Color-Shape Histograms for Image Representation and Retrieval

Renato O. Stehling *Mario A. Nascimento*
Alexandre X. Falcão

Relatório Técnico IC–00-11

Julho de 2000

Color-Shape Histograms for Image Representation and Retrieval

Renato O. Stehling[1]
renato.stehling@dcc.unicamp.br

Mario A. Nascimento[2]
mn@cs.ualberta.ca

Alexandre X. Falcão[1]
afalcao@dcc.unicamp.br

[1]Institute of Computing, State Univ. of Campinas, Brazil
[2]Dept. of Computing Science, Univ. of Alberta, Canada

Abstract

Color is a commonly used feature for realizing content-based image retrieval (CBIR). In this context, this paper presents a new approach for CBIR which is based on well known and widely used color histograms. Contrasting to previous approaches, such as using a single color histogram for the whole image, or local color histograms for a fixed number of image cells, the one we propose (named Color-Shape) uses a variable number of histograms, depending only on the actual number of colors present in the image, which our experiments have shown often to be low. Our experiments using a large set of heterogeneous images and pre-defined query/answer sets show that the Color-Shape approach offers good retrieval quality with relatively low space overhead, outperforming previous approaches. Furthermore, we also show that the proposed approach is very flexible in the sense that the user may easily tune it, in order to calibrate the trade-off between space overhead and retrieval effectiveness. For instance, when compared to using global color histograms, our approach can retrieve images 80% more effectively, requiring 29 times more space for metadata. Although large, this space overhead is 55% smaller than the overhead of more traditional partition-based approaches with equivalent parameters. On the other hand, it can be tuned to save 15% in space requirements, when compared to storing a single global color histogram, while still being capable of yielding a 30% more effective retrieval.

1 Introduction

Image databases are becoming more and more common in several domains. The evolution of techniques for acquisition, transmission and storage of images have allowed the construction of very large databases, containing not only so-called traditional data, but also multimedia data, e.g., images. As image databases become larger and larger, the interest for content based image retrieval (CBIR) increases. This interest is partially motivated by the fact that representing images as textual data (e.g., keywords) is not adequate for image retrieval in large and heterogeneous image databases. This inadequacy stems from three main problems

[4]: (1) it is difficult for text to capture the perceptual salience of visual features, (2) text is not well suited for modeling perceptual similarity and (3) text descriptors reflect the point of view of the annotator, who is, usually, a different personal from the final user interrogating the system. Techniques for image databases management, including issues such as: creation, processing, representation, presentation, interfacing, organization, browsing, querying and indexing, have not followed the database growth at the same speed as traditional relational database management systems have and are still somewhat limited [17]. In addition, current techniques do not scale well and pose severe drawbacks when applied to heterogeneous databases. We consider heterogeneous a large set of images that are not strongly related; they belong to potentially different domains, were acquired with different devices, have different size and resolution and are stored in different graphic formats.

A distinguishing aspect of CBIR is that it is based on the availability of a representation (metadata) of the visual content of the images. The complex nature of images, and the complexity of the analysis algorithms, do not allow a direct comparison between two images and as a result, representative and summarized metadata, which may be obtained through automatic or semi-automatic processing, is needed. Clearly, the use of semi-automatic image processing techniques is not adequate nor sufficient for large image databases. The type of metadata used to access images has a direct impact on the internal organization of the retrieval system, on the way in which the retrieval is performed and on its effectiveness. CBIR is complex since different types of metadata may be associated with images. Such different types of metadata include [13]:

- Content-independent metadata - data which is not directly concerned with image content, but in some way related to it. Examples of such data are: the format, the authors name, date, location and ownership.
- Content-dependent metadata - data which refers to perceptual low/intermediate-level features, like color, texture, shape, spatial relationship and their combinations.
- Content-descriptive data - data which refers to content semantics. It is concerned with relationships of image entities with real-world entities or emotions, impressions and meaning associated with visual signs.

A CBIR system may be used in several application domains, for example, digital libraries, geographic information systems and visual search engines for the WWW. The advances in the area of CBIR are highly dependent on the collaboration among specialists from different areas, for example, images analysis/processing, databases, information retrieval and interfaces. In general, the goal of CBIR is to retrieve images similar to an image/sketch provided by the user. This contrasts with traditional databases where the goal is an exact-match of a set of attributes. Similarity-based retrieval differs from matching in the following aspects [4]. Matching is a binary partition, being intrinsically committed to deciding whether or not the object observed corresponds to a model. Similarity-based retrieval is, instead, the task of reordering database images according to their measured similarity to a query example. It is therefore concerned with ranking rather than classification. In matching, uncertainties and impressions are commonly managed during the process; features used to perform classification are chosen according to the problem. In retrieval by similarity, the user is part of the solution, in the sense that his/her interaction is required, e.g., it is he or she that defines a query, analysis the system response and possible refines the query.

There are two principal methods to query images given an image as an example [23]: finding the K most similar images or finding all images within a certain degree S of similarity relative to the reference image. Usually, the similarity metric is based on the distance between the feature vectors that represent two images. Sample images can be: a new image or sketch thereof provided by the user, an answer to a previous query, or an image belonging to a sample image set. These images may be also further edited and/or modified. In order to realize image-based queries, we believe that one needs: (1) a domain-independent characterization of the visual content of the images, (2) image processing techniques to automatically extract such visual characteristics, (3) a compact yet representative abstraction for these characteristics, (4) a similarity metric to effectively compare images and (5) indexing techniques to efficiently access relevant images in the database. The first two requirements are related to image metadata extraction and representation. The last three are related to CBIR efficiency and effectiveness. A great challenge in this area is to find the best compromise between these conflicting requirements.

In large and heterogeneous image databases, the above issues are even more critical. The universe of potential metadata is restricted by the constraint of efficient metadata extraction, as well as analysis and indexing. In other words, the metadata should be representative, simple and compact. The similarity metrics should also be simple from the computational complexity viewpoint. Finally, any semi-automatic image processing should be avoided as much as possible and not relied upon as they are not scalable. Thus, in order to tackle the problem of efficient retrieval in large heterogeneous image databases, one must focus on techniques which are automatic, efficient and scalable.

This paper deals with the first four requirements above. Our goal is to propose and evaluate a new technique for image abstraction and retrieval using color distribution. Our basic motivation is based on the fact that global color histograms are not able to adequately capture image content. In addition, alternative approaches pose a large overhead in terms of metadata representation/storage. Our main contribution is a histogram-based representation which yields a low overhead, while being able to capture more information about the spatial distribution of image's colors.

The remainder of this paper is organized as follows. Section 2 discusses some related works relative to color features and their spatial distribution. Section 3 presents a new CBIR approach, named Color-Shape, which is more robust and flexible than those discussed in Section 2. We also show that the user may easily tune it in order to calibrate the trade-off between space overhead and retrieval effectiveness. Section 4 presents a similarity metric, based on L1 distance, and which is common to all approaches that will be compared in the experiments described in Section 5. Section 5 evaluates the Color-Shape retrieval effectiveness in comparison to two other approaches (1) a global color histogram (GCH) and (2) a traditional partition-based approach that we named Grid. Finally, Section 6 presents our conclusions and directions for future work.

2 Related Work

With respect to the use of colors as the visual feature of interest when performing CBIR, there are two main issues which one should be concerned with. The first issue is the underlying model, which contains two subissues: color space and a suitable representation. The second issue is related to the spatial layout of the color distribution. In this section we review previous work related to both of these issues.

2.1 Color features

Color is a visual feature which is immediately perceived when looking at an image. Retrieval by color similarity requires that models of color stimuli are used, such that distances in the color space correspond to human perceptual distances between colors. Color stimuli are commonly represented as points in three-dimensional color space [4]. Color models can be classified in hardware-oriented models (RGB, CMY, YIQ) and user-oriented models (HSI, HSV, HSB, MTM, L*u*v*, L*a*b* and L*C*h*). The hardware oriented models are defined according to properties of the devices used to reproduce the colors (computer screen, color printer and TV monitor). User-oriented models are based on human perception of colors.

RGB is the most commonly used hardware-oriented scheme for digital images. Colors in RGB are obtained as the addition of the three primary colors: Red, Green and Blue. The RGB color space is a solid having the shape of a unit cube and is a non-uniform color space; uniform color spaces are spaces such that a color difference perceived by a human observer is approximated as the Euclidean distance between two points in the color space. Hence, uniform quantization of the RGB space results in perceptually redundant bins and perceptual holes. The Hue-Intensity-Saturation family (HSI, HSV and HSB), despite being user-oriented models, have similar problems to RGB. They do not represent color differences on a uniform scale. Perceptually uniform spaces, such as MTM, L*u*v*, L*a*b*, L*C*h* are more suited than RGB for CBIR [4].

A color histogram is a simple and well known approach to encode the low-level color information of an image [27, 10]. It is obtained by discretizing an image's colors and counting how many pixels belong to each color. Histograms, by themselves, do not include spatial or shape information and as a result, images with very different layout can have similar representations. Hence, retrieval of images based on color histograms are prone to yield a large number of false hits, specially when used with a large database of heterogeneous images. Histograms may also have problems in representing color content because of color quantization. If a perceptually uniform color space is chosen, uniform quantization will be appropriate. For non-uniform color spaces (RGB, HSV), a non-uniform quantization should be chosen to ensure a correct representation of the color space. Clearly, the choice of representative colors affects the perception of similar images.

QBIC system [3] uses a partition-based approach to extract and represent color features. For each grid cell relative to a rectangular partitioning superimposed on the image, the average Munsell color and the five most frequently occurring colors and their frequencies (i.e., a partial histogram) are computed. In Dimai's approach [8], the features that represent the color distribution in the L*a*b* color-space are the average color and the covariance matrix of the color channels. Androutsos et al [1] used color segmentation, in the HSV color-space, to extract regions of prominent color. Each region is described by its average color vector. Sciascio et al [19] also uses the average color to describe a cell content in a 4x4 grid superimposed on the image. In Color-WISE approach [21], it was used dominant hue and saturation values in the HSI color space. These values are determined from different parts of an image through a process of block-based histogram building and peak detection.

Stricker and Orengo [26] propose two approaches to color representation. The first one is a more robust version of the commonly used color histogram that uses cumulative color histograms. The cumulative histograms are always completely dense vectors, even if only a few colors of the discrete color space appear in each image. The robustness of this technique allows one to work with very coarsely quantized color spaces. The pertinence of cumulative histograms for CBIR lies on the property that the color similarity between two nearby histogram bins should be bigger than the color similarity between two further separated histogram bins. Zhang et al [28] proposed a variation of cumulative histograms. The modified procedure applies the cumulative histogram technique to each sub-range of the histogram. This technique was called Local Cumulative Histogram. The second approach presented by Stricker and Orengo [26], instead of representing the complete color distributions, represents only their dominant features, via the first three moments of each color channel in the HSV color space.

Appas et al [2] decompose an image into five regions and also represents the color feature of each region by the first three moments of each color channel in the HSV color space. Two differences distinguish this work from previous work: (1) the third moment is computed from the second moment (instead of the first one) and (2) the three moment values are combined into a single descriptor. Liang and Kuo [15] proposed an integrated wavelet coding system for CBIR. All the features (color, texture and shape) are based on wavelet coefficients and their energy distribution among sub-bands, across quantization layers and in the space. These features were extracted during the successive approximation quantization (SAQ) stage of the wavelet compression process. The obtained SAQ histograms are represented by their first three central moments (mean, variance and skewness), as proposed by Stricker and Orengo [26].

2.2 Spatial Distribution of Colors

Partitioning of the image data is an important factor in determining the functionality and efficiency of the large image storage and retrieval systems [24]. For instance, by breaking the images into smaller, more manageable units, it usually becomes easier for the systems to compress, store, access and retrieve the image data. However, no single partitioning scheme is known to be optimal for all image storage and retrieval applications. Several partitioning schemes are reviewed next. Each offers different advantages for access, storage and retrieval.

Spatial grids partition the images from space into equally sized blocks, where each block corresponds to a spatial portion of the image. A fixed decomposition leads to cells that can straddle image regions with different visual contents. In principle, this problem could be addressed by blocks of variable shape and size. However, determining these variable blocks seems to require the solution of problems, e.g., region segmentation, for which only partial and usually expensive solutions have been proposed. Gonzalez and Woods [11] describe some image segmentation techniques, including region-oriented segmentation techniques.

The QBIC system [3] decompose an image using two approaches: partition-based and region-based. The partition-based approach is similar to the method described in [18]. The images are divided into a 6x8 or a 9x12 grid of cells. The region-based approach uses an approximate segmentation of each image into a hierarchical set of colored rectangles. Sciascio et al [19] also uses a 4x4 grid to partition the images. Androutsos et al [1] uses color segmentation, in the HSV color-space, to extract regions of prominent color. The approach presented by Appas et al [2] decomposes an image into five regions (the center and the four corners).

The approach proposed by Guibas et al [12], decomposes images using a fixed quadtree [22]. In Leung and Ng's approach [14], each image has a 4-level multi-resolution representation. At the first level, the image is represented by a single color histogram. In the second level, the image is divided into four non-overlapping blocks, each one represented by one color histogram. In the next levels, each block is successively divided into four new blocks. The idea is the same as the quadtree-based approaches. The work of Sebe et al [20] decomposes images into three levels. The first level is the whole image itself. The second level is a 3x3 grid and the third level is a 5x5 grid. This decomposition results in 34 regions plus the image (level 1). The regions in this approach are of different sizes (according to their level) and overlap in different levels. The approach of Malki et al [16] is similar to the previous approach: they use a quadtree of three levels to decompose an image. Color-WISE approach [21] uses a fixed image partitioning scheme which allows overlapping blocks.

Dimai [8] proposed an approach which combines a global vector describing the color features of the whole image with a set of inter-hierarchical distances, which are based on a fixed partition of the image (nine non-overlapping fixed-size regions). The inter-hierarchical distances are the difference measured between feature vectors of a region and its sub-regions. Thus, only the difference between the global vector and the vector of one sub-region is stored. Chen and Wong [5] proposed an augmented color histogram that captures the spatial distribution of pixels in addition to the color distribution. The spatial information is incorporated by computing features from the spatial distance between pixels belonging to the same intensity color. The mean, variance and entropy of the distances are computed to form an Augmented Image Histogram. The augmented histogram therefore captures the spatial distribution of pixels with the same color relative to each other, instead of containing information about absolute spatial location of the pixels. The SFGraph [25] is an approach that partitions the images symmetrically in space and frequency. The SFGraph view elements correspond to wavelet sub-bands of portions of the images. The SFGraph embeds many image representations, such as the multiresolution pyramid, wavelet, tiled-wavelet, wavelet packet, spatial quadtrees and Flashpix. Flashpix [7] is another approach that combines the wavelet and spatial grids by partitioning the images into equal-volume tiles, at different resolutions.

3 The Color-Shape Approach

As described in the previous section, the most widely used technique to spatially locate the image's visual features partitions the image into fixed-size blocks and then extracts visual features for each block individually. Our main contribution in this paper is yet another simple, but very effective partition-based technique. As will be discussed, our technique may be considered complementary to the previous work. To the best of our knowledge, it is original in the way that visual features are encoded. Our motivation was to reduce the space overhead of partition-based approaches taking advantage of the fact that only a relatively low number of distinct values of a particular visual feature is present in most images. Indeed, we will use color features to investigate the idea. However, it is possible to encode any other visual feature of an image with the same idea. We will refer to our approach as Color-Shape.

Using 20,000 heterogeneous images in JPEG format, and the RGB color-space uniformly quantized in 64 colors, we obtained the data shown in Figure 1. In the average, there are only 28,71 colors per image from a total of 64 colors. Moreover, 90% of the image content corresponds to only 9 colors. These values show that, at least 55% of a color histogram has no useful information (bins with zero value) and that only 14% of its bins are able to describe 90% of the visual content of the image. If we use a local color histogram for each block of an image partitioning, the amount of useless information grows proportionally to the number of blocks.

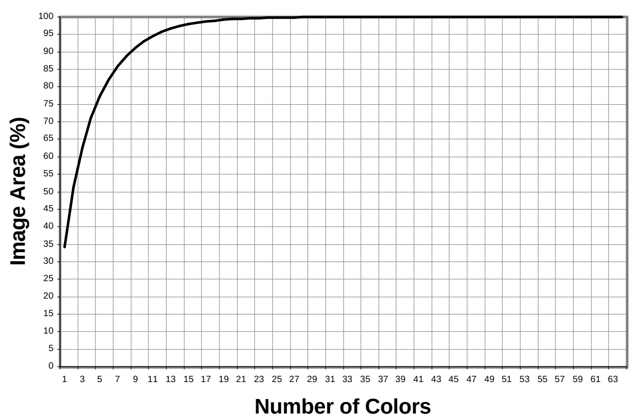

Figure 1: Color contribution for image content.

We thus aim to explore the fact that colors not present in an image need not to be represented. The color histogram does not allow this type of optimization directly. Thus, we will use what we call a *Color-Shape Histogram—CSH*. Using CSHs, the image abstraction is compact, yet representative. Consider an image partitioned into 8x8 non-overlapping cells and the RGB color-space uniformly quantized in 64 colors. A CSH for a given color c, $0 \leq c < 64$, is a set of 64 bins (one for each image cell), where the bins' values are described by the function $p(cell_k) = n_k/n$. In this function, $cell_k$ is the k^{th} cell of the image ($0 \leq k < 64$), n_k is the number of pixels in the $cell_k$ with color c and n is the total number of pixels in the image. An image composed by m colors is thus described by m color-shape histograms, each one describing the spatial distribution of one color. In this type of decomposition, if a color is not present in an image, its Color-Shape histogram does not need to be represented, nor stored. Figure 2 shows an image with a 2x2 grid superimposed and its respective Color-Shape histograms.

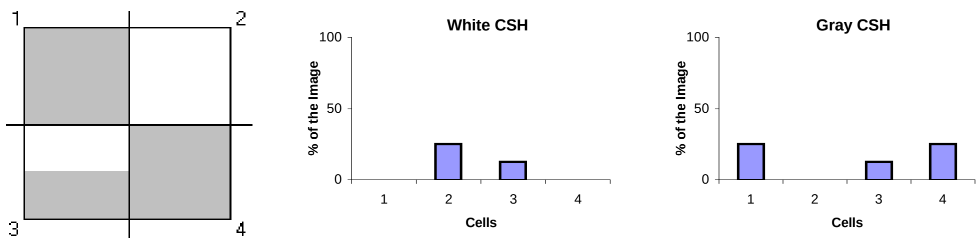

Figure 2: An example image with a 2x2 grid superimposed and the associated Color-Shape histograms.

Color-Shape histograms combine, in an elegant way, the information represented by local histograms in a partition-based approach, with the information of a global color histogram, while likely reducing the space overhead. The decomposition of an image into a grid of cells allows one to simultaneously locate a color inside the image and approximate its shape. Color-shape histograms may be represented directly as a set of bins or using other approaches described in section 2.1, like cumulative histograms or the initial moments of the spatial distribution. They can also be used in multiresolution approaches like quadtrees. We plan to tackle those issues in our future research.

4 Similarity metric

In this section, we will describe the similarity metric that will be used in all approaches compared in the experiments of Section 5. The chosen metric is based on the L_1 distance metric (Equation 1), where $h_q[i][j]$ and $h_d[i][j]$ represent the j^{th} bin of the i^{th} histogram used to represent the query image (h_q) and the database image (h_d), respectively. We assume that the histogram bins are normalized with respect to the image size, i.e., number of pixels.

$$D(h_q[i], h_d[i]) = \sum_{j=1}^{m} |h_q[i][j] - h_d[i][j]| \tag{1}$$

Recall that in the case of global color histograms (GCHs), there is only one histogram to be considered. In traditional partition-based approaches, there are a fixed number of local color histograms. In Color-Shape approach, there is a variable number of histograms per image. Also recall that a color not present in an image does not yield a CSH. However, even though it does not need to be stored, for the purpose of the metric computation, all of its (virtual) bins are assumed to be of zero height.

To normalize the result obtained with the L_1 distance, we divide it by the sum of the areas (number of pixels) of the regions described by each histogram $(a_q[i] + a_d[i])$. These areas are also normalized with respect to the images' sizes. The normalized metric is shown in Equation 2. Note that if we use GCHs, the denominator in Equation 2 is always 2, as usual and expected. However, in the Color-Shape approach, the areas being compared are, in general, smaller than the whole image, in fact, they are equal to the percentage of each image that the color being compared represents. This case requires one to use the sum of the areas explicitly to normalize the L_1 result (notice that the sum is not known *a priori*). So far, D_n measures the normalized distance between two histograms. The similarity between two histograms is the complement of the distance D_n. The similarity S between two images (Equation 3) is the weighted sum of the similarity between the histograms present in the images:

$$D_n(h_q[i], h_d[i]) = \frac{D(h_q[i], h_d[i])}{a_q[i] + a_d[i]} \tag{2}$$

The goal of the weight values is to normalize the similarity between two images and describe the importance of each compared histogram. We chose $w[i] = min(a_q[i], a_d[i])$. Clearly, the sum of the $w[i]$s is at most 1.

$$S(h_q, h_d) = \sum_{i=1}^{n} w[i] \times (1 - D_n(h_q[i], h_d[i])) \tag{3}$$

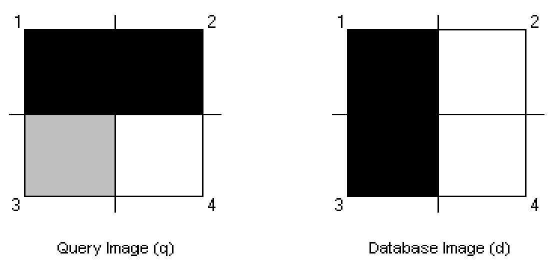

Figure 3: Images used to exemplify the application of the similarity metric with a 2x2 grid superimposed

The two images in Figure 3 will be used to exemplify the application of the similarity metric in three histogram-based approaches: (1) a global color histogram—GCH; (2) local color histograms (LCH) obtained from the cells of a grid superimposed on the image—Grid; (3) color-shape histograms—CSH. For simplicity, we divide the images into 4 cells (2x2 grid) in order to spatially locate colors. The cells are compared from top to bottom, left to right, and the color space has only three colors: black, gray and white, in this order. In Figure 3, q is the query image, and d is the database image to be compared against q. Figure 4 shows the histograms obtained from the query image using each approach. The first row has the

GCH. The second line set of four LCHs of the Grid approach, one for each cell. The bottom of the figure shows the three CSHs, one for each color. Figure 5 shows the same histograms for the database image.

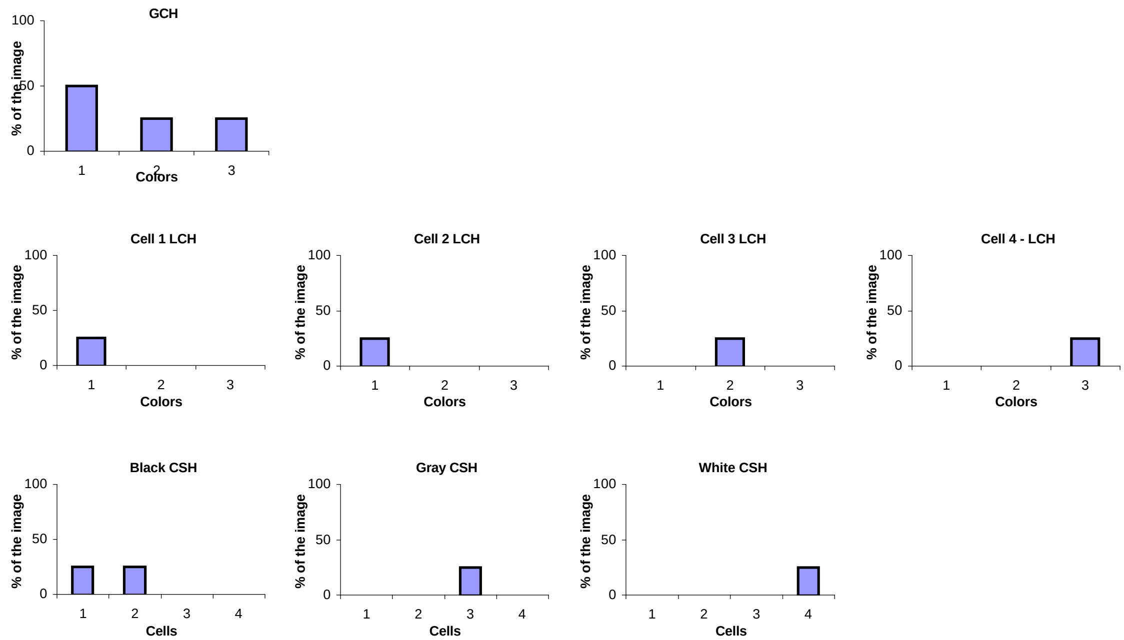

Figure 4: Histograms of the query image (Figure 3) in each approach

.

The GCH for q could be represented as $h_q = [0.5, 0.25, 0.25]$ meaning that it has 50% of black, 25% of gray and 25% of white pixels, respectively. Similarly, $h_d = [0.5, 0.0, 0.5]$. Using Equation 3 we have:

$$S_{GCH}(q,d) = 1 \times (1 - \frac{|0.5 - 0.5| + |0.25 - 0.0| + |0.25 - 0.5|}{2}) = 0.75 \tag{4}$$

As for the Grid approach, the normalized distance for the first cell is $D_n^1 = 0$, because both cells have only black pixels. For the other three cells, the distances are $D_n^2 = 1, D_n^3 = 1$ and $D_n^4 = 0$, respectively. In addition, the weights $w[i]$ are all the same, because all cells have the same relative area. Thus, we have:

$$S_{Grid}(q,d) = 0.25 \times (4 - (D_n^1 + D_n^2 + D_n^3 + D_n^4)) = 0.5 \tag{5}$$

Finally, using the Color-Shape approach, we compare the three CSHs (for black, gray and white) using Equation 2. We have 25% of black pixels in cells 1 and 2 of q, and in cells 1 and 3 of d (recall that the quantity of pixels in a cell is normalized with respect to the image size), hence: $D_n^{black} = \frac{|0.25-0.25|+|0.25-0|+|0-0.25|+|0-0|}{0.5+0.5} = 0.5$. Likewise we obtain $D_n^{gray} = 1$, and $D_n^{white} = 0.33$.

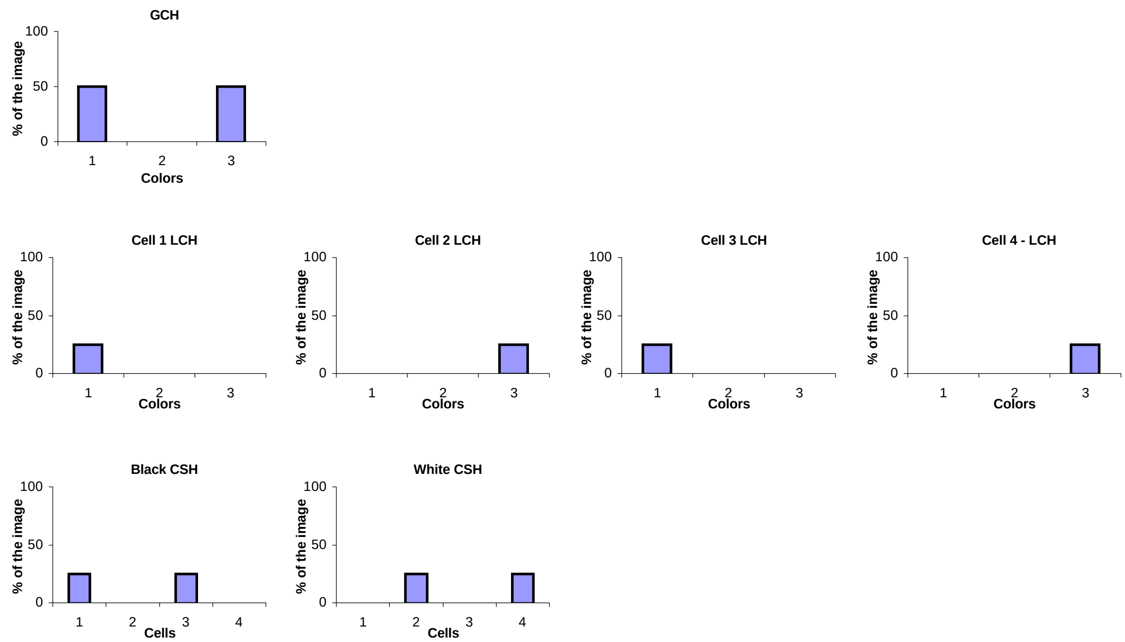

Figure 5: Histograms of the database image (Figure 3) in each approach.

Lastly, the normalized CSHs distances are complemented and weighted according to Equation 3 (notice that, unlike in the GCH and Grid approaches, the weights $w[i]$ are now variable, depending on the areas occupied by each color). Therefore, the similarity between the two images according to their CSHs is:

$$S_{CSH}(q,d) = 0.5 \times (1-0.5) + 0 \times (1-1) + 0.25 \times (1-0.33) = 0.42 \quad (6)$$

It is interesting to note that $S_{GCH} > S_{Grid} > S_{CSH}$, which is expected, given that the CSH uses more information (with not much more overhead), than the other approaches. As we will see in the results obtained, this more accurate similarity measure will yields a higher quality retrieval. There are several alternative distances that may be used to compare color histograms in an image-based query. We believe that the relative results should not depend on the particular metric used. That is to say that we expect any possible bias, if existent, to affect all approaches equally. We plan to investigate alternative similarity metrics in our future research.

5 Evaluation of Retrieval Effectiveness

We compare the Color-Shape approach with the two other approaches that we called global color histogram (GCH) and Grid. A GCH is described by a set of bins (one for each color in an image), each one with a height given by the function $p(c_k) = n_k/n$, where c_k is the k^{th} color (bin) in quantized color space, n_k is the number of pixels whose color is c_k and n

is the total number of pixels in the image. The Grid is a partition-based approach which decompose an image in a set of fixed size cells, accordingly with a grid superimposed on the image. For each cell a LCH is extracted. A LCH is equivalent to a global color histogram; the only difference is that n_k is the number of pixels whose color is c_k into the cell being represented, not in the whole image. The normalization is also relative to the whole image (n).

The dataset used in our experiments was a set of 20,000 JPEG images from a stock CD by Corel Corp. The images' content are heterogeneous but, at the same time, there is a large number of small subsets where the images are similar, allowing one to distinguish them from the others without ambiguity. Some of these subsets were used to evaluate query results. Out of the 20,000 images in the data set, we chose 15 images to be used as query images. Those images not only had a similar color distribution but also had the same semantics. For instance, there is a set of several distinct "Halloween" pumpkins, a set of Bonzai trees, and so forth. The answer sets for these query images were also built *a priori* in order to facilitate our evaluation. We call these sets the "required result set" (RRset) for each query image[1]. The average number of images in such RRsets was 11.27. The database creation and query times were not measured. At this point, our focus is only the quality evaluation of the image retrieval. The efficiency of the retrieval is an aspect which is tightly related to indexing structures/techniques, which is subject of further research.

In order to evaluate retrieval performance in a similarity-based scenario such as the one we are concerned with, we needed a performance measure that embodies the position in which target items appear in the retrieval sequence (ordered by some similarity measure). We choose to use a measurement called *normalized recall*, which was used in the QBIC project [9]. Normalized recall measures how close to the top of retrieved items the set of relevant items appears, compared to an ideal retrieval in which the most relevant items appear in the very top of the ranked answer. The better the retrieval, the higher the images in the RRset will rank in the returned answer set. Therefore, for each query image we measured the average rank (AvgR) of its RRset in the returned answer set. The relative ordering of the RRset elements is not important, what matters most is the location (rank) of all elements in the answer set. We calculated the ratio of the RRset average rank, relative to its ideal rank ($(|RRset|-1)/2$, where $|RRset|$ denotes the cardinality of the RRset, and the first image is assumed to have rank zero). We call this ratio the normalized AvgR (NAvgR) (or normalized recall in [9]). The NAvgR can be used to give a measure of average retrieval accuracy, or effectiveness. Perfect performance would yield NAvgR=1.

In the first experiment, the three compared approaches used the RGB color-space, uniformly quantized in 64 colors. Each image was decomposed into a fixed 8x8 grid, resulting in 64 cells per image. Although one may dispute the suitability of such parameters, we contend that they affect the three approaches in the same way, because all three approaches describe color features, which are histogram-based, and use the same similarity metric. In our judgment, this fact allows a correct comparative analysis. In the near future, we will investigate alternative color-spaces, quantization schemes, partition algorithms and other

[1]The sets of queries and respective RRsets can be seen at http://www.cs.ualberta.ca/~mn/CBIRdataset, however, due to copyright constraints the data set cannot be distributed nor copied from that site.

histogram representations. With these parameters, in the GCH approach, each image was represented by one color histogram with 64 bins (one for each of the quantized colors). In the Grid approach, each image was described with 64 LCHs (with 64 bins/colors), one for each cell. In the Color-Shape approach, each image was described by a variable number of CSHs, depending on the number of colors present in the images. Each CSH had 64 bins, one for each image cell. GCH is the most compact image abstraction. In the Color-Shape approach, the maximum number of histograms is 64. In average, for the 20,000 database images, only 28.71 histograms were needed per image, according to Figure 1. Therefore, in average, the CSHs required 55% less space than the Grid histograms. Note that this means potential savings of over 50% in space when the image abstraction is stored for further processing, e.g., indexing, which is a desirable feature of the proposed approach.

Table 1: NAvgR values for the RRsets of each query image.

		NAvgR		
Query	$\|RRset\|$	**GCH**	**Grid**	**Color-Shape**
A0004.JPG	13	15.32	3.80	4.56
A0190.JPG	11	40.44	11.51	6.91
A10127.JPG	14	40.97	5.24	3.98
A10219.JPG	12	18.17	6.36	4.51
A10576.JPG	9	23.39	25.17	6.06
A11895.JPG	8	21.43	8.86	9.71
A12632.JPG	6	4.47	6.33	3.80
A13719.JPG	8	50.54	3.64	11.25
A14937.JPG	14	3.59	1.31	1.78
A15344.JPG	14	7.28	1.86	1.42
A15434.JPG	19	13.22	3.89	1.87
A16144.JPG	17	49.81	1.80	2.25
A4171.JPG	8	11.46	23.50	3.64
A4959.JPG	6	29.80	1.60	3.00
A6124.JPG	10	8.87	5.22	4.04
Average	**11.27**	**22.58**	**7.34**	**4.59**

The results of first experiment are shown in Table 1. As can be seen, the Grid approach yields a NAvgR 68% smaller than the GCH approach. More importantly, however, the NAvgR obtained by using CSHs is 38% smaller than if we were using the Grid Approach. The Grid approach uses more information than GCH to represent an image. Thus, its NAvgR is considerably better than the GCH's NAvgR. Although the Color-Shape approach uses less histograms to represent an image than the Grid, it combines the global information of GCH with the local information of Grid in an elegant way. These two types of information yield in a smaller (therefore better) NAvgR for the Color-Shape approach. In this experiment, we also determined the average similarity of each RRset with the respective query image, in each approach. The average similarity values for the 15 RRsets are: 0.78 for GCH, 0.55 for Grid and 0.47 for Color-Shape. The GCH yields the larger average similarity. The Grid's similarity is 30% smaller than the GCH's similarity, and the Color-Shape's similarity is 15% smaller than the Grid's similarity. The large similarity values from GCH are due to a gross representation, which ultimately yields a large number of false-hits. In other words, a more strict similarity yields less false-hits. As the database size increases and becomes more heterogeneous, the probability of completely different images (from the viewpoint of the user) having greater similarity increases. Although the Grid and the Color-Shape approaches may be affected in the same way, within these approaches this effect is minimized simply because there is more information to distinguish images, e.g., some spatial information. The main motivation of using the GCH approach was to use it as a yardstick in comparison of the Grid and Color-Shape approaches.

The next two experiments explore the flexibility of the Color-Shape approach. Our goal is to reduce its space overhead in two ways. The first one reduces the number of cells in which we partition the images. The second one uses the information in Figure 1 to reduce the number of colors that must be represented and compared per image. Some experiments not discussed in this paper have indicated that the second approach is more adequate than simply reducing the color-space quantization.

Table 2: Effects of the reduction of the number of cells in the Color-Shape NAvgR.

		NAvgR				
Query	$\lvert RRset \rvert$	**8x8**	**6x6**	**4x4**	**3x3**	**2x2**
A0004.JPG	13	4.56	5.82	5.89	5.23	11.32
A0190.JPG	11	6.91	7.75	8.45	12.82	43.29
A10127.JPG	14	3.98	4.56	5.76	7.69	14.92
A10219.JPG	12	4.51	5.53	6.42	11.77	12.71
A10576.JPG	9	6.06	6.31	7.53	10.67	22.17
A11895.JPG	8	9.71	8.50	9.82	13.32	19.86
A12632.JPG	6	3.80	2.87	4.13	3.80	7.60
A13719.JPG	8	11.25	14.64	18.36	25.00	45.86
A14937.JPG	14	1.78	1.81	2.44	2.12	2.79
A15344.JPG	14	1.42	1.58	1.70	2.18	3.72
A15434.JPG	19	1.87	2.26	2.65	3.44	8.38
A16144.JPG	17	2.25	2.90	5.27	3.85	16.89
A4171.JPG	8	3.64	4.54	4.46	5.46	4.18
A4959.JPG	6	3.00	2.93	4.67	6.33	11.73
A6124.JPG	10	4.04	4.62	4.29	4.04	7.53
Average	**11.27**	**4.59**	**5.11**	**6.12**	**7.85**	**15.53**

Table 2 shows the effect of reducing the number of cells in which an image is partitioned in the Color-Shape's retrieval effectiveness (NAvgR). We compared the use of 8x8, 6x6, 4x4, 3x3 and 2x2 grids superimposed on the images. The results show that, as the number of cells decreases, the retrieval effectiveness also decreases or, in other words, the NAvgR value increases due to the loss of spatial information. There is less information to distinguish images, increasing the number of false-hits, and thus increasing the NAvgR (decreasing the retrieval effectiveness). It is interesting to note that, with a small number of cells (i.e 4x4), the Color-Shape NAvgR (6,12) is still 19% smaller (better) than the Grid approach NAvgR (7.34, Table 1). Using only 4x4 cells, each of the 28.71 CSHs has 16 bins, resulting in 459 bins per image, on average. This value is 89% smaller than the number of histogram bins used be the Grid approach (64x64=4096).

Table 3: Comparing partial content of the images in the Color-Shape approach. Images are partitioned in 8x8 cells

		NAvgR				
		100%	**95%**	**90%**	**80%**	**70%**
Query	$\|RRset\|$	**29* colors**	**12* colors**	**9* colors**	**6* colors**	**4* colors**
A0004.JPG	13	4.56	5.23	5.95	7.68	30.67
A0190.JPG	11	6.91	7.60	7.93	10.13	20.64
A10127.JPG	14	3.98	4.62	5.15	6.04	8.67
A10219.JPG	12	4.51	5.11	5.44	15.95	43.47
A10576.JPG	9	6.06	7.39	6.97	31.75	58.28
A11895.JPG	8	9.71	11.64	14.61	26.97	31.29
A12632.JPG	6	3.80	4.20	4.80	5.87	12.47
A13719.JPG	8	11.25	11.79	14.50	20.54	18.75
A14937.JPG	14	1.78	1.94	2.42	3.15	3.46
A15344.JPG	14	1.42	1.42	1.37	1.56	14.00
A15434.JPG	19	1.87	2.01	2.06	2.26	3.25
A16144.JPG	17	2.25	2.57	2.83	4.21	7.24
A4171.JPG	8	3.64	3.11	2.86	4.75	6.61
A4959.JPG	6	3.00	3.20	3.87	5.40	6.40
A6124.JPG	10	4.04	4.11	4.71	4.31	6.18
Average	**11.27**	**4.59**	**5.06**	**5.70**	**10.04**	**18.09**

* these are average values

As can be seen in Figure 1, on average, only about 29 colors (out of all 64 possible) are present in our 20,000 heterogeneous database images. This clearly shows also that only a small number of colors is responsible for the majority of the image content. The experiments described in Table 3 explore exactly this fact to optimize Color-Shape space overhead. The smaller the number of represented colors, the smaller the number of CSHs needed to describe an image. Still according to Figure 1, on average approximately 100% of the image content corresponds to 29 colors, 95% corresponds to 12 colors, 90% corresponds to 9 colors, 80% corresponds to 6 colors and 70% corresponds to 4 colors. We use these values to compute the effect of comparing a partial content of the images in the Color-Shape retrieval effectiveness (NAvgR). The results show that, representing only 80% of the images content results in an increase of 120% in the Color-Shape NAvgR. The range between 90%

and 100% results in smaller increases of the NAvgR in comparison with the space overhead reduction. For instance, for 90% of the image content we have on average 9 colors, resulting in 9 CSHs. This number is 69% smaller than the number of CSHs required to represent 100% of the image content (29). The NAvgR increases 24%, from 4.59 to 5.70.

In the last experiment, we compare the Grid and Global Color Histogram (GCH) approaches with some variations of the Color-Shape approach, using different number of cells and representing different percentages of the images. Our goal is to summarize the previous experiments, analyzing simultaneously the space overhead and the retrieval effectiveness of the traditional approaches in comparison with some variations of the Color-Shape approach. As the Color-Shape approach will use histograms with different numbers of bins according to the chosen parameters, the space overhead will use the total number of bins instead of the number of histogram to evaluate space overhead. The retrieval effectiveness is represented by the average NAvgR of the RRsets of the 15 query images.

We used the RGB color-space uniformly quantized in 64 colors. The Grid approach used 8x8 cells and represents the best case of the traditional partition-based approaches. If we use the Grid approach with less cells, its NAvgR becomes too large. On the other hand, using more cells, its space overhead becomes too large. The traditional partition-based approaches, as well as the GCH, does not allow for space reduction by partially representing the content of an image. Thus, their NAvgR will increase without reduction in space overhead.

Table 4: Results of an experiment that summarizes the previous ones

	Parameters		**Results**	
Approach	**N. of Cells**	**% of the image**	**N. of bins**	**NAvgR**
Global Color Histogram	1x1	100% (64 colors)	64	22.58
Grid	8x8	100% (64 colors)	4096	7.34
Color-Shape	8x8	100% (29* colors)	1856*	4.59
Color-Shape	6x6	95% (12* colors)	432*	5.62
Color-Shape	4x4	90% (9* colors)	144*	7.79
Color-Shape	3x3	80% (6* colors)	54*	16.19

* these are average values

The results of the last experiment are shown in Table 4. The Color-Shape approach, representing 100% of the image colors and partitioning the images in 8x8 cells, offers the best retrieval effectiveness, 38% smaller than the Grid approach, with a reduction of 55% in its space overhead. Alternatively, Color-Shape representing 80% of the images' colors and partitioning the images into 3x3 cells implies the smaller space overhead, 15% smaller than the GCH approach, yet yielding a NAvgR 29% smaller (better). It is also possible to obtain intermediate results, for instance, the Color Shape approach with 90% of the colors and using a 4x4 partition provides a NAvgR approximately equal to the Grid NAvgR, but with a respectable 96,5% of reduction in space overhead. In other words, using twice the number of bins that the GCH uses, it is possible to obtain a NAvgR approximately equal to the Grid

NAvgR. Other intermediate results may be obtained by choosing an adequate compromise between the number of cells and the percentage of the images (number of colors) being represented. It is possible to emphasize retrieval effectiveness, space overhead reduction, or both. These results confirm the potential and the flexibility of the Color-Shape approach in comparison to the traditional partition-based approaches.

6 Conclusions and Future Work

Our main contribution in this paper is a simple, yet very effective, variation of the partition-based techniques called Color-Shape. To the best of our knowledge, it is original in the way that visual features are encoded. Our motivation was to reduce the space overhead of partition-based approaches taking advantage of the fact that only a relatively low number of distinct values of a visual feature are present in most images. We used color features to verify the idea. However, it is possible to encode any other image visual feature with the same idea. Color-Shape histograms combine, in an elegant way, the information represented by local histograms in a partition-based approach, with the information of a global color histogram, while likely reducing the space overhead. The image abstraction is more compact, yet representative. The decomposition of an image into a grid of cells permits simultaneous location of a color inside the image and the approximation of its shape. We also proposed a similarity metric which is based on the L_1 distance metric.

We compared the Color-Shape approach with two other approaches which we called Global Color Histogram and Grid (chosen to represent the traditional partition-based approaches). These two approaches do not allow for space reduction by partially representing the content of an image. We compared the space overhead of each approach as well as their retrieval effectiveness. The retrieval effectiveness was measured using a technique called *normalized recall*, used in QBIC project. The comparison used only image-based queries. The data set used in our experiments was a set of 20,000 heterogeneous JPEG images from a stock CD by Corel Corp. Out of this data set, we chose 15 images to be used as query images. The answer sets for these query images were also built *a priori*. The three compared approaches used the RGB color-space uniformly quantized in 64 colors. Each image was decomposed in at most 64 cells to spatially locate color features.

Table 4 summarizes our findings. We compared the Grid and Global Color Histogram approaches with some variations of the Color-Shape approach, using different number of cells and representing different percentages (number of colors) of the images. These results confirm the potential and the flexibility of the Color-Shape approach in comparison to the traditional partition-based approaches. It is possible to emphasize retrieval effectiveness, space overhead reduction or both. The Color-Shape approach is able to offer the best retrieval effectiveness, 38% smaller than the Grid approach with a reduction of 55% in the space overhead. If desired, the Color-Shape may imply the smallest space overhead, 15% smaller than the GCH approach with a NAvgR 29% smaller (better).

Future research will concentrate on the Color-Shape approach which seems to be promising. We will investigate the effect of retrieval effectiveness using different color spaces, for example, HSV and uniform spaces such as L*a*b* and L*u*v* [4]. Accordingly to the color-space, we will also investigate some alternative quantization schemes and alternative similarity metrics. Another topic of interest is alternative representations for the Color-Shape histograms, for instance, cumulative histograms or initial moments of their spatial distribution. Experiments using object-based queries and the study of possibly using automatic image segmentation techniques to decompose images in a more consistent way will also be pursued. To optimize query processing, we will investigate indexing structures, for example the M-tree [6]. The M-tree is a promising approach because it indexes the distance between images and as such, it does not suffer from the problems related to the higher dimensionality of the indexing space (usually histograms are mapped to high-dimensional spaces).

7 Acknowledgments

Renato O. Stehling was supported by a Graduate Scholarship from FAPESP, Brazil. Mario A. Nascimento was partially supported by a Research Grant from NSERC, Canada. Alexandre X. Falcão was partially supported by a Research Fellowship Grant from CNPq, Brazil. The authors thank Curt Mastaller for suggestions which improved the paper's presentation.

References

[1] D. Androutsos, K.N Plataniotis, and A.N. Venetsanopoulos. Vector angular distance measure for indexing and retrieval of color. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases VII*, volume 3656, pages 604–613, 1999.

[2] A.R. Appas, A.M. Darwish, A.I. El-Desouki, and S.I. Shaheen. Image indexing using composite regional color channels features. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases VII*, volume 3656, pages 492–500, 1999.

[3] J. Ashley, R. Barber, M. Flickner, J. Hafner, D. Lee, W. Niblack, and D. Petkovic. Automatic and semi-automatic methods for image annotation and retrieval in QBIC. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases III*, volume 2420, pages 24–35, 1995.

[4] A. Del Bimbo. *Visual Information Retrieval.* Morgan Kaufmann, 1999.

[5] Y. Chen and E.K. Wong. Augmented image histogram for image and video similarity search. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases VII*, volume 3656, pages 523–429, 1999.

[6] P. Ciaccia, M. Partella, and P. Zezula. M-tree: An efficient access method for similarity search in metric spaces. In *Proc. of* 23^{th} *Intl. Conf. on Very Large Databases*, pages 426–435, 1997.

[7] Kodak Company. Flashpix format and architecture white paper. http://www.kodak.com/, July 1996.

[8] A. Dimai. Spatial encoding using differences of global features. In *Proc. of SPIE - Storage and Retrieval for Image and Video Databases IV*, volume 3022, pages 352–360, 1997.

[9] C. Faloutsos, W. Equitz, M. Flickner, W. Niblack, D. Petkovic, and R. Barber. Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3(3/4):231–262, 1994.

[10] B. Funt and G. Finlayson. Color constant color indexing. *IEEE Trans. Pattern Analysis Machine Intelligence*, 17(5):522–529, 1995.

[11] R.C. Gonzalez and R.E. Woods. *Digital Image Processing.* Addison-Wesley, 1992.

[12] L.J. Guibas, B. Rogoff, and C. Tomasi. Fixed-window image descriptors for image retrieval. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases III*, volume 2420, pages 352–362, 1995.

[13] V. Kashyap, K. Shah, and A. Sheth. Metadata for building the multimedia patch quilt. In *Multimedia Data Systems: Issues and Research directions.* Springer-Verlag, 1995.

[14] K-S. Leung and R. Ng. Multiresolution subimage similarity matching for large image databases. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases VI*, volume 3312, pages 259–270, 1998.

[15] K.-C. Liang and C.-C.J. Kuo. Implementation and performance evaluation of a progressive image retrieval system. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases IV*, volume 3312, pages 37–48, 1998.

[16] J. Malki, N. Boujemaa, C. Nastar, and A. Winter. Region queries without segmentation for image retrieval by content. In *Proc. of Proc. of* 4^{th} *Intl. Conf. on Visual Information Systems*, pages 115–122, 1999.

[17] M.T. Maybury. Intelligent multimedia information retrieval. In *Intelligent Multimedia Information Retrieval.* AAAI/Press / The MIT Press, 1997.

[18] H. Sakamoto, H. Suzuki, and A. Uemori. Flexible montage retrieval for image data. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases II*, volume 2185, pages 25–33, 1994.

[19] E. Di Sciascio, G. Mingolla, and M. Mongiello. Content-based image retrieval over the web using query by sketch and relevance feedback. In *Proc. of Proc. of* 4^{th} *Intl. Conf. on Visual Information Systems*, pages 123–130, 1999.

[20] N. Sebe, M.S. Lew, and D.P Huijsmans. Multi-scale sub-image search. In *Proc. of* 7^{th} *ACM Int. Conf. (part 2) on Multimedia*, pages 79–82, 1999.

[21] I.K. Sethi, I. Coman, B. Day, F. Jiang, D. Li, J. Segovia-Juarez, G. Wei, and B. You. Color-wise: A system for image similarity retrieval using color. In *Proc. of SPIE - Storage and Retrieval for Image and Video Databases IV*, volume 3312, pages 140–149, 1998.

[22] E. Shusterman and M. Feder. Image compression via improved quadtree decomposition algorithms. *IEEE Trans. Image Processing*, 3(2):207–215, 1994.

[23] J.R. Smith. *Integrated Spatial and Feature Image Systems: Retrieval, Compression and Analysis.* PhD thesis, Graduate School of Arts and Sciences, Columbia University, Feb 1997.

[24] J.R. Smith, V. Castelli, and C.-S. Li. Adaptive storage and retrieval for large compressed images. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases VII*, volume 3656, pages 467–478, 1999.

[25] J.R. Smith and S.-F. Chang. Joint adaptive space and frequency graph basis selection. In *Proc. of IEEE Int. Conf. Image Processing (ICIP)*, pages 702–705, 1997.

[26] M. Stricker and M. Orengo. Similarity of color images. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases III*, volume 2420, pages 381–392, 1995.

[27] M.J. Swain and D.H.Ballard. Color indexing. *Intl. J. Computer Vision*, 7(1):11–32, 1991.

[28] Y.J Zhang, Z.W. Liu, and Y. He. Comparison and improvement of color-based image retrieval tchniques. In *Proc. of SPIE -Storage and Retrieval for Image and Video Databases VI*, volume 3312, pages 371–382, 1998.