# SIFT applied to CBIR

Jurandy Almeida, Ricardo da S. Torres and Siome Goldenstein
Institute of Computing, University of Campinas
13083–970, Campinas, SP – Brazil
Email: {jurandy.almeida, rtorres, siome}@ic.unicamp.br

***Abstract*—Content-Based Image Retrieval (CBIR) is a challenging task. Common approaches use only low-level features. Notwithstanding, such CBIR solutions fail on capturing some local features representing the details and nuances of scenes. Many techniques in image processing and computer vision can capture these scene semantics. Among them, the Scale Invariant Features Transform (SIFT) has been widely used in a lot of applications. This approach relies on the choice of several parameters which directly impact its effectiveness when applied to retrieve images. In this paper, we discuss the results obtained in several experiments proposed to evaluate the application of the SIFT in CBIR tasks.**

***Index Terms*—Scale Invariant Features Transform (SIFT), Content-Based Image Retrieval (CBIR).**

## I. Introduction

Advances in data storage, data transmission, and image acquisition have enabled the creation of large image datasets. It has spurred great interest for systems that are able to efficiently retrieve images from these collections.

This task has been addressed by the so-called *Content-Based Image Retrieval* (CBIR) systems [1]. In these systems, image content is represented by their low-level features, such as color, shape, and texture [2], [3]. It requires the construction of an image descriptor, which is characterized by: (1) an extraction algorithm to encode image features into feature vectors, and (2) a similarity measure to compare two images [4].

There are some quite powerful image descriptors designed to represent global features of images [5]–[10]. These approaches have been widely used in image retrieval due to their usually low computational costs and acceptable effectiveness. Notwithstanding, such CBIR solutions fail on capturing some local features representing the details and nuances of the scenes [11].

These details can be obtained by mapping low-level features into middle- and high-level semantics [12]. Many techniques in image processing and computer vision can encode these semantics [13]–[16]. Among them, the Scale Invariant Features Transform (SIFT) [13] has been widely used in a lot of applications, such as object recognition [17], recognizing panoramas [18], and tridimensional reconstruction [19].

This approach relies on the choice of several parameters which directly impact its effectiveness when applied to retrieve images. In this sense, this work discuss the results obtained in several experiments proposed to evaluate the application of the SIFT in CBIR tasks. These experiments were conducted to analyze the most relevant characteristics of this technique for image retrieval.

In special, our experiments address the following research questions:

1) Is the SIFT approach color invariant?
2) What is the suitable feature-vector size?
3) How effective is the SIFT-based image description approach?

The remainder of this work is organized as follows. Section II introduces some basic CBIR concepts. Section III presents the SIFT approach. The experimental results obtained from the application of this technique for image retrieval are discussed in Section IV. Finally, Section V offers conclusions and directions for future work.

## II. Image descriptors

A typical CBIR solution requires the construction of an image descriptor, which is characterized by: (1) an extraction algorithm to encode image features into feature vectors, and (2) a similarity measure to compare two images. The similarity measure is a matching function, which gives the degree of similarity for a given pair of images as represented by their feature vectors [4].

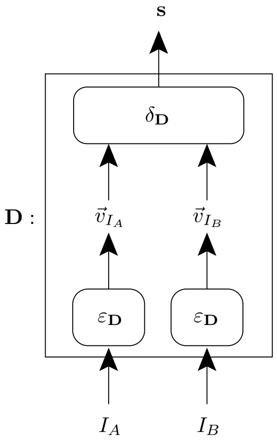

Fig. 1. The use of a simple descriptor for computing the similarity between two images.

Formally, a feature vector $\vec{v}_I$ of an image $I$ can be thought as a point in $\Re^n$ space: $\vec{v}_I = (v_1, v_2, \ldots, v_n)$, where $n$ is the dimension of the vector. They essentially encode image

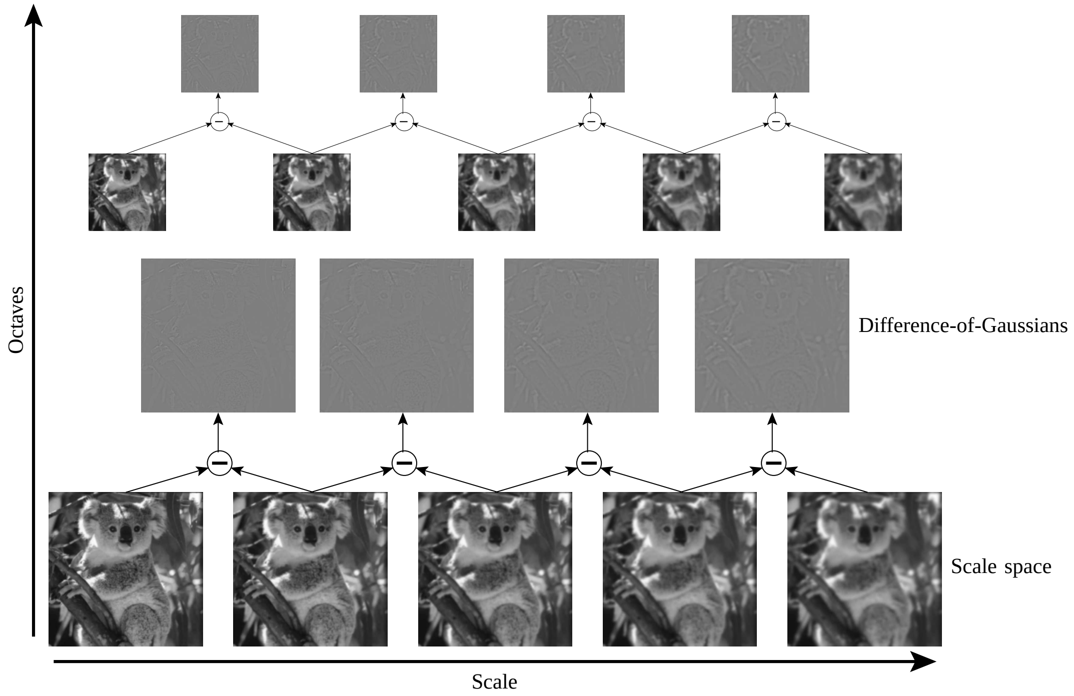

Fig. 2. The construction of scale space extrema based on difference-of-Gaussians.

properties, such as color, shape, and texture. Note that different types of feature vectors may require different similarity functions [4].

A simple image descriptor $\mathbf{D}$ is defined as a tuple $(\varepsilon_{\mathbf{D}}, \delta_{\mathbf{D}})$, where [4]:

- $\varepsilon_{\mathbf{D}} : I \rightarrow \Re^n$ is a function, which extracts a feature vector $\vec{v}_I$ from an image $I$.
- $\delta_{\mathbf{D}} : \Re^n \times \Re^n \rightarrow \Re$ is a similarity function that computes the similarity between two images.

Figure 1 illustrates the use of a simple descriptor $\mathbf{D}$ to compute the similarity between two images $I_A$ and $I_B$. First, the extraction algorithm $\varepsilon_{\mathbf{D}}$ is used to compute the feature vectors $\vec{v}_{I_A}$ and $\vec{v}_{I_B}$ associated with the images. Next, the similarity function $\delta_{\mathbf{D}}$ is used to determine the similarity value $\mathbf{s}$ between $I_A$ and $I_B$. Eventually, multiple descriptors can be combined into a complex descriptor, which is able to encode multiple image properties at a same time [20].

## III. The SIFT approach

Lowe [13] has presented a powerful framework to recognize/retrieve objects: the *Scale Invariant Features Transform* (SIFT). This approach can be viewed as a texture descriptor composed by four major stages [13]:

1) scale-space extrema detection;
2) keypoint localization;
3) orientation assignment;
4) keypoint description.

In the following, we describe each one of these steps.

### A. Detection of scale-space extrema

In the first stage, the method identifies locations and scales that can be repeatably assigned under differing views of the same object. Detecting locations that are invariant to scale change of the image can be accomplished by searching for stable features across all possible scales, using a continuous function of scale known as scale space [13].

The scale space of an image $I(x, y)$ is defined as a function $L(x, y, \sigma)$, that is produced from the convolution of $I(x, y)$ with a variable-scale Gaussian $G(x, y, \sigma)$ [13]:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \tag{1}$$

where $*$ is the convolution operation in $x$ and $y$, and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \tag{2}$$

To efficiently detect stable keypoint locations in scale space, it is used a scale space extrema based on the difference-of-Gaussian function, $D(x, y, \sigma)$, which can be computed from the difference of two nearby scales separated by a constant multiplicative factor $k$ [13]:

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned} \tag{3}$$

Figure 2 shows an efficient approach to construction of $D(x, y, \sigma)$. The initial image is incrementally convolved with Gaussians to produce images separated by a constant factor $k$ in scale space, shown stacked on the bottom. The scale space is divided into octaves (e.g., doubling $\sigma$) with an integer number, $s$, of intervals, so $k = 2^{1/s}$. Adjacent image scales are subtracted to produce the difference-of-Gaussian images shown on the top. Next, we resample the Gaussian image that has twice the initial value of $\sigma$ by taking every second pixel in each row and column, and recursively iterate all the process.

### *B. Local extrema detection*

In order to detect the local maxima and minima of $D(x, y, \sigma)$, each sample point is compared to its eight neighbors in the current image and nine neighbors in the scale above and below (see Figure 3). It is selected only if it is larger than all of these neighbors or smaller than all of them [13].

Once a keypoint candidate has been found, the next step is to adjust its accuracy. It is performed by a Taylor expansion of the scale-space function, $D(x, y, \sigma)$, shifted so that the origin is at the sample point [13]:

$$D(X) = D + \frac{\partial D^T}{\partial X} X + \frac{1}{2} X^T \frac{\partial^2 D}{\partial X^2} X, \qquad (4)$$

where $D$ and its derivatives are evaluated at the sample point and $X = (x, y, \sigma)^T$ is the offset from this point. The location of the extremum, $\hat{X}$, is determined by taking the derivative of Eq. 4 with respect to $X$ and setting it to zero, giving

$$\hat{X} = \frac{\partial^2 D^{-1}}{\partial X^2} \frac{\partial D}{\partial X}. \qquad (5)$$

The function value at the extremum, $D(\hat{X})$, is useful for rejecting unstable extrema with low contrast. It is giving by [13]:

$$D(\hat{X}) = D + \frac{1}{2} \frac{\partial D}{\partial X} \hat{X}. \qquad (6)$$

The difference-of-Gaussian function will have a strong response along edges, even if the location along the edge is poorly determined and therefore unstable to small amounts of noise. It defines peaks in the difference-of-Gaussian function, which have a large principal curvature across the edge but a small one in the perpendicular direction [13].

The principal curvatures can be computed from a Hessian matrix, $H$, obtained by taking differences of neighboring sample points in $D$ at the location and scale of the keypoint [13]:

$$H = \left[ \begin{array}{cc} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{array} \right]. \qquad (7)$$

The eigenvalues of $H$ are proportional to the principal curvatures of $D$. Let $\alpha$ be the eigenvalue with the largest magnitude and $\beta$ be the smaller one. Then, we can compute the sum of the eigenvalues from the trace of $H$ and their product from the determinant [13]:

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta, \qquad (8)$$

$$Det(H) = D_{xx} D_{yy} - (D_{xy})^2 = \alpha\beta. \qquad (9)$$

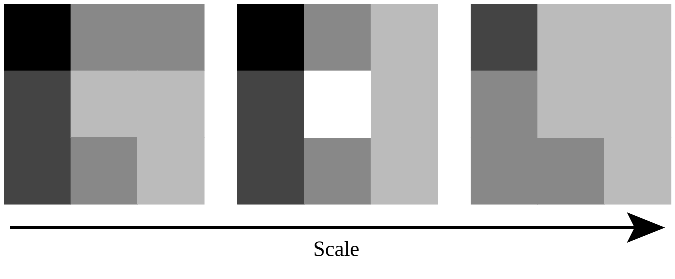

Fig. 3. Keypoint localization at different scales.

Let $r$ be the ratio between the largest magnitude eigenvalue and the smaller one, so that $\alpha = r\beta$. Then,

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r + 1)^2}{r}. \qquad (10)$$

Therefore, to check that the ratio of principal curvatures is below some threshold, $r$, we only need to check

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r + 1)^2}{r}. \qquad (11)$$

### *C. Orientation assignment*

By assigning a consistent orientation to each keypoint based on local image properties, its feature vector can be represented relative to this orientation and therefore achieve invariance to image rotation [13].

The scale of the keypoint is used to select the Gaussian smoothed image, $L$, with the closest scale, so that all computations are performed in a scale-invariant manner. For each image sample, $L(x, y)$, at this scale, the gradient magnitude, $m(x, y)$, and orientation, $\theta(x, y)$, is precomputed using pixel differences [13]:

$$\frac{\partial_L}{\partial_x}(x, y) = L(x + 1, y) - L(x - 1, y) \qquad (12)$$

$$\frac{\partial_L}{\partial_y}(x, y) = L(x, y + 1) - L(x, y - 1) \qquad (13)$$

$$m(x, y) = \sqrt{\frac{\partial_L}{\partial_x}(x, y)^2 + \frac{\partial_L}{\partial_y}(x, y)^2} \qquad (14)$$

$$\theta(x, y) = \tan^{-1} \left( \frac{\frac{\partial_L}{\partial_y}(x, y)}{\frac{\partial_L}{\partial_x}(x, y)} \right) \qquad (15)$$

Next, an orientation histogram is formed from the gradient orientations of sample points within a region around the keypoint. The orientation histogram has $36$ bins covering $360$ degree range of orientations. Each sample added to the histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with a $\sigma$ that is $1.5$ times the scale of the keypoint [13].

Peaks in the orientation histogram correspond to dominant directions of local gradients. The highest peak is detected, and then any other local peak that is within $80\%$ of the highest peak is used to also create a keypoint with that orientation. Therefore, for locations with multiple peaks of similar magnitude, there will be multiple keypoints created at the same location and scale but different orientations [13].

### *D. The local-feature vector*

Figure 4 illustrates the computation of the feature vector of each keypoint. First, the image gradient magnitudes and orientations are sampled around the location of the keypoint using its scale for selecting the level of Gaussian blur. In order to achieve orientation invariance, the coordinates of the feature vector and the gradient orientations are rotated by taking into account the keypoint orientation [13].

A Gaussian weighting function with $\sigma$ equal to one half the width of the feature-vector window is used to assign a weight

Scale

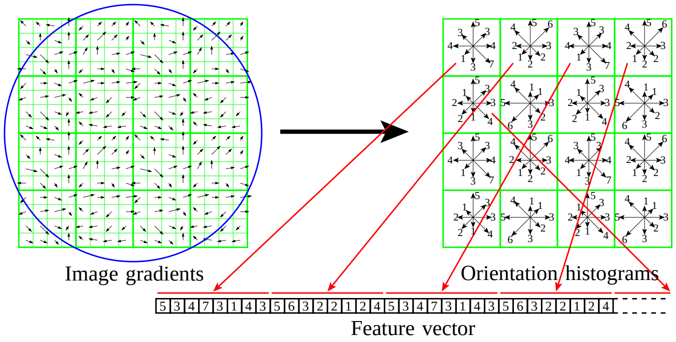

Fig. 4. The computation of the feature vector of a keypoint.

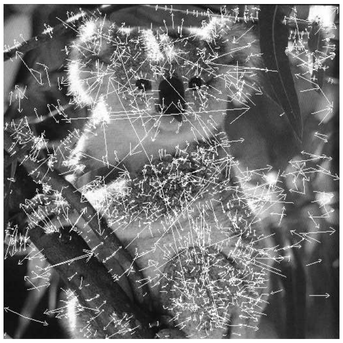

Fig. 5. Example of an image and its keypoints.

to the magnitude of each sample point, as showed on the left side of Figure 4. The purpose of this Gaussian window is to avoid sudden changes in the feature vector, and to give more emphasis to gradients that are close from its center [13].

The feature vector is shown on the right side of Figure 4. It allows for significant shift in gradient positions by creating orientation histograms over $4 \times 4$ sample regions. The figure shows eight directions for each orientation histogram, with the length of each arrow corresponding to the magnitude of that histogram entry [13].

In order to avoid all boundary affects in which the feature vector abruptly changes as a sample shifts smoothly from being within one histogram to another or from one orientation to another, each entry into a bin is multiplied by a weight of $1-d$ for each dimension, where $d$ is the distance of the sample from the central value of the bin as measured in units of the histogram bin spacing [13].

The feature vector is formed from a vector containing the values of all the orientation histogram entries, corresponding to the lenghts of the arrows, as shown at the bottom of Figure 4. Unless when otherwise stated, the experiments of this paper use a $4 \times 4 \times 8 = 128$ element feature vector for each keypoint.

Finally, the feature vector is modified to reduce the effects of illumination change. First, the vector is normalized to unit length. Next, we reduce the influence of large gradient magnitudes by thresholding each value in the unit feature vector to be no larger than $0.2$, and then renormalizing to unit length, giving more emphasis to the distribution of orientations [13].

Figure 5 illustrates the resulting keypoints detected by the application of this approach in an image. The white arrows show their location, scale, and orientation.

### *E. Distance function*

The number of keypoints obtained for each image can be different. The more complex an image, the more keypoints SIFT provides. Therefore, we need to compare images with a different number of keypoints. Hence, we model each feature vector as a hyper point under an unknown distribution. Further, we use the Earth Mover's Distance (EMD) metric [21] to evaluate the dissimilarity between two multi-dimensional distributions (keypoints). The advantage is that EMD "lifts" the distance from individual features to full distributions.

Intuitively, given two distributions $\mathbf{B_p}$ and $\mathbf{B_q}$, we can view $\mathbf{B_p}$ as a mass of earth properly spread in space, and $\mathbf{B_q}$ as a collection of holes in that same space. The EMD measures the least amount of work needed to fill the holes with earth. Here, a unit of work corresponds to transporting a unit of earth by a unit of ground distance [21].

EMD provides a way to compare images based on their discrete distributions of local features. Let $(\mathcal{X}, \mathcal{D})$ be a metric space, $\mathbf{B_p}, \mathbf{B_q} \subset \mathcal{X}$ be two equal-mass sets, and $\pi$ be a matching between $\mathbf{B_p}$ and $\mathbf{B_q}$. The EMD is the minimum possible cost of $\pi$ [21]:

$$EMD(\mathbf{B_p}, \mathbf{B_q}) = \min_{\pi:\mathbf{B_p} \rightarrow \mathbf{B_q}} \sum_{s \in \mathbf{B_p}} \mathcal{D}(s, \pi(s)). \tag{16}$$

The computation of $\mathcal{D}$ is based on establishing the correspondence between two images' unordered local features. However, the complexity of finding the optimal correspondence between two equal-mass sets is cubic in the number of features per set [22]. Hence, we use a low-distortion EMD embedding [22], [23] to reduce the problem of correspondence between sets of local features to an $L_1$ distance. In this approach, we use an approximation function $h$ to map the EMD distance into one $L_1$ distance with low distortion, such that

$$\frac{1}{C} EMD(B_p, B_q) \leq \|h(B_p) - h(B_q)\|_{L_1} \leq EMD(B_p, B_q), \tag{17}$$

where $C$ is the distortion factor.

We perform that mapping by using grids $G_j \in \Re^n$, $-1 \leq j \leq \log \Delta$, where $G_j$ is composed by $2^j$-sized square cells and $\Delta$ is the diameter of $B_p \cup B_q$. We translate each grid $G_j$ by a random vector $\vec{v}_j \in [0, \Delta]^n$. In order to embed a point set $B_p$ into the grid, we create a vector $\vec{v}_j \in \Re^n$ for each grid $G_j$ with one coordinate per grid cell. Each coordinate counts the number of points in its corresponding cell. Roughly speaking, each $\vec{v}_j$ forms a histogram of $B_p$ [22].

The mapping $h$ is given by the concatenation of the vectors $\vec{v}_j$ scaled by the size of their corresponding grid

$$h(B_p) = [\vec{v}_{-1}(B_p), \vec{v}_0(B_p), 2\vec{v}_1(B_p), \ldots 2^j \vec{v}_j(B_p)]. \tag{18}$$

After the embedding of $B_p$ and $B_q$, the distance between them is given by

$$d(B_p, B_q) = \|h(B_p) - h(B_q)\|_{L_1}. \tag{19}$$

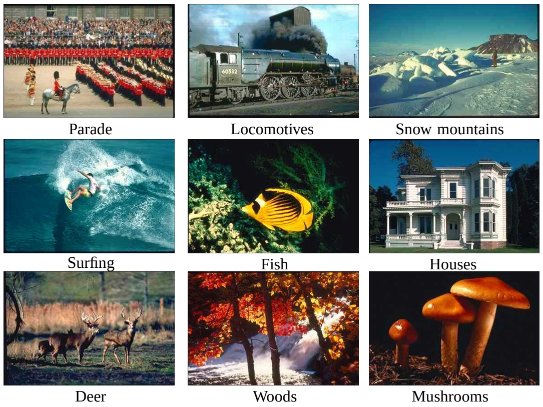

Fig. 6. Some categories of the *RRSets* database.

## IV. Experiments and results

In this work, we have adopted the widely used query-by-example (QBE) paradigm [24], as it seems to be the most adequate way to submit queries in CBIR systems based on visual features. In QBE, an image is given as a visual example of the information needed. This image is analyzed and its visual features are extracted. These features are used to measure the similarity between the query image and the images stored in an image database. The stored images are retrieved in decreasing order of their similarity to the query image (similarity-search).

The purpose of our experiments is to evaluate the effectiveness of the similarity-search of the SIFT approach in retrieving relevant images ahead of non-relevant ones. 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, and an adequate retrieval effectiveness measure.

We use a heterogeneous collection of 1,624 images from Corel Photo Gallery[1] reported in [6]. This database contains 50 image categories and is referred as the Corel Relevant sets (*RRSets*). Here, our set of query images are equal to the reference collection, and we test all images in the database against the remaining images, one at a time. Figure 6 shows some categories of the *RRSets* database.

In our experiments, we use the well-known *Precision* × *Recall* curves [24] to assess the retrieval effectiveness. *Precision* is the ratio of the number of relevant images retrieved to the total number of irrelevant and relevant images retrieved. *Recall* is the ratio of the number of relevant images retrieved to the total number of relevant images in the database. In general, the closest curve to the top of the chart indicates the best performance.

In the following, we discuss the results obtained in several experiments proposed to evaluate the application of the SIFT in CBIR tasks. The goal of these experiments is to analyze the most relevant characteristics of this technique for image retrieval, such as: (1) the color invariance; (2) the feature-vector size; and (3) the effectiveness of the SIFT-based image description approaches.

### A. Color channels

The SIFT approach encodes no color information. Hence, we performed the first experiments to analyze which color channel is more relevant to retrieve images using this technique. Initially, we evaluated the use of a single color channel: (1) luma (Y of YCbCr), and (2) brightness (V of HSV). Figure 7 shows the results obtained in these experiments.

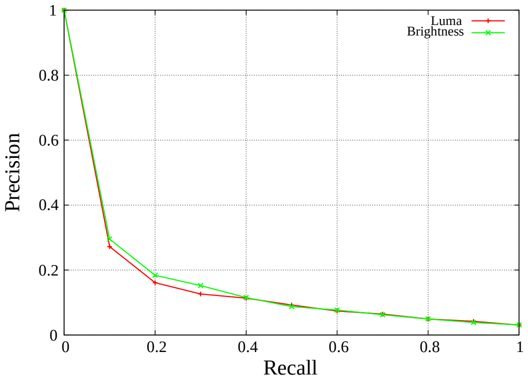

Fig. 7. Retrieval effectiveness of the SIFT approach using a single color channel.

Note that the results are close for both color channels. It happens because the SIFT approach is invariant to color channels. In order to prove it, we performed an experiment grouping keypoints extracted from multiple color channels. In Figure 8, we show the retrieval effectiveness of this technique using all color channels of the RGB color-space.

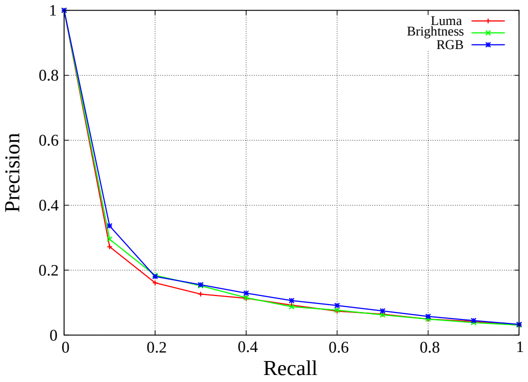

Fig. 8. Retrieval effectiveness of the SIFT approach using multiple color channels.

In fact, the retrieval effectiveness is not affected by different color channels. The SIFT approach encodes the image gradient magnitudes and orientations, which are preserved in all color channels. In the following experiments, we use the brightness as default information.

### B. Size of the feature-vector window

Each keypoint codifies the image gradient magnitudes and orientations of a region sampled around its location. The size of this region is four times the size of the array of orientation histograms that comprises the feature vector.

[1] http://www.cs.ualberta.ca/∼mn/BIC/queries.html

Therefore, we can vary the size of the region analyzed around each keypoint by controling the size of the feature-vector window. Figure 9 shows how the retrieval effectiveness of the SIFT approach is affected by the size of that region.

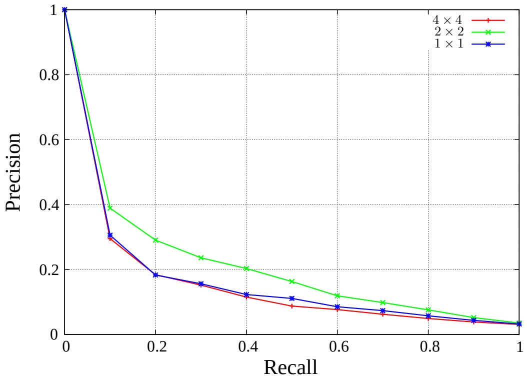

Fig. 9. Retrieval effectiveness of the SIFT approach varying the size of the feature-vector window.

It is important to realize that the retrieval effectiveness improves when we reduce the size of the feature-vector window from $4 \times 4$ to $2 \times 2$. However, the results worsen varying this size from $2 \times 2$ to $1 \times 1$. It occurs because the description quality decreases when we codify a very small region around each keypoint. On the other hand, the larger the size of the feature-vector window, the noisier each keypoint encodes.

### C. Feature-vector dimesionality

Each feature vector is formed from a vector containing the values of all orientation histogram entries. We can reduce the feature-vector dimensionality by reducing the size of the feature-vector window.

However, the smaller the size of the feature-vector window, the worse the description quality of the keypoint. In this sense, we performed an experiment varying the feature-vector dimensionality, but keeping the description quality of the keypoint as better as possible. For this, we use a well-known statistic, called *Principal Components Analysis* (PCA) [25].

In Figure 10, we show how the retrieval effectiveness of the SIFT approach is affected by applying the PCA approach to reduce the feature-vector dimensionality from $4 \times 4 \times 8 = 128$ to 96, 64, 32, 16, 8, and 4; but keeping the size of the feature-vector window ($4 \times 4$).

In general, the smaller the feature-vector dimensionality, the better the retrieval effectiveness. However, there is a trade-off between the description quality of the keypoint and the number of elements in the feature vector.

Ke and Sukthankar [14] have introduced the application of PCA for reducing the feature-vector dimensionality of the SIFT approach. However, they performed all experiments in object recognition/retrieval tasks and used an one-to-one keypoint matching. Here, we extend their results for image retrieval tasks and use a many-to-many keypoint matching.

### D. Relevance degree of the features

Figures 11 and 12 show the top-8 images retrieved by queries that achieved the best and the worst results, respectively, using a feature vector with 128 elements. We show the query image on left and its resulting retrieved images sorted from left to right.

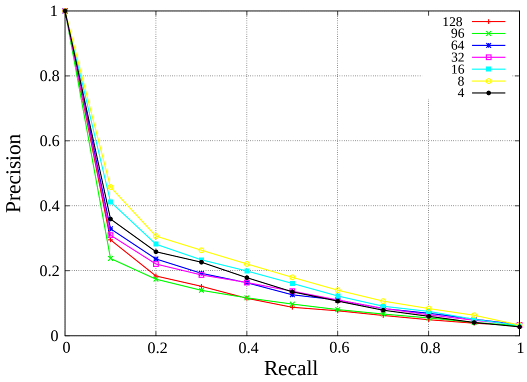

Fig. 10. Retrieval effectiveness of the SIFT approach reducing the feature-vector dimensionality.

Note that all images in the query that achieved the best results have a single object on the same background (see Figure 11). Nevertheless, the oscillations of the water in the query image are confused with the re-entrances of the caves' stalactites in Figure 12. It happens because gradients have a strong response along edges, which are suitable for object recognition/retrieval.

The simplest descriptor to encode the color information is the Global Color Histogram[2] (GCH) [5]. A GCH is a set of ordered values, one for each distinct color, representing the probability of a pixel being of that color. The most commonly used GCH implementation relies on the RGB color-space uniformly quantized into $64$ distinct colors and the $L_1$ distance function to compare two histograms.

In Figure 13, we show the retrieval effectiveness of the SIFT approach replacing the array of orientation histograms that comprises the feature vector by a single color histogram. In fact, colors yield better results than gradients for image retrieval.

Abdel-Hakim and Farag [16] have augmented the SIFT approach based on grayscale images to become color invariant. However, they transform the input image into an invariant color-space prior to the description. Here, we extract a GCH from the region analyzed around each keypoint.

## V. Conclusions

In this paper, we have discussed the results obtained in several experiments conducted to evaluate the application of the SIFT in CBIR tasks. This approach relies on the choice of several parameters which directly impact the effectiveness when applied to retrieve images.

Our experiments showed that the SIFT approach is invariant to color channels. In addition, we have found that there is a trade-off between the size of the feature vector and its description quality in order to produce good results. Moreover, the use color information in the local-feature vector outperforms

[2] As in the literature, we define a histogram as a graphical representation of a distribution, which tabulates a data set into bins [12].

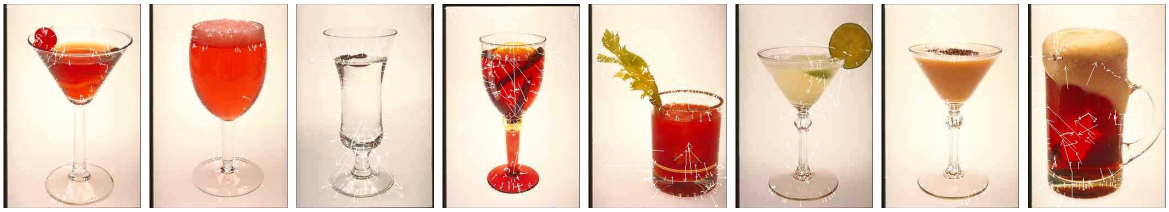

Fig. 11. Top-8 images retrieved by the query of the best results.

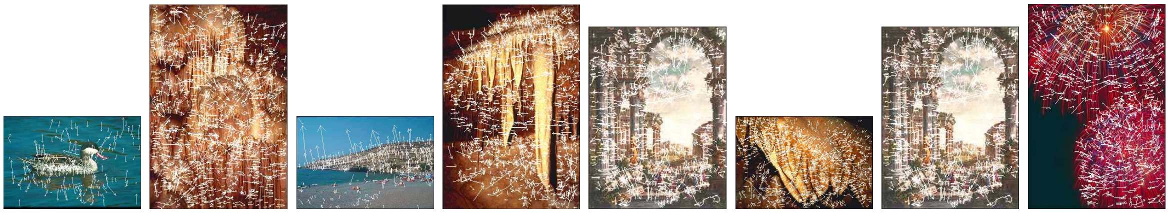

Fig. 12. Top-8 images retrieved by the query of the worst results.

the gradient-based histogram of traditional SIFT approach for image retrieval.

Future work includes the evaluation of other many-to-many feature matching approaches and low-level image descriptors to improve the image representation.

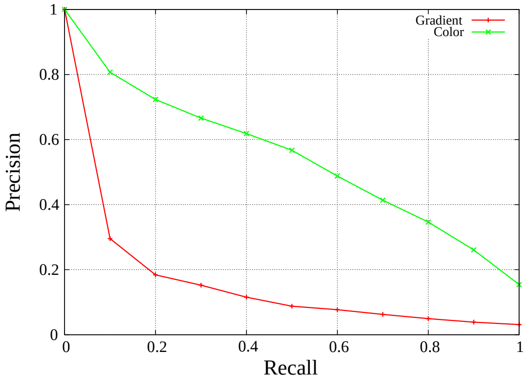

Fig. 13. Retrieval effectiveness of the SIFT approach replacing gradients by colors.

## Acknowledgment

We would like to thank Tiago A. Almeida, who has suggested numerous improvements to both the content and presentation of this paper. This research was supported by Microsoft EScience Project and Brazilian agencies FAPESP (Grant 05/52959-3, 07/52015-0, 08/54443-2, and 08/50837-6), CNPq (Grant 311309/2006-2, 477039/2006-5, 472402/2007-2, and 309254/2007-8), and CAPES (Grant 01P-05866/2007).

## References

[1] R. C. Veltkamp and M. Tanase, “Content-based image retrieval systems: A survey,” Department of Computing Science, Utrecht University, Tech. Rep. UU-CS-2000-34, 2000.

[2] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, “Content-based image retrieval at the end of the early years,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, 2000.

[3] R. Datta, D. Joshi, J. Li, and J. Z. Wang, “Image retrieval: Ideas, influences, and trends of the new age,” *ACM Comput. Surv.*, vol. 40, no. 2, pp. 1–60, 2008.

[4] R. S. Torres and A. X. Falcão, “Content-based image retrieval: Theory and applications,” *Revista de Informática Teórica e Aplicada*, vol. 13, no. 2, pp. 161–185, 2006.

[5] M. J. Swain and B. H. Ballard, “Color indexing,” *Int’l J. Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[6] R. O. Stehling, M. A. Nascimento, and A. X. Falcão, “A compact and efficient image retrieval approach based on border/interior pixel classification,” in *Proc. Int’l Conf. Information and Knowledge Management*, 2002, pp. 102–109.

[7] N. Arica and F. T. Y. Vural, “BAS: a perceptual shape descriptor based on the beam angle statistics,” *Pattern Recognition Letters*, vol. 24, no. 9–10, pp. 1627–1639, 2003.

[8] R. S. Torres, E. M. Picado, A. X. Falcão, and L. F. Costa, “Effective image retrieval by shape saliences,” in *Proc. Brazilian Symp. Computer Graphics and Image Processing*, 2003, pp. 167–174.

[9] Y. Rubner and C. Tomasi, “Texture-based image retrieval without segmentation,” in *Proc. Int’l Conf. Computer Vision*, 1999, pp. 1018–1024.

[10] M. Unser, “Sum and difference histograms for texture classification,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 118–125, 1986.

[11] J. Almeida, A. Rocha, R. Torres, and S. Goldenstein, “Making colors worth more than a thousand words,” in *Proc. Int’l Symp. Applied Computing*, 2008, pp. 1180–1186.

[12] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Prentice-Hall Inc., 2007.

[13] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int’l J. Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[14] Y. Ke and R. Sukthankar, “PCA-SIFT: A more distinctive representation for local image descriptors,” in *Proc. Int’l Conf. Computer Vision and Pattern Recognition*, 2004, pp. 506–513.

[15] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[16] A. E. Abdel-Hakim and A. A. Farag, “CSIFT: A SIFT descriptor with color invariant characteristics,” in *Proc. Int’l Conf. Computer Vision and Pattern Recognition*, 2006, pp. 1978–1983.

[17] D. G. Lowe, “Object recognition from local scale-invariant features,” in *Proc. Int’l Conf. Computer Vision*, 1999, pp. 1150–1157.

[18] M. Brown and D. G. Lowe, “Recognising panoramas,” in *Proc. Int’l Conf. Computer Vision*, 2003, pp. 1218–1227.

[19] ——, “Unsupervised 3D object recognition and reconstruction in unordered datasets,” in *Proc. Int’l Conf. 3D Digital Imaging and Modeling*, 2005, pp. 56–63.

[20] R. S. Torres, A. X. Falcão, B. Zhang, W. Fan, E. A. Fox, M. A. Gonçalves, and P. Calado, “A new framework to combine descriptors for content-based image retrieval,” in *Proc. Int’l Conf. Information and Knowledge Management*, 2005, pp. 335–336.

[21] Y. Rubner, C. Tomasi, and L. J. Guibas, “The earth mover’s distance as a metric for image retrieval,” *Int’l J. Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.

[22] K. Grauman and T. Darrell, “Efficient image matching with distributions of local invariant features,” in *Proc. Int’l Conf. Computer Vision and Pattern Recognition*, 2005, pp. 627–634.

[23] M. Charikar, “Similarity estimation techniques from rounding algorithms,” in *Proc. Int’l Symp. Theory of Computing*, 2002, pp. 380–388.

[24] A. Bimbo, *Visual information retrieval*. Morgan Kaufmann Publishers Inc., 1999.

[25] I. T. Jolliffe, *Principal Component Analysis*. Springer-Verlag, Inc., 2002.