A Compact and Efficient Image Retrieval Approach Based on Border/Interior Pixel Classification

Renato O. Stehling[*]
Institute of Computing
University of Campinas
Brazil
renato.stehling@ic.unicamp.br

Mario A. Nascimento[†]
Dept. of Computing Science
University of Alberta
Canada
mn@cs.ualberta.ca

Alexandre X. Falcão[‡]
Institute of Computing
University of Campinas
Brazil
afalcao@ic.unicamp.br

ABSTRACT

This paper presents *BIC* (*B*order/*I*nterior pixel *C*lassification), a compact and efficient CBIR approach suitable for broad image domains. It has three main components: (1) a simple and powerful image analysis algorithm that classifies image pixels as either border or interior, (2) a new logarithmic distance (*dLog*) for comparing histograms, and (3) a compact representation for the visual features extracted from images. Experimental results show that the *BIC* approach is consistently more compact, more efficient and more effective than state-of-the-art CBIR approaches based on sophisticated image analysis algorithms and complex distance functions. It was also observed that the *dLog* distance function has two main advantages over vectorial distances (e.g., L_1): (1) it is able to increase substantially the effectiveness of (several) histogram-based CBIR approaches and, at the same time, (2) it reduces by 50% the space requirement to represent a histogram.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*image databases*; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*abstracting methods, indexing methods*.

General Terms

Measurement, Algorithms, Experimentation.

Keywords

Content-Based Image Retrieval, CBIR, Distance Function, Color Histogram, Image Analysis

1. INTRODUCTION

Recently, there has been a great interest in techniques for *Content-Based Image Retrieval* – CBIR. This interest has spurred from the need to efficiently manage and search large volumes of multimedia information, mostly due to the exponential growth of the World-Wide-Web (WWW).

CBIR is performed based on abstract descriptions of the images that are extracted during the image analysis phase. Image analysis algorithms may depend on the properties of the images being analyzed. These algorithms are usually distinct for different image domains, and gradually change when the focus moves from a narrow to a broad image domain [12]. A *narrow image domain* has a limited and predictable variability in all relevant aspects of its appearance. Collections of fingerprints, faces recorded over a clear background and X-rays of the human brain are examples of narrow image domains. A *broad image domain*, on the other hand, has an unlimited and unpredictable variability of the image's content. In general, the interpretation of the image's content is not unique, and the collection of images is very large. As a consequence, it is not possible to use semi-automatic techniques and domain-dependent knowledge during the analysis and comparison of images. The broadest collection of images nowadays is likely formed by the very large amount of images available at the WWW.

In this paper, our focus is on CBIR techniques suitable for broad image domains. In this scenario, low-level visual features of the images such as color and texture are especially useful to represent and to compare images automatically. In fact, color is the most commonly used low-level feature in CBIR systems. Color-based image retrieval techniques can be classified into three main categories: (1) *global* approaches [2, 6], (2) *partition-based* approaches (e.g. [11, 16]) and (3) *regional* approaches (e.g. [5, 13]). Each of these categories poses a distinct compromise among the complexity of the image analysis algorithm, the amount of space required to represent the visual features extracted from images, the complexity of the distance function used to compare these features, and the retrieval effectiveness [15].

Global approaches describe the visual content of an image as a whole without spatial or topological information. Partition-based approaches introduce some spatial information about the visual content of the images decomposing them in spatial cells according to a fixed scheme, and describing the content of each cell individually. Regional approaches are a natural evolution of partition-based approaches in the sense that, instead of decomposing images in a

[*] Supported by FAPESP, Brazil.

[†] Partially supported by NSERC, Canada.

[‡] Partially supported by CNPq, Brazil.

CIKM'02, November 4–9, 2002, McLean, Virginia, USA.
Copyright 2002 ACM 1-58113-492-4/02/0011 ...$5.00.

fixed way, these approaches exploit their visual content to achieve a more flexible and robust segmentation. Ideally, the obtained regions correspond to the high-level concept of objects that a user can easily distinguish when he/she looks at the image. Unlike partition cells, segmented regions from two distinct images may have different sizes, positions and shapes. Moreover, the number of regions in two images may be different.

In general, the result of the image analysis algorithm in regional CBIR approaches cannot be used directly to represent and to compare images, because the number of segmented regions is usually very high. A precise description and a precise comparison of a large number of regions are very expensive in computational terms. As a consequence, the image analysis result is post-processed in order to reduce the number of segmented regions, and also to simplify the description of the remaining regions. However, this simplification certainly affects the effectiveness of the approach.

In this paper, we propose a different alternative for CBIR in broad image domains. Instead of using sophisticated image analysis algorithms whose result should be simplified to keep the problem tractable in computational terms, we propose the use of a simple yet powerful image analysis algorithm, whose result can be efficiently stored and compared without simplification. We call our approach *BIC* (*B*order/*I*nterior pixel *C*lassification). The *BIC* approach has three main components: (1) a simple and powerful image analysis algorithm that classify image pixels as border or interior; (2) a new logarithmic distance to compare histograms; (3) a compact representation for the visual features extracted from images. The key for the compactness, efficiency and effectiveness of *BIC* is the consistency among the analysis, representation and comparison of the images.

The remainder of this paper is organized as follows. Section 2 reviews four existing CBIR approaches which were implemented and used as reference in our experiments. In Section 3, we discuss limitations and drawbacks of the use of regional CBIR approaches in broad image domains, and introduce the *BIC* approach. Section 4 presents our experimental setup in terms of reference collection of images, query images, set of relevant images and retrieval effectiveness measures. Our experimental results are discussed in Section 5. Finally, Section 6 offers conclusions and directions for future work.

2. RELATED WORK

In this section we review four CBIR approaches used in our experiments as references: GCH, CCV, Grid9 and CBC. These approaches were implemented and their effectiveness are compared in Section 5. A more complete discussion about color-based image retrieval and existing color-based CBIR approaches can be found in [15].

The most simple and well-known approach to encode the color information present in an image is the global color histogram - GCH [2, 6]. A GCH is made of a set of ordered values, one for each distinct color, representing the probability of a pixel being of that color. Uniform quantization schemes are often used to reduce the number of distinct colors present in an image and the pixel count is normalized to avoid scaling bias. Note that such histograms are actually feature vectors which can be easily compared by means of well-known vectorial metrics such as L_1 (City-block) or L_2 (Euclidean).

Our implementation of the GCH approach rely on the RGB color-space uniformly quantized in 64 distinct colors, and the L_1 vectorial distance to compare histograms. The uniform quantization of the RGB space in 64 colors is largely adopted in practice and seems to be a good uniform quantization scheme for the RGB color-space [16]. We normalize the pixel count of each histogram bin between 0 and 255. This normalization is helpful because if we approximate the pixel count to integer values in the interval [0,255], we are able to represent a histogram bin using only one byte of memory. We have also observed in practice that there is no clear advantage in using more than 255 distinct values per histogram bin.

Pass et al [10] proposed the Color-Coherence Vector (CCV), a variation of the GCH that improves its effectiveness. The idea consists of classifying each image pixel as coherent or incoherent, depending upon whether the pixel is part of a larger and connected region with a homogeneous color. Pixels belonging to connected regions larger than 1% of the image size are considered coherent, while pixels belonging to connected regions smaller than 1% of the image size are considered incoherent. One color histogram is computed considering only the coherent pixels and a second color histogram is computed considering only the incoherent pixels. The combination of these two histograms is the so-called Color-Coherence Vector – CCV.

The Grid9 is probably the simplest partition-based CBIR approach. It consists of decomposing images according to a 3×3 grid of equally sized non-overlapping cells. The content of each partition cell is described individually, by means of a local color histogram – LCH. The distance between two Grid9 representations is the average of the L_1 distances between the LCHs of the corresponding partition cells.

CBC is a regional CBIR approach based on a fully automatic clustering algorithm which runs in time $O(n \log n)$, where n is the size of the input image (number of pixels) [13]. In CBC, each image is decomposed in a set of disjoint connected regions. Each region is larger (in number of pixels) than a threshold size s_0. Additionally, all pixels of a region have a predefined degree of color similarity, according to a threshold color-distance d_0. We denote the procedure to obtain these regions CBC(d_0, s_0), where the thresholds d_0 and s_0 are parameters defined by the user. Each image region obtained with CBC is characterized by a 6D feature vector (L, a, b, s, h, v), where L, a, b represents the average Lab color of the region, s is the size (number of pixels) of the region normalized by the image size, and h, v are the normalized horizontal and vertical coordinates of the geometric center of the region.

In this paper, we adopted the CBC(3, 0.1) configuration[1] since it offers an intermediate compromise between the number of obtained regions (which affects the space overhead and the query processing time) and the retrieval effectiveness, and it has been shown [13] to outperform other well-known CBIR techniques (e.g., CCV). Finally, two images (sets of connected regions) are compared using the MiCRoM (Minimum-Cost Region Matching), a true-metric distance [14]. The main idea of MiCRoM consists of modeling the comparison of segmented images as a *minimum-cost network flow problem* [1].

[1] A set of 50 heterogeneous images segmented with the CBC(3 0.1) algorithm can be found at: `http://db.cs.ualberta.ca/mn/BIC/cbc-sample.html`.

3. THE BIC APPROACH

During the last years, we have worked with regional CBIR approaches. In this period, we could observe the potential and also the limitations of these approaches when applied to large collections of heterogeneous images (broad image domains). In this kind of application, it is not possible to assume any *a priori* knowledge about the properties of the images being analyzed. As a consequence, the image analysis algorithm and also the distance function used to compare segmented images should be as general as possible.

Our experience taught us that, even using very general image properties and automatic segmentation algorithms, it is possible to obtain very good segmentation results in the sense that the obtained regions match very much with the visual properties observed by users. The main drawback of these algorithms is that sometimes the obtained regions are only part of a real object, i.e., an object an user would likely identify by looking at the image. Thus, it does not have a semantic by itself and should be combined with some neighbor regions in order to represent a meaningful object. This problem is treated in general at query time, by using complex distance functions to compare weakly segmented images.

A second drawback of the automatic image segmentation algorithms is that the criterion of homogeneous visual properties usually leads to a super segmentation of the image. As a result, a precise representation of the obtained regions is prohibitive in terms of storage space, and their comparison using a complex distance function is impractical. The aforementioned problems become even more critical if one recalls that the number of regions per image is variable and the obtained regions are also variable in size, shape and spatial location.

In order to keep the problem of representing and comparing segmented images tractable, the output of the segmentation algorithm is usually simplified, relaxing properties in order to preserve only a few regions, and also representing approximately the remaining regions. As it is not possible to use additional knowledge about the content of the images (domain-dependent knowledge) to perform this simplification, the consequence is that the effectiveness of the approach is reduced in the same proportion in which the problem is simplified. If the result of the image analysis algorithm must be relaxed in order to keep the problem tractable in computational terms, it is very likely that the algorithm used is not the most adequate one for the problem in hand.

As we show in forthcoming sections, the key to reach efficient and effective CBIR systems in broad image domains is the use of simple and robust image analysis algorithms whose result can be preserved (without approximations) during the representation and the comparison of the visual features. There is no point to use complex image analysis algorithms if the properties of these algorithms must be relaxed and sometimes discarded in order to make the representation and the comparison of the images a tractable problem.

Next we present *BIC* (Border/Interior pixel Classification), a new CBIR approach suitable for broad image domains. The *BIC* approach has three main components: (1) a simple and powerful image analysis algorithm that classifies image pixels as border or interior; (2) a new logarithmic distance to compare histograms; (3) a compact representation for the visual features extracted from images. Each of these components is explained in details in the following subsections.

3.1 Image Analysis

The algorithm for image analysis in *BIC* approach rely on the RGB color-space uniformly quantized in 4×4×4=64 colors. It is important to notice that any other color-space and quantization scheme could be used as well. We chose this configuration because it is largely used and it is effective, as discussed in Section 2. Another reason is to have fair comparisons with other histogram-based CBIR approaches we have implemented that also rely on the same scheme (RGB uniformly quantized in 64 colors).

After the quantization step, image pixels are classified as border or interior pixels. A pixel is classified as border if it is at the border of the image itself or if at least one of its 4-neighbors (top, bottom, left and right) has a different quantized color. A pixel is classified as interior if its 4-neighbors have the same quantized color. It is important to observe that this classification is mutually exclusive (either a pixel is border or it is interior) and it is based on a inherently binary visual property of the images. We choose 4-neighbors instead of 8-neighbors because, given the simplicity and generality of the problem, the use of 4-neighbors is able to reduce the image analysis complexity without perceptual losses in terms of retrieval effectiveness.

After the image pixels are classified, one color histogram is computed considering only border pixels, and another color histogram is computed considering only interior pixels. In this way, we have the border/interior classification represented for each quantized color. A binary classification of image pixels was also proposed within the CCV approach (Section 2). However, the CCV binary classification is based on a non-binary visual property of the images – the size of the connected components. In order to have a binary classification in CCV, an empirical size threshold was introduced and most of the useful information about the size of the connected components was lost in this reduction. Moreover, the approach may be very sensitive to the chosen threshold that, in practice, should vary according to the visual content of the images. The consequence is that the CCV approach is only a little bit more effective than a simple GCH, as shown in Section 5. The implication of the approximation introduced in CCV in terms of effectiveness follows the discussion presented in Section 3.

The classification of the pixels in border/interior for each quantized color is much more discriminative than a simple GCH or CCV, as shown in the experimental results of Section 5. This discriminative power can be analyzed for each individual color in terms of shape, texture and connected components. If the number of interior pixels for a given color is smaller than the number of border pixels for the same color, than at least one of the following visual properties is true: (1) the color is distributed in relatively large regions with very irregular shape; (2) the color is distributed in small connected regions such that the border of each region is larger than its interior; (3) the color is part of an image region that is rich in texture information. Similarly, if the opposite situation is true, i.e., the number of border pixels for a given color is smaller than the number of interior pixels for the same color, than we can conclude that (4) the color is distributed in relatively large and homogeneous

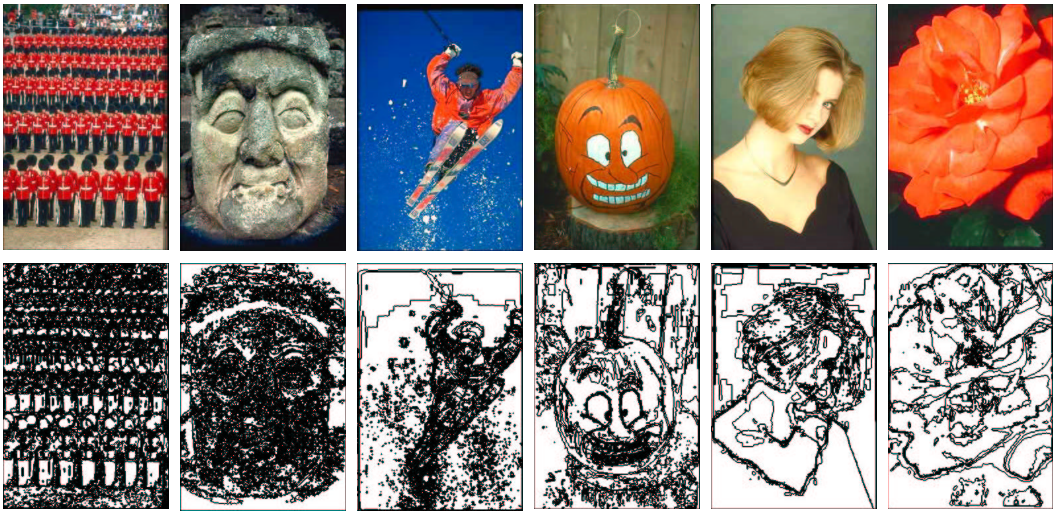

Figure 1: Examples of the result of the *BIC* pixel classification. (A set of 50 images analyzed in terms of border/interior pixels can be viewed in color at: `http://db.cs.ualberta.ca/mn/BIC/bic-sample.html`.)

regions with regular shape. The degree to which each of the four aforementioned visual properties is true depends on the portion of the image covered by and also on the proportion between border/interior pixels for each quantized color.

Figure 1 shows examples of images analyzed in terms of border and interior pixels. The original images are at the top row. The corresponding binary images showing border pixels in black and interior pixels in white are at the bottom row. The amount of border pixels is highest at the leftmost image (soldiers), and gradually diminishes when we move toward the rightmost image (flower).

3.2 dLog Distance Function

As discussed in previous section, each image is described within *BIC* by means of two color histograms with 64 bins each (one for each quantized color). In fact, these two histograms can be stored and compared as a single histogram with 128 bins. As such, we are able to use any vectorial distance function like L_1 or L_2 to compare the *BIC* visual features. The main advantage of vectorial distances is their efficiency in comparing histograms. Moreover, they allow the use of spatial or metric access methods to speedup query processing [15]. The use of access methods is important for large collections of images, as the query processing time should not increase in the same proportion that the image collection increases.

Vectorial distances have also well-known limitations. One of such limitations is that a high value in a single histogram bin dominates the distance between two histograms, no matter the relative importance of this single value [6, 8]. If we think about an image in terms of background/foreground regions, in general it is true that the foreground determines the semantic of the image and as such, it is more important in determining the similarity among images. It is equally true that, in general, the background covers the majority of the image area. Thus, the regions that compose the background are usually larger than the regions that compose the foreground. For instance, consider a set of images where the background covers 60% of the image's content and this background is homogeneous in the sense that it can be represented in just one histogram bin. Now imagine that we perform a similarity search using one of such images as example. What does happen when a vectorial distance is used to compare these histograms? Images having a background with the same color but a different foreground are retrieved ahead of any other image having the same foreground (a high degree of semantic similarity) but a background with a different color.

In order to deal with this distortion using only the information available within the histogram representation, we propose the *dLog* distance function. The *dLog* function compares histograms in a logarithmic scale, and is defined as:

$$dLog(q,d) = \sum_{i=0}^{i<M} |f(q[i]) - f(d[i])| \tag{1}$$

$$f(x) = \begin{cases} 0, & \text{if } x = 0 \\ 1, & \text{if } 0 < x \leq 1 \\ \lceil \log_2 x \rceil + 1, & \text{otherwise} \end{cases} \tag{2}$$

In the previous equation, q and d are two histograms with M bins each. The value $q[i]$ represents the i^{th} bin of histogram q and $d[i]$ represents the i^{th} bin of histogram d. The histogram bins are normalized between 0 and 255, as discussed in Section 2. A similar, but experimentally defined encoding function $f(.)$ was also used in [8].

The comparison of histograms with the *dLog* function does not solve the problem of histogram bins with very high values, but diminishes its effects in most of the situations. In a log-scale, the difference between the largest and the smallest distances between histogram bins becomes smaller than in the original scale. In the original scale, the smallest distance between histogram bins is zero (both images have the same amount of a particular color) and the largest distance is 255 (when the images have just one color and they are different). In our log-scale, the smallest distance is 0 and the largest distance is just 9. The range of distances in the original scale is thus $255/9 = 28$ times larger than in the proposed log-scale.

In Section 5 we apply the *dLog* distance function to compare histograms in different histogram-based CBIR approaches. In all cases, the use of the *dLog* function (instead of L_1) increases substantially the effectiveness of the approaches, making simple approaches such as GCH almost as effective as a regional approach such as CBC.

3.3 Representation of Visual Features

When histograms are compared using the *dLog* distance function, it is possible to store the result of the $f(x)$ function (Equation 2) instead of the normalized pixel count. The advantages of this log-based representation for histograms are: (1) the comparison of the histograms according to the *dLog* distance becomes computationally simpler; (2) the histogram can be stored in half of the space of the original representation; (3) as in [9], we can interpret, represent, index and compare histograms as binary signatures.

If the log-based representation is adopted, we can compare histograms using simply the L_1 distance. A careful look at Equation 1 reveals that the *dLog* distance is in fact an L_1 distance of the log of the pixel count – $f(x)$. If $f(x)$ is already computed and stored, all we have to do is just compare the log-based represented histograms using the L_1 vectorial distance. Moreover, observing Equations 1 and 2, and remembering that $0 \leq x \leq 255$, we perceive that $0 \leq f(x) \leq 9$. Thus, $f(x)$ can assume only 10 distinct values and these values can be stored in just 4 bits ($10 < 2^4$). This means that the log-based representation of histograms requires half of the space necessary to store the normalized pixel count (original representation).

The log-based representation allows a reduction of 50% in the required storage space for any histogram-based CBIR approach. In the particular case of the *BIC* approach, each *BIC* histogram has 128 bins (64 for border pixels and 64 for interior pixels). Thus, it is possible to store a *BIC* histogram in just 64 bytes of memory. This is a very compact representation for the visual features of an image. As an example, it is possible to store 16,000 *BIC* histograms in just 1Mbyte of memory. Considering a single desktop PC with 1Gbyte of free RAM memory, it is possible to keep in main memory (for the purpose of similarity search) the *BIC* representation of approximately 16 million images. High-end workstations can thus maintain fairly large collections of images in memory, completely avoiding the necessity of disk-based access methods to speedup query processing.

4. EXPERIMENTAL SETUP

In our experiments we adopted the widely used query-by-example (QBE) paradigm, as it seems to be the most adequate way to submit queries in CBIR systems based on low-level visual features. In QBE, an image is given as a visual example of the information needed. This image is analyzed and visual features are extracted. These features are used to measure the distance between the query image and the images stored in the image database. The stored images are retrieved in increasing order of their distance to the query image (similarity-search).

The purpose of our experiments is to evaluate the effectiveness of the similarity-search of different CBIR approaches in retrieving relevant images ahead of non-relevant ones. Effectiveness evaluation is a very complex task. While in textual information retrieval there are several reference collections of documents available (e.g., CACM, ADI, INSPEC, Medilars and ISI) and even a full conference (TREC) dedicated to the issue of effectiveness evaluation [17], in the domain of CBIR the situation is quite different. The CBIR community has not been nearly as active in this respect, though some work has begun to appear recently (e.g. [4, 7]).

In order to evaluate CBIR effectiveness, it is necessary to have at least a reference collection of images, a set of query images, a set of relevant images for each query image (ground truth), and adequate retrieval effectiveness measures. Next we discuss how we dealt with these requirements in our experiments.

We are using as reference a heterogeneous collection of 20,000 JPEG images from a Corel stock CD[2]. This collection has approximately 200 distinct image domains, each one composed of approximately 100 images. We believe this is a sufficiently large number of distinct domains (and also images per domain) for the purpose of our evaluation study.

Out of the reference collection, we selected 50 images of distinct domains to be used as query images. Once the query images were selected, the next step was to establish the set of images inside the reference collection that we accept as relevant for each query image. We call this set of relevant images the *relevant result set* (RRSet) of a query image. Given a query image, an ideal CBIR approach retrieves the images of its RRSet ahead of any other image within the reference collection. We selected the RRSets using a technique similar to the *pooling method* adopted in TREC conferences [17, Ch. 3], which is detailed next.

We extract the RRset for a given query from a pool of possible relevant images. This pool consists of the top 30 images retrieved by each compared CBIR approach. The pool of candidate images is visually analyzed to ultimately decide on the relevance of each image. The subset of relevant images in the pool is the RRSet of the query image. The decision about the relevance of a given image was based on its visual properties, its domain properties and its semantics. An example of one RRSet is shown in Figure 2.

In our experiments, we adopted a total of 11 different measures of retrieval effectiveness. We used two graphical measures (Precision vs. Recall and θ vs. Recall), and nine single value measures ($p(R)$, $p(30)$, $r(30)$, $p(100)$, $r(100)$, *3P-Precision*, and *11P-Precision*). Each of these measures evaluates a different aspect of the retrieval algorithm, and their combination gives a clear characterization of effectiveness according to several distinct criteria. Next, these retrieval effectiveness measures are discussed in details.

Precision vs. Recall ($P \times R$) curves [17] are well-known and widely used to evaluate retrieval effectiveness. Precision is defined as the fraction of the retrieved images that are relevant to the query. In contrast, recall measures the proportion of relevant images among the retrieved images. As recall is a non-decreasing function of rank, precision can be regarded as a function of recall rather than of rank. In general, the curve closest to the top of the chart indicates the best performance.

A variation of the $P \times R$ curve we propose is the θ vs. Recall curve ($\theta \times R$). We define θ as the average of the precision values measured whenever a relevant image is retrieved. For 100% of recall, the θ value is equivalent to the average precision used in [3]. The main difference between θ and precision

[2] Corel GALLERY Magic 65,000 - Stock Photo Library 2.

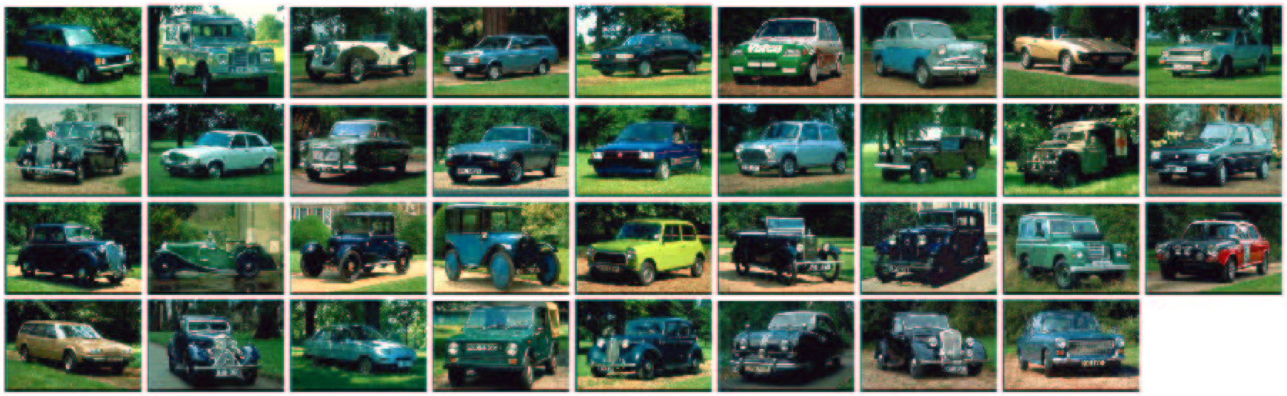

Figure 2: A sample RRSet. (All 50 query images used in our experiments and the corresponding RRSets can be viewed in color at: `http://db.cs.ualberta.ca/mn/BIC/queries.html`.)

is that, unlike precision, the θ value is accumulative, i.e., its computation considers not only the precision at a specific recall level but also the precision at previous recall levels. This accumulative computation is more consistent with the ranking imposed by CBIR algorithms. While precision rely on a simple binary property of the retrieved images (relevant or not), the θ value takes into account additionally the ordering of the retrieved images in its computation.

Some researchers believe that a retrieval effectiveness measure should be expressible as a single number that can be put on a scale to give absolute and relative values. One of such possibility is to measure the precision when the number of retrieved images is just sufficient to include all the relevant images for a query. This value is known as *R-value* [17], and we call the precision at this point $p(R)$. We also measure the values $p(30)$, $r(30)$, $p(100)$ and $r(100)$. The first two measures correspond to the precision and the recall after 30 images are retrieved. The choice of the value 30 was based on the fact that it corresponds to the retrieval cutoff point used to determine the RRSets of the query images, as discussed at the beginning of this section. Similarly, we compute the precision and the recall after 100 images are retrieved. This value is an estimate of the number of retrieved images an average user would accept to inspect in order to determine their relevance to his/her needs. Finally, the two other single value measures are the 3-point and the 11-point average precision [17]. The 3-point average precision (*3P-Precision*) is computed by averaging the precision taking at three predefined recall levels, typically, 20%, 50% and 80%. The 11-point average precision (*11P-Precision*) is computed by averaging the precision taking at eleven predefined recall levels: 0%, 10%, ... , 90%, 100%.

5. EXPERIMENTAL RESULTS

This section discusses our experimental results relative to the effectiveness of the proposed *BIC* approach. Initially, we compare *BIC* and the existing CBIR approaches reviewed in Section 2, showing that *BIC* outperforms all of them. After that, we evaluate the effectiveness of the *dLog* distance function when used with other histogram-based approaches, and show that it indeed improves the effectiveness of all investigated approaches. We conclude showing that *BIC* still prevails, outperforming all *dLog*-improved approaches.

In Figure 3 and in the first lines of Table 1, we compare *BIC* [3] with the CBIR approaches reviewed in Section 2. The results of the eleven measures confirm the general belief that partition-based approaches are more effective than global approaches (Grid9 is better than GCH), and that regional CBIR approaches are more effective than partition-based approaches (CBC is better than Grid9). The comparison of CCV and GCH reveals that the pixel classification of CCV becomes effective only after 20% of recall. However, the gain in terms of effectiveness obtained with CCV approach is not very expressive, especially if one considers its storage overhead. More important, however, is the fact that the proposed *BIC* approach is clearly more effective than all investigated CBIR approaches, including CBC.

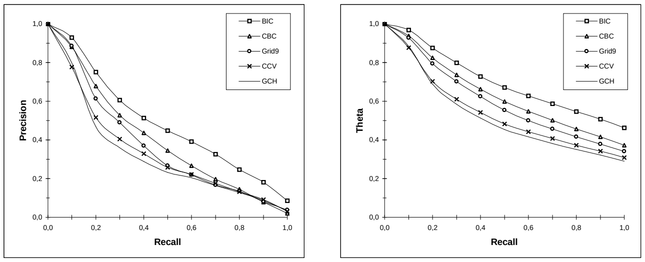

Figure 3: *BIC* versus existing approaches

Besides being more effective than CBC, the *BIC* approach is also more compact and efficient. The *BIC* approach is based on a very simple (but powerful) image analysis algorithm that runs in time $O(n)$, where n is the size (in pixels) of the image being analyzed. Moreover, as discussed in Section 3.3, the *BIC* visual features can be stored in just 64 bytes of memory, and the comparison of these visual features is based on the very efficient and effective *dLog* distance function. The *dLog* distance function is several orders of magnitude more efficient than the MiCRoM metric adopted in CBC approach. While a visual query in our reference collection of images takes only a small fraction of a second using the *BIC* approach, in CBC this same visual query takes several minutes to be processed.

The second part of our experiments evaluated the effect of

[3] The top 30 images retrieved by the *BIC* approach for all 50 query images used in our experiments can be viewed in color at: `http://db.cs.ualberta.ca/mn/BIC/bic30.html`

Table 1: Single-value effectiveness results

Approach	3P-Precision	11P-Precision	p(30)	r(30)	p(100)	r(100)	p(R)
BIC	0.48	0.50	0.46	0.44	0.22	0.70	0.44
L_1 *BIC*	0.35	0.39	0.35	0.33	0.19	0.57	0.34
GCH	0.28	0.34	0.30	0.30	0.17	0.52	0.31
CCV	0.30	0.36	0.33	0.32	0.17	0.52	0.32
Grid9	0.34	0.39	0.35	0.35	0.17	0.56	0.34
dLog GCH	0.38	0.43	0.39	0.37	0.20	0.64	0.39
dLog CCV	0.41	0.44	0.42	0.40	0.20	0.63	0.40
dLog Grid9	0.40	0.43	0.40	0.40	0.19	0.61	0.39
CBC	0.39	0.42	0.40	0.39	0.18	0.58	0.39

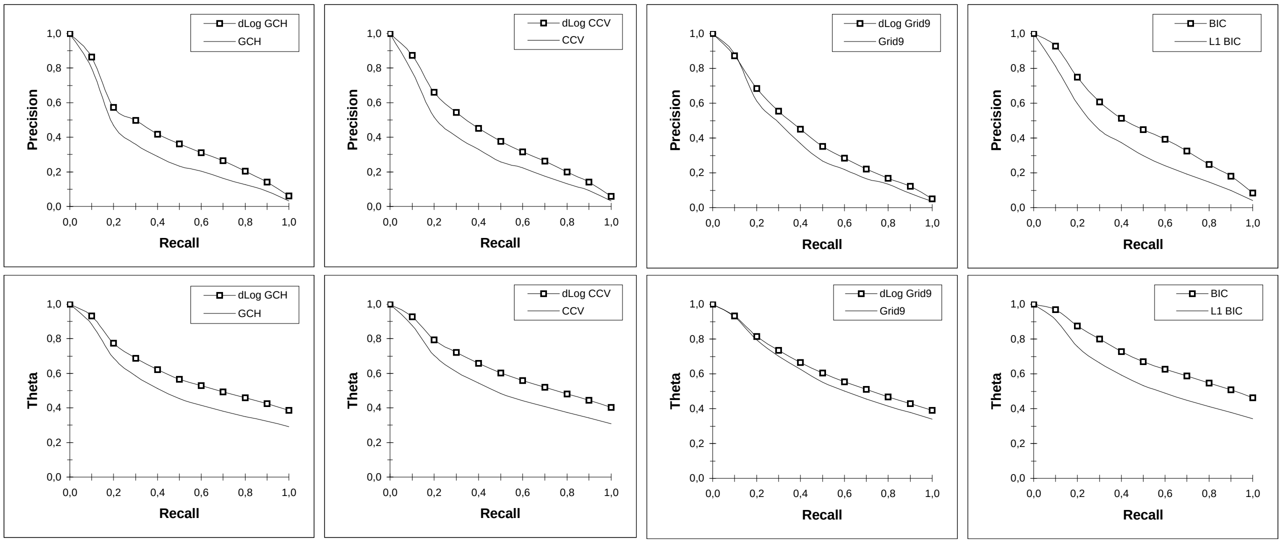

Figure 4: Effectiveness results of the *dLog* distance function

using the *dLog* distance instead of L_1 in existing histogram-based approaches. The effectiveness results of this experiment, which are also supported by the many measures used in Table 1, can be observed in Figure 4. In that figure, each column is related to a CBIR approach. We have plotted two graphs per approach, comparing its original effectiveness (using L_1 distance) with the effectiveness when *dLog* is used instead of L_1. An exception is the last column where we show how the use of L_1 would adversely affect *BIC* (recall that *dLog* is the "native" distance designed for *BIC*). The top row shows $P \times R$ graphs while the bottom row shows the $\theta \times R$ graphs.

Observing Figure 4 and Table 1, one can conclude that the *dLog* distance function clearly increases the effectiveness of all histogram-based approaches tested. This increase in effectiveness is more accentuated in GCH and CCV than in Grid9. We have observed that, when the *dLog* function is used, the spatial information of Grid9 becomes less important as it is unable to make the *dLog* Grid9 more effective than *dLog* CCV. We have observed a similar behavior also in the context of the proposed *BIC* approach. We have tried several ways to add spatial information into the *BIC* visual features. However, none of these attempts were successful as they were unable to increase the effectiveness of the *BIC* approach as it was proposed. We explain this behavior in the following way. When the comparison of the visual features is based on less effective distances like L_1, the approaches are able to retrieve only a small fraction of the relevant images for a given query image in the top T retrieved images, where T is a retrieval threshold. In this context, the addition of spatial information is useful because it adds to the set of retrieved images relevant ones with similar spatial distribution of colors (that were not originally retrieved). However, if the visual features are compared using more robust and effective distances like the *dLog* distance, the approaches are able to retrieve most of the relevant images for a given query. In this context, if we add restrictions about the spatial distribution of colors, we not only do not include more relevant images to the set of retrieved images (relevant images with similar spatial layout were already retrieved) but, in fact, we eliminate from the set of retrieved images those relevant images that are not similar to the query image in terms of spatial layout of colors.

Finally, observing Figure 5, again supported by Table 1, we can conclude that the *BIC* approach is clearly more effective than any of the *dLog*-improved histogram-based approaches, including *dLog* CCV. As the *dLog* CCV uses the same representation and distance function used in *BIC*, we can conclude that this gain in effectiveness is due solely to the *BIC* image analysis algorithm. As discussed in Section 3.1, the binary classification of image pixels in border/interior adopted in *BIC* is more robust and effective than the classification adopted in CCV, that makes a binary classification of pixels based on a non-binary image property – the size of the connected regions.

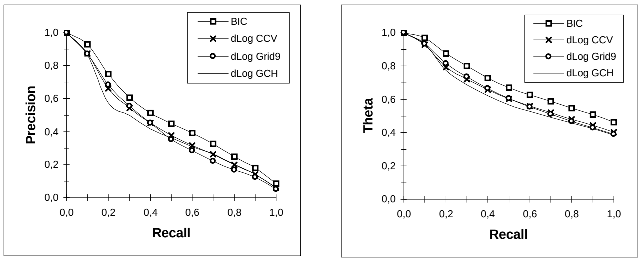

Figure 5: *BIC* versus the *dLog*-improved approaches

6. CONCLUSIONS AND FUTURE WORK

This paper presented *BIC* (*B*order/*I*nterior pixel *C*lassification), a compact and efficient CBIR approach for broad image domains. The *BIC* approach has three main components: (1) a simple and powerful image analysis algorithm that classifies image pixels as border or interior; (2) a new logarithmic distance to compare histograms; (3) a compact representation for the visual features extracted from images.

The *BIC* image analysis algorithm classifies image pixels in either border or interior. Our experimental results show that the *BIC* approach is consistently more effective than state-of-the-art regional CBIR approaches based on more sophisticated image analysis algorithms. As well, unlike those other approaches it also does not require computationally expensive post-processing simplification steps, which contributes to strenghten *BIC*'s advantage.

The second component of the *BIC* approach is the *dLog* metric distance function. This function compares two histograms according to a log scale, diminishing distortions in the measured distance generated by histogram bins with very high values. As our experimental results show, the use of the *dLog* function has two major advantages over vectorial distances like L_1. First, the *dLog* function clearly increases the effectiveness of any histogram-based CBIR approach. Second, the use of this function allows a log-based representation for the histograms that makes possible to store a histogram bin in just 4 bits of memory. This log-based representation reduces the space required to store a histogram in any histogram-based CBIR approach. In the particular case of *BIC* approach, each *BIC* histogram has 128 bins. Thus, it is possible to store a *BIC* histogram in just 64 bytes of memory. This is a very compact representation for the visual features of an image.

Our future work includes the investigation of other properties of the image pixels that can be exploited during the *BIC* image analysis algorithm in order to have a more precise description of the image's visual content. We also plan to investigate the use of relevance feedback to reduce the semantic gap between the low-level visual features automatically extracted from images, and the human interpretation of the image's visual content. Lastly, even though we argue that the use of disk-based access structures could be avoided for relatively large image datasets, we believe that the scalability of any CBIR approach is paramount. Hence we should also investigate the use of access structures such as the Signature Tree (e.g., [9]) to speedup query processing.

7. REFERENCES

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, 1993.

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

[3] C. Buckley and E.M. Voorhees. Evaluating evaluation measure stability. In *Proc. of the ACM SIGIR Intl. Conf.*, pages 33–40, 2000.

[4] C.H.C. Leung and H.H.S. Ip. Benchmarking for content-based visual information search. In *Proc.of VISUAL Intl. Conf.*, pages 442–456, 2000.

[5] J. Li, J.Z. Wang, and G. Wiederhold. IRM: Integrated region matching for image retrieval. In *Proc.of ACM Multimedia Intl. Conf.*, pages 147–156, 2000.

[6] G. Lu. *Multimedia Database Management Systems.* Artech House, 1999.

[7] H. Muller et al. Performance evaluation in content-based image retrieval: Overview and proposals. *Pattern Recognition Letters*, 22(5):593–601, 2001.

[8] M.A. Nascimento and V. Chitkara. Color-based image retrieval using binary signatures. In *Proc. of ACM SAC Intl. Conf.*, pages 687–692, 2002.

[9] M.A. Nascimento et al. Image indexing and retrieval using signature trees. *Data and Knowledge Engineering Journal*, 43(1):57–77, 2002.

[10] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In *Proc. of ACM Multimedia Intl. Conf.*, pages 65–73, 1996.

[11] N. Sebe, M.S. Lew, and D.P. Huijsmans. Multi-scale sub-image search. In *Proc. of ACM Multimedia Intl. Conf.*, pages 79–82, 1999.

[12] A.W.M. Smeulders et al. Content-based image retrieval at the end of the early years. *IEEE PAMI*, 22(12):1349–1380, 2000.

[13] R.O. Stehling, M.A. Nascimento, and A.X. Falcão. An adaptive and efficient clustering-based approach for content based retrieval in image databases. In *Proc. of IDEAS Intl. Symposium*, pages 356–365, 2001.

[14] R.O. Stehling, M.A. Nascimento, and A.X. Falcão. MiCRoM: A metric distance to compare segmented images. In *Proc. of VISUAL Intl. Conf.*, pages 12–23, 2002.

[15] R.O. Stehling, M.A. Nascimento, and A.X. Falcão. Techniques for color-based image retrieval. In *Multimedia Mining - a Highway to Intelligent Multimedia Document*, Chapter 4. Kluwer Academic Publishers, 2002.

[16] R.O. Stehling, M.A. Nascimento, and A.X. Falcão. Cell histograms versus color histograms for image representation and retrieval. *Knowledge and Information Systems Intl. Journal*, 2003. To appear.

[17] R. Baeza-Yates and B. Ribeiro Neto. *Modern Information Retrieval.* Addison Wesley, 1999.