similarity also plays a fundamental role in content-based retrieval, as images with "similar" feature distributions are often considered similar in appearance without requiring any semantic expression of this.

Most of the methods described here have been tested on different databases, of very different sizes, ranging from 50 to 200,000 images, for different retrieval tasks. This makes it extremely difficult, if not impossible, to provide an absolute ranking of the effectiveness of the algorithms. We can make the general observation that color alone can not suffice to index large, heterogeneous image databases. The combination of color with other visual features is a necessary approach that merits further study. Much more dubious, instead, are methods that assume that affordable, unsupervised image segmentation is always possible, and that the evaluation of image similarity can be dealt with as a graph matching problem, with a reasonable computational burden. The design of a content-based image retrieval system must address issues of efficiency in addition of those of effectiveness.

A promising direction for future research is, in our opinion, the exploitation of color image similarity for image database navigation and visualization ([64] is an attempt in this direction) and the retrieval of color images based on psychological effects [19]. We would also like to see new generation systems that support querying by similar emotions (e. g. joy), and open-ended searches in the image database where similar images are located next to each other.

10 REFERENCES

[1] Amadasun M., King R., Textural features corresponding to textural properties, IEEE Transaction on System, Man and Cybernetics, Vol. SMC-19(5), pp. 1264-1274, 1989.

[2] Androutsos “A novel vector-based approach to color image retrieval using vector angular-based distance measure”, *Computer Vision and Image Understanding*, Vol. 1/2, pp. 46-58, 1999.

[3] Barolo B., I. Gagliardi, R. Schettini, An effective strategy for querying image databases by color distribution, *Computer and the History of Art Journal*, Vol .7(1), pp. 3-14, 1997.

[4] Berens J., G.D. Finlayson, G. Qiu, Image indexing using compressed histograms, Proc. The Challenge of Image Retrieval, 1999.

[5] Berens J. and G. Finlayson, “Log-opponent chromaticity coding of colour space”, *International Conference on Pattern Recognition*, Barcelona, 2000.

[6] Berman A.P., L.G. Shapiro, A flexible image database system for content-based retrieval. *Computer Vision and Image Understanding,* Vol. 75, pp. 175-195, 1999.

[7] Berretti S., A.Del Bimbo, E.Vicario, "Modeling spatial Relationships between Color Sets," *IEEE workshop on Content Based Access to Image and Video Libraries CBAIVL'00,* June 2000.

[8] Binaghi E., I. Gagliardi, R. Schettini, Image retrieval using fuzzy evaluation of color similarity. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 8, pp. 945-968, 1994.

[9] Birney K. A. and T. R. Fischer, “On the modelling of DCT and subband image data for compression”, IEEE Trans. Image Process. Vol. 4(2), 1995.

[10] Borsotti M., P. Campadelli, R. Schettini, Quantitative evaluation of color image segmentation results, *Pattern Recognition Letters*, Vol. 19, pp. 741-747, 1998

[11] Brambilla C., A. Della Ventura, I. Gagliardi, R. Schettini, Multiresolution wavelet transform and supervised learning for content based image retrieval, *IEEE Multimedia Systems 99,* IEEE Computer Society, Vol. I, pp.183-188, 1999.

[12] Cinque L., S. Levialdi, A. Pellicano', Color-Based Image Retrieval Using Spatial-Chromatic Histograms, *IEEE Multimedia Systems 99,* Vol. II, pp. 969-973, 1999.

[13] Ciocca G., D. Marini, A. Rizzi, R. Schettini, S. Zuffi, “On pre-filtering with Retinex in color image retrieval” *Proc. Second Internet Imaging Conference* (G. Beretta, R. Schettini eds.)**,** S Jose, January 2001.

[14] Ciocca G., R. Schettini, A relevance feedback mechanism for content-based image retrieval, *Information Processing and Management*, Vol. 35, pp. 605-632, 1999.

[15] Colombo C., A. Del Bimbo, Color-induced representation and retrieval, *Pattern Recognition,* Vol. 32, pp. 1685-1695, 1999.

[16] Cox I.J., M.L. Miller, S.O. Omohundro, P.N. Yianilos, PicHunter: Bayesian Relevance Feedback for Image Retrieval. Proc. ICPR'96, pp. 361-369, (1996)

[17] Del Bimbo A., E.Vicario, Using Weighted Spatial Relationships in Retrieval by Visual Contents, *Workshop on Image and Video Content-Based Retrieval*, Italian Chapter of the IAPR, Milano, February 1998.

[18] Del Bimbo A., M. Mugnai, P. Pala, F. Turco, Visual querying by color perceptive regions, *Pattern Recognition*, Vol. 31, pp. 1241-1253, 1998.

[19] Del Bimbo A., Visual Information Retrieval, Morgan Kaufmann Publishers, San Francisco, California, 1999

[20] Du Buf J.M.H., Kardan M., Spann M. *Texture feature performance for image segmentation* Pattern Recognition, Vol. 23, pp. 291-309, 1990.

[21] Equitz W., Niblack W. Retrieving images from a database: using texture algorithms from the QBIC system, IBM Research Division, Research Report 9805, 1994.

[22] Finlayson G. D., S. S. Chatterjee and B. V. Funt, “Color Angular Indexing and Image Retrieval”, Image Description and Retrieval ed. Vicario, Plenum Press, 1998.

[23] Finlayson G.D., B. Schiele, J.L. Crowley, Comprehensive Colour Image Normalization, *Proc. 5th European Conference on Computer Vision*, 1998.

[24] Francos J.M., Meiri A.Z., Porat B. A unified texture model based on a 2-D Word like decomposition, IEEE Trans. on Signal Processing, pp. 2665-2678, 1993.

[25] Francos J.M., Meiri A.Z., Porat B. Modeling of the texture structural components using 2-d deterministic random field Visual Communication and Image Processing, Vol. SPIE 1666, pp. 554-565, 1991.

[26] Funt B., K. Barnard, L. Martin, Is machine colour constancy good enough ?, *Proc. 5th European Conference on Computer Vision*, 1998.

[27] Funt B., F. Ciurea, J. McCann, “Retinex in Matlab”, *Proceedings of IS&T/SID Eighth Color Imaging Conference*, Scottsdale, Arizona (USA), pp.112-121, November 7-10, 2000.

[28] Funt B.V., G.D. Finlayson, “Color Constant Color Indexing”, *IEEE Transactions on PAMI*, Vol. 17, pp. 522-529, 1995.

[29] Gagliardi I., R. Schettini, A method for the automatic indexing of color images for effective image retrieval. *The New Review of Hypermedia and Multimedia*, Vol. 3, pp. 201-224, 1997.

[30] Gevers T., A.W.M. Smeulders, “A Comparative Study of Several Color Models for Color Image Invariant Retrieval”, *Proc. IDB-MMS’96*, 1996.

[31] Gevers T., A.W.M. Smeulders, “Color Based Object Recognition”, *Pattern Recognition*, Vol. 32, pp.453-464, 1999.

[32] Gong Y., C.H. Chuan, G. Xiaoyi, Image indexing and retrieval using color histograms, *Multimedia Tools and Applications*, Vol. 2, pp. 133-156, 1996.

[33] Hafner J., Sawhney H.S., Esquitz W., Flickner M., Niblack W. Efficient color histogram indexing for quadratic form distance functions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.17, pp. 729-736, 1995.

[34] Haralick R.H., L.G. Shapiro, Image Segmentation Techniques, *Computer Vision, Graphics, and Image Processing*, Vol. 29, pp. 100-132, 1985.

[35] Jacob C., Frinkelstein A., Salesin D. “Fast multiresolution image query” Technical Report 95-01-06, University of Washington, 1995.

[36] Jain A.K., A. Vailaya, Shape-based retrieval: a case study with trademarks databases, Pattern Recognition, Vol. 31, pp. 1369-1390, 1998.

[37] Jaynes E.T., On the rationale of maximum-entropy methods, *Proc. of IEEE*, Vol. 70, pp. 939-952, 1982.

[38] Kankanhalli M.S., B.M. Methre, J.K. Wu, Cluster-based color matching for image retrieval, *Pattern Recognition*, Vol. 29, pp. 701-708, 1996.

[39] Kankanhalli M. S., B.M. Mehtre, H.Y. Huang, Color and spatial feature for content-based image retrieval, *Pattern Recognition Letters*, Vol. 20, pp. 109-118, 1999.

[40] La Cascia M., S. Sethi, and S. Sclaroff Combining Textual and Visual Cues for Content-based Image Retrieval on the World Wide Web Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries, June, 1998

[41] Land E., J. McCann , "Lightness and Retinex Theory", Journal of Optical Society of America, Vol. 61, pp. 1-11, 1971.

[42] Leung C.H.C., H:H:S. Ip, Benchmarking for content-based visual information search, Advance in Visual Information Systems (R. Laurini) pp. 442-456,

[43] Liu J., Y-H Yang Multiresolution color image segmentation *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(7), pp. 689-700, 1994

[44] Liu F., Picard R.W. *Periodicity, directionality and randomness: Wold features for perceptual pattern recognition* MIT Vision and Modeling Lab., Tech. Report #320, 1994.

[45] Ma W.Y, B.S. Manjunath A comparison of wavelet transform features for texture image annotation, *Proc. IEEE International Conference on Image Processing*, 1995.

[46] Mandal M. K., T. Aboulnasr, and S. Panchanathan, "Image indexing using moments and wavelets", IEEE Trans. Consumer Electron. Vol. 42(3), pp. 557-565, 1996.

[47] Mandal M. K., T. Aboulnasr, S. Panchanathan, "Illuminant invariant image indexing using moments and wavelets", Journal of Electronic Imaging Vol. 7(2), pp. 282-293, 1998

[48] Marini D., A. Rizzi, L. De Carli, "Multiresolution retinex : comparison of algorithms." First International Conference on Color in Graphics and Image Processing CGIP'2000, Saint-Etienne, France, October 1-4, 2000.

[49] McGill M.J., Salton G. *Introduction to modern Information Retrieval*, McGraw-Hill, 1983.

[50] Mehtre B.M., M.S. Kankanhalli, A. Desai Narasimhalu, G.C. Man, Color matching for image retrieval, *Pattern Recognition Letters*, Vol. 16, pp. 325-331, 1995.

[51] Mehtre B.M., M.S. Kankanhalli, W.F. Lee, Content-based image retrieval using a composite color-shape approach Information Processing and Management, Vol. 34 (1), pp. 109-120, 1998.

[52] Mehtre B.M., M.S. Kankanhalli, W.F. Lee, Shape measures for content based image retrieval: a comparison. Information Processing and Management, Vol. 33, pp. 319-337, (1997)

[53] Minka T., Picard R.W., Interactive learning with a "Society of Models". Pattern Recognition, Vol. 30(4), pp. 565-581, 1997.

[54] Miyahra M., Y. Yoshida, Mathematical transform of (RGB) color data to Munsell (HVC) color data. *Proc. SPIE Visual Communications and Image Processing,* Vol. 1001, pp. 650-657, 1988.

[55] Mitra M., J. Huang, S.R. Kumar, Combining Supervised Learning with Color Correlograms for Content-Based Image Retrieval, *Proc. of the Fifth ACM Multimedia Conference*, 1997.

[56] Narasimhalu Desai A., M.S. Kankanhalli, Benchmarking multimedia databases. Multimedia Tools and Applications, Vol. 4, pp. 333-356, 1997.

[57] Nastar C:, Mitschke M., Meilhac C., Boujemaa N. “Surfimage: a Flexible Content-Based Image Retrieval System” Proceedings ACM-Multimedia, Bristol, England, September 12-16, 1998.

[58] Park I.K., I.D. Yun, S.U. Lee, Color image retrieval using hybrid graph representation, *Image and Vision Computing*, Vol. 17, pp. 465-474, 1999.

[59] Pass G., R. Zabih, J Miller, Comparing Images Using Color Coherence Vectors. *Proc. Fourth ACM Multimedia Conference*, 1996.

[60] Pentland A., Picard R., and Sclaroff S., “Photobook: Tools for Content-Based Manipulation of Image Databases ”, International Journal of Computer Vision, Vol. 18(3), pp. 233-254, 1996.

[61] Picard R.W., Minka T.P. *Vision texture for annotation* Multimedia Systems, No. 3, pp. 3-14, 1995.

[62] Rao A.R., Lohse G.L. Identifying High Level Features of Texture Perception CVGIP: Graphical Models and Image Processing, Vol. 55(3), pp. 218-233, 1993.

[63] Rao A.R., Lohse G.L. Towards a texture naming system: identifying relevant dimensions of texture IBM Research Division, Research Report 19140, 1993.

[64] Rubner Y., C. Tomasi, L.J. Guibas., *A metric for distributions with applications to image databases*, *Proc. IEEE ICPR* , 1998.

[65] Rui Y., T.S. Huang, Image retrieval: current technologies, promising directions, and open issues, Journal of Visual Communication and Image Representation, Vol. 10, pp. 39-62 (1999).

[66] Rui Y., T.S. Huang, M. Ortega, S. Mehrotra, Relevance feedback: a power tool in interactive content-based retrieval, IEEE Transaction on Circuits and Systems for Video Technologies, Special Issue on Interactive Multimedia Systems for the Internet, Vol. 8(5), pp. 644-655, 1998.

[67] Sclaroff S., L. Taycher, M. La Cascia, Image-Rover: A Content-Based Image Browser for the World Wide, *Proc. IEEE Workshop on Content-based Access Image and Video Libraries*, 1997.

[68] Scasselati B., S. Alexopoulos, M. Flickner, Retrieving images 2D shape: a comparison of computational methods with human perceptual judgments. SPIE Conf. on Storage and Retrieval for Images and Videos (1994).

[69] Smith J.R., S-F Chang, Single color extraction and image query, *Proc. ICIP* ,1995.

[70] Syeda-Mahmood T.F., *Data and model-driven selection using color regions*, MIT, 1992 (AI memo n. 1270).

[71] Stricker M. A., Color and Geometry as Cues for Indexing, *Tecnical Report CS92-22*, 1992.

[72] Stricker M. and M. Orengo, Similarity of Color Images, *Proc. Storage and Retrieval for Image and Video Databases* III, pp. 381-392, 1995.

[73] Stricker M. and A. Dimai, Color Indexing with Weak Spatial, Constraints, *Proc. Storage and Retrieval for Image and Video Databases* IV, 1996.

[74] Stricker M. and A. Dimai. Spectral covariance and fuzzy regions for image indexing. *Machine Vision and Applications*, Vol. 10, pp. 66-73, 1997.

[75] Swain M.J. and D.H.Ballard, “Color Indexing”, *International Journal of Computer Vision*, Vol. 7, pp.11-32, 1991.

[76] Tamura H., Mori S., Yamawaky T. Textural Features Corresponding to Visual Perception IEEE Transaction on Systems, Man and Cybernetics, Vol. SMC-8(6), pp. 460-473, 1972.

[77] Tuceryan M., Jain A.K. *Texture analysis Handbook of Pattern Recognition and Computer Vision* (Eds. C.H. Chen, L.F. Pau, P.S.P. Wang), pp. 236-276, 1994.

[78] Uchiyama T., M.A. Arbib, Color Image Segmentation Using Competitive Learning, *IEEE Transaction on PAMI*, Vol. 16, 1197-1206, 1994.

[79] Vailaya A., A. Jain, H.J. Zhang, On image classification: Citie images vs. landscapes, *Pattern Recognition*, Vol. 31, pp. 1921-1935, 1998.

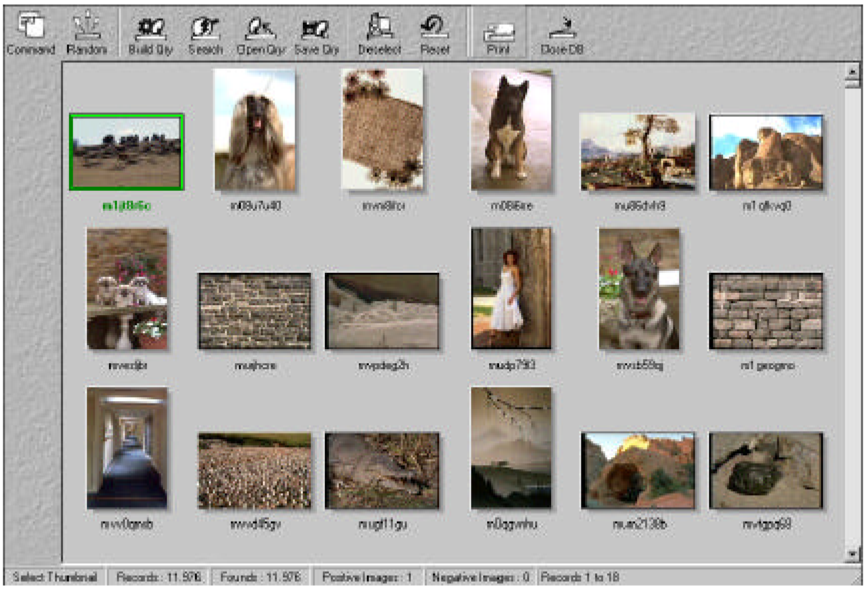

Figure10: Retrieval results using the first three moments of each color channel on a photo database of about 12,000 images

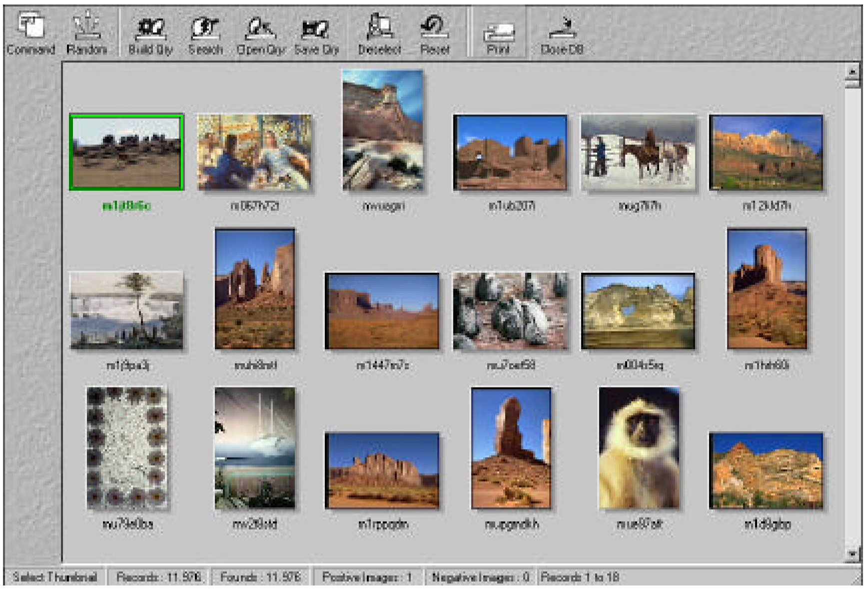

Figure11: Retrieval results using histograms intersection (64 colors) on a photo database of about 12,000 images

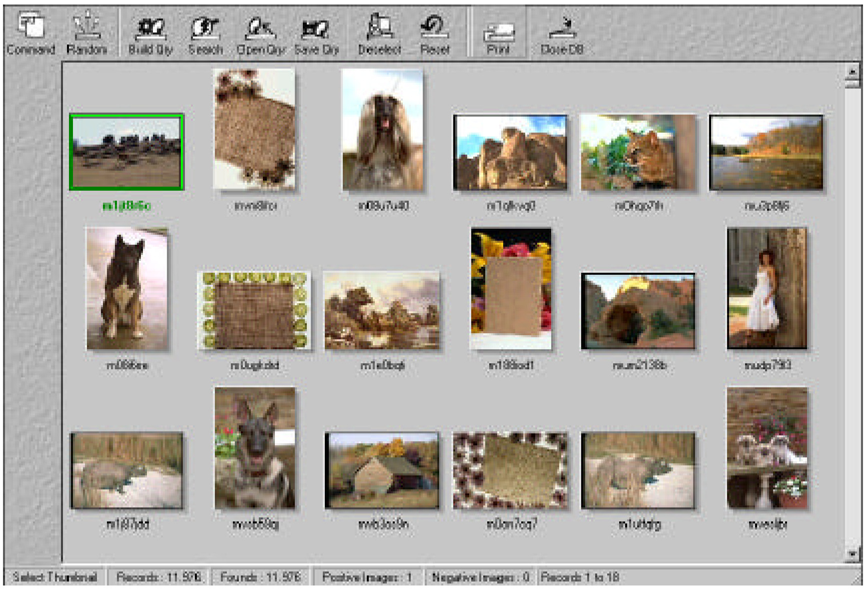

Figure12: Retrieval results using color coherence vectors (64 colors) on a photo database of about 12,000 images

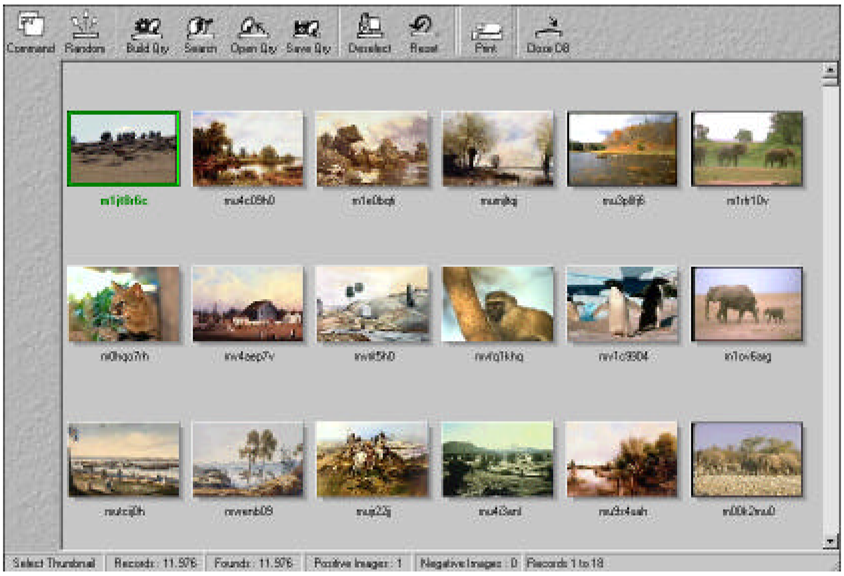

Figure13: Retrieval results using multiple features search on a photo database of about 12,000 images

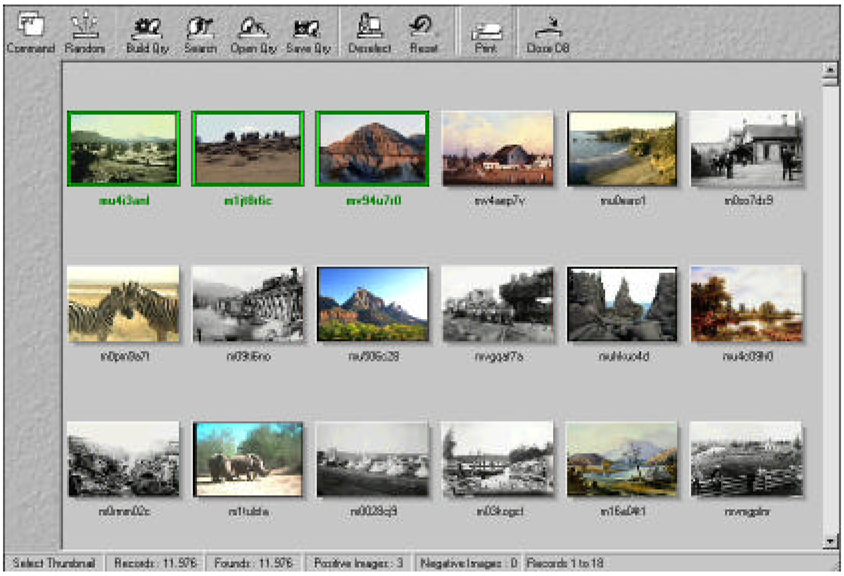

Figure 14: Retrieval results using multiple features search with relevance feedback on a photo database of about 12,000 images