

Combination of Color and Significant Edge Features in Image Retrieval

Chen Liu^{1,2}

1 Department of Electronic and Information Engineering
Central China Normal University

2 School of Electrical and Information Engineering
Wuhan Institute of Technology
Wuhan, Hubei Province, China
chenliu97@163.com

Zhou Wei³

3 School of Computer
Central China Normal University
Wuhan, Hubei Province, China
w.zhou@ccnu.edu.cn

Abstract—Content Based Image Retrieval (CBIR) has been stirred in last decades since the dramatic rise in the sizes of images databases. In this paper, a CBIR system is proposed combining features of color and significant edge. To promote the whole performance, compact quantizing method in HSV color space and revised Canny edge retrieval methods are adopted in feature retrieval process. User regulated weights make the integrated similarity customized. Experiments show the accuracy and efficiency of the proposed retrieval method.

Keywords- CBIR; feature retrieval; HSV color; Canny edge

I. INTRODUCTION

With the development of the Internet, and the availability of image capturing devices such as digital cameras, image scanners, the size of digital image collection is increasing rapidly. Efficient image searching, browsing and retrieval tools are required by users from various domains, including remote sensing, fashion, crime prevention, publishing, medicine, architecture, etc. The dramatic rise in the sizes of images databases has stirred the development of effective and efficient retrieval systems. The development of these systems started with retrieving images using textual connotations but later introduced image retrieval based on content. This came to be known as CBIR or Content Based Image Retrieval [1]. In CBIR, images are indexed by their visual content, such as color, texture, shapes, which allow users to interactively search image databases looking for those images which are similar to a specified query image.

In the past decade, a few commercial products and experimental prototype systems have been developed, such as QBIC, Photobook, Virage, VisualSEEK, Netra, SIMPLicity. Comprehensive surveys in CBIR can be found in Refs. [2, 3, 4].

Although considerable progress has been made accounts for many researchers' effort, Content-based image retrieval is not yet commercial success. The large commercial image providers are still using human indexers to select keywords for their images, even though their databases contain thousands or, in some cases, millions of images.

In this paper an attempt has been made to integrate the image representation on the basis of color and shape to improve the retrieval performance for general application. The paper is organized as follows. In the next section we describe the features retrieval methods in our system, especially the color space and quantizing method and edge

histogram retrieval process. Section III shows Normalization and Weighted Distance Method for Similarity Measurement. Section IV discusses the experiment and result. Finally, conclusion remarks and future work make up Sections V.

II. RETRIEVAL METHOD

Feature extraction of images is a method of capturing visual content of them for indexing and retrieval [5]. The feature should carry enough information about the image and should not require any domain-specific knowledge for their extraction. Also, they should be easy to compute in order for the approach to be feasible for a large image collection and rapid retrieval. In addition, they should be well related with the human perceptual characteristics since users will finally determine the suitability of the retrieved images. In Content Based Image Retrieval Search Engine, the challenge is to choose features that satisfy the previous characteristics and retrieve the result to users in reasonable time. Color and significant edge are among the more expressive of the visual features. Considerable work has been done in designing efficient descriptors for these features for applications such as similarity retrieval. In this section, we will discuss two of feature extraction methods we used in our search system and the way we use them in image retrieval.

A. Retrieval based on Color

Color is one of the most important attributes for image content. The color representation schemes are essentially invariant under rotation and translation of the input image. Common color features or descriptors in CBIR systems include color-covariance matrix, color histogram, color moments, and color coherence vector [6]. The widely used color histograms are stable object representations in the presence of occlusion and over change in view, but it is sensitive to noisy interference such as illumination changes and quantization errors. While cumulative color histogram produces [7] only slightly better results than color histogram methods, it is more robust with respect to the quantization parameter of the histogram. So in our system cumulative color histogram is chose as an ameliorate descriptor.

The generic Color Histogram Descriptor (CHD) defined in early MPEG-7 [8] experiments is a compound descriptor consisting of color space, color quantization and histogram descriptors. This would allow specification of color histograms with varying numbers of bins and nonuniform quantization of different color spaces.

The different color spaces used in MPEG-7 include the familiar monochrome, RGB, HSV, YC_rC_b, and the new HMMD. The HSV color space is a popular choice for manipulating color. The HSV color space is developed to provide an intuitive representation of color and to approximate the way in which humans perceive and manipulate color. RGB to HSV is a nonlinear, but reversible, transformation. The hue (H) represents the dominant spectral component—color in its pure form, as in green, red, or yellow. Adding white to the pure color changes the color: the less white, the more saturated the color is. This corresponds to the saturation (S). The value (V) corresponds to the brightness of color.

Since most of the color images are define in RGB space due to the availability of images in the RGB format from image scanners, we transform RGB images into HSV color space before the color features retrieval.

To reduce the computational complexity, the HSV color space is then quantized into simple one-dimensional color set of 17 bins based on human’s nature percept of color. We proposed non-linear quantizing method as follows:

- a) The colors with lightness value $V \leq 0.2$ are quantized to black, which is represented as $L=1$. For human eyes can hardly tell the hue and saturation and the pixel seems very dark.
- b) The colors with $S \leq 0.2$ and $0.7 \geq V \geq 0.2$ are quantized to gray, which is represented as $L=2$, and colors of $S \leq 0.2$ and $V \geq 0.7$ are quantized to white, which is represented as $L=3$. Because when $S \leq 0.2$ and $V \geq 0.2$, human eyes can hardly tell the hue and the pixel seems like gray. Human eyes are not sensitive to gray pixel in color images, which are majority in the images of the CBIR systems.
- c) Colors other than those 3 situations defined above in (a), (b) are considered chromatic, which divide into 14 bins depended on the H value and S value. The hue H is divided into 7 regions standing for red, orange, yellow, green, cyan, blue, and violet. The S is divided into 2 regions standing for light colors and deep colors. These add up to $7*2=14$ bins. In detail, while $H \in [296, 360]$ and $[1, 20]$, the color is took for red. $H \in [21, 40]$ is took for orange. $H \in [41, 75]$ is took for yellow. $H \in [76, 155]$ is took for green. $H \in [156, 190]$ is took for cyan. $H \in [191, 270]$ is took for blue. $H \in [271, 295]$ is took for violet. When the hue value is in the same region, the colors look similar, while for $0.7 \geq S > 0.2$, the colors look darker and for $1 \geq S > 0.7$, the colors look brighter. So the saturation S is separated to 2 levels. Since our CBIR system focuses on the most popular color image retrieval, the difference of the lightness value V is not considered in these situations to reduce the influence of environment illumination as much as possible.

Thus, the HSV color space is quantized to $(1+2+7*2) = 17$ bins. The color of each pixel is represented as a quantized value of L . This quantization can greatly reduce the

dimensions of color descriptors and simplifies color feature retrieval process.

After suitable color quantization, an image’s normalized cumulative color histogram of is defined as:

$$I(k) = \sum_{i=0}^k \frac{n_i}{N} \quad k = 0, 1, \dots, L-1 \quad (1)$$

Where L is the total number of the color bins, n_i is the number of pixels with color of i th bin, N is the number of pixels in the image.

B. Retrieval based on significant Edge

Significant edge detection is widely used in computer vision to locate sharp intensity changes and to find object boundaries in an image. For edge gives mickle information such as texture and object shape in an image, it’s widely used in CBIR system. For example, Edge directions histogram descriptor produced from image edge are used to descript texture feature in ref [9] and shape feature in ref [10].

The edge histogram descriptor [11] captures the spatial distribution of edges, which is found to be quite effective for representing natural images with the primary application being image-to-image matching. The performance can be further enhanced by using this descriptor in conjunction with other image features, such as color.

The histogram is calculated based on the edge image generated in the preprocessing stage using the Canny edge operator. The histogram of the edge directions is represented by 72 bins, each spanning 5° .

To reduce the effect of rotation we smooth the histograms. A histogram can be treated as a 1-D discrete signal. Smoothing can then be defined as:

$$I_s[i] = \frac{\sum_{j=i-k}^{i+k} I[j]}{2k+1} \quad (2)$$

where I_s is the smoothed histogram, I is the original histogram and the parameter k determines the degree of smoothing. In our experiments we used $k = 1$.

Among the edge detection methods proposed so far, the Canny edge detector [12]. is the most rigorously defined operator and is widely used. The popularity of the Canny edge detector can be attributed to its optimality according to the three criteria of good detection, good localization, and single response to an edge. It also has a rather simple approximate implementation. Although the work was done in the early days of computer vision, the Canny edge detector (including its variations) is still a state-of-the-art edge detector. Unless the preconditions are particularly suitable, it is hard to find an edge detector that performs significantly better than the Canny edge detector.

The Canny edge detector classifies a pixel as an edge if the gradient magnitude of the pixel is larger than those of pixels at both its sides in the direction of maximum intensity change. The Canny edge detector follows the steps below.

Step 1. Filter out any noise in the original image using Gaussian filter before trying to locate and detect any edges. The Gaussian smoothing can be performed using standard convolution methods.

Step 2. Determine the gradient magnitude and gradient direction at each pixel to find the edge strength with Sobel operator.

Step 3. Nonmaximum suppression is used to trace along the edge in the edge direction and suppress any pixel value (sets it equal to 0) that is not considered to be an edge. This will give a thin line in the output image.

Step 4. Trace edges through the image using hysteresis thresholds.

During Canny edge detecting process, we revise some parameters to improve the system performance from two points below.

1) In Step 1, Gaussian filter mask is a very important component while smoothing the given image. The 5×5 Gaussian mask matrix used in our implementation is shown in (3), which has been revised after exploratory analysis. The discrete parameter of Gaussian smoother function is $\sigma = 1$.

$$G(i, j; \sigma) = \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 4 & 6 & 4 & 2 \\ 3 & 6 & 7 & 6 & 3 \\ 2 & 4 & 6 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix} \quad (3)$$

2) The 2 hysteresis thresholds used in Step 4 are referred to T_1 as a high threshold and T_2 as a low threshold. Any pixel in the image that has a value greater than T_1 is presumed to be an edge pixel, and is marked as such immediately. Then, any pixels that are connected to this edge pixel and that have a value greater than T_2 are also selected as edge pixels. If you think of following an edge, you need a gradient of T_2 to start but you don't stop till you hit a gradient below T_1 . In our experiment, we use 70% of the image gradient magnitude as T_1 , and $T_2 = 0.4 * T_1$.

III. NORMALIZATION AND WEIGHTED DISTANCE METHOD

A. Normalization

In the retrieval algorithm described in the previous section, we have assumed that the similarity values of each representation, S_i (refer to S_C or S_E), are of the same dynamic range, say, from 0 to 1. Otherwise, the linear combination of S_i 's to form S becomes meaningless. One S_i may overshadow the others just because its magnitude is large. For the same reason, when calculating S_i 's, the vector components, r_{ij} 's, should also be normalized before applying the similarity measure. We refer the normalization of r_{ij} 's as intra-normalization and the normalization of $S(r_{ij})$'s as inter-normalization. The Gaussian normalization method is adopted.

B. Weighted Distance Method

In our QBIC system, the three features of color and edge direction histogram may be used separately or integrated together to measure the similarity of the images.

Euclidean distance is used as a metric to compute the distance between the color and edge feature vectors. To measure the similarity of the images by color feature of cumulative color histogram, we use three optional methods

of Euclidean distance, histogram intersection or weighed distance.

We integrated the results of the edge-based retrieval and the color-based retrieval by combining the associated similarity values. Let S_C be the similarity index between Q and I on the basis of color and S_E be the similarity index between Q and I on the basis of edge. We define an integrated similarity index S , between Q and I as:

$$S = w_C S_C + w_E S_E \quad (4)$$

where S_C , and S_E are the separate similarity index, and w_C and w_E are the weights assigned to the color and edge-based similarity, respectively. Note that the value of $S(I, Q)$ lies in the interval $[0, 1]$. If images I and Q are identical then $S(I, Q) = 1$.

A set of top 20 images on the basis of the total similarity index S are presented as a result of the query. Note that S lies in the interval $[0, 1]$. We used user regulated weights with constraint of $w_C + w_E = 1$ in our experiment. This affords the customs an convenient way to adjust the parameter of the retrieval process.

IV. IMAGE DATABASE

The image database uses images from the open image source of Washington University, downloaded from <http://www.cs.washington.edu/research/imagedatabase/groundertruth/>. Our image database includes 200 images of 10 classes categorized by semantic meanings, which are arborgreens, cannonbeach, cherries, football, greenlake, Indonesia, Italy, Japan, Swissmountains, yellowstone. Each image has its brief text annotations by some key words.

For illustration, we use 5 images selected in categories as query images separately, which is cannonbeach1、cheeries11、football5、greenlake1、swissmountains10. In order to test the stability and robustness of the system, we have tried to retrieve every image in the database under a variety of conditions. The query image each has copies of rotated, scaled, and with 5% Gaussian noise added in the database. The experiments were conducted on the 200 color images in our database.

V. RETRIEVAL RESULTS

Table 1 represent the results of retrieval based on integrated similarity index, where we initialized the weight as $w_C = 0.5$ and $w_E = 0.5$. All the weight can be changed manually during the processes of retrieval as a simple method of relevance feedback. Fig.1 shows one retrieval result in the system. Where left one is the query image and the 20 images in the right are retrieved images from the image database showed according the similar rank.

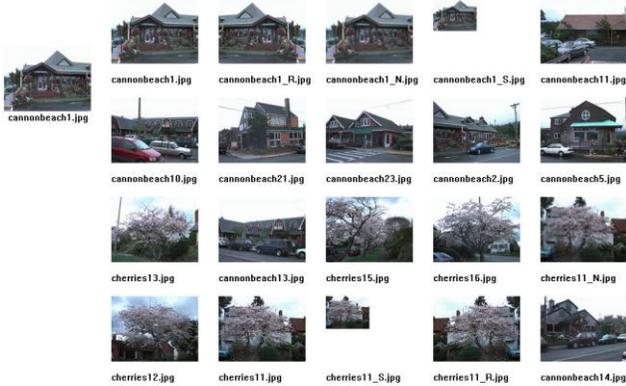


Figure 1. A sample of output of the CBIR system

The most obvious conclusions from those experiments are:

- The same image is obtained as the first output, and relevant images including its rotated, mirrored, noised or scaled copies always get top few ranks.
- A lot of similar images come as outputs in the beginning of the results, even the images are rotated, scaled or noised. This shows the robust of the CBIR system.

But with the number of result image increases, the precision rate decreases notably. This may partly due to the ground truth database classifies it images in high level semantic way, not all of the images in the same category are of the same visual characteristics. Some image in same classed may have very different visual characters, so the visual features' combination may not enough to realize semantic retrieval.

Fig. 2 shows the comparison of the performance of using different retrieval features in our experiment. The performance of the CBIR system is represented by the average precision rate of top n images in the ranked retrieval result. From Fig.2 we can see that the combination of the two features of color and significant edge produce better performance than one feature based system.

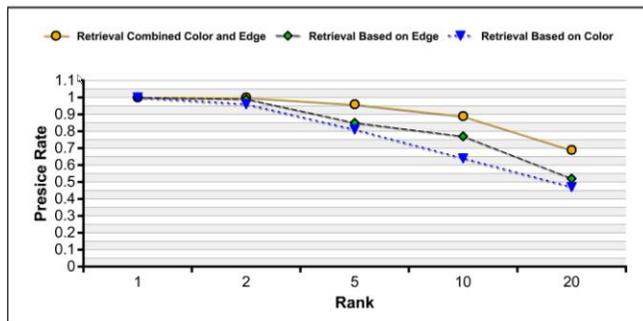


Figure 2. Comparison of the performance of using different retrieval features

VI. CONCLUSIONS AND FUTURE WORK

All experiments come with satisfied results and show the efficiency of the system.

But different classes of images may get different level of results on same parameters or same weight. Introducing adaptive weights determination and appropriate relevance feedback method may push forward the performance of CBIR system greatly.

REFERENCES

- [1] J. R. Smith, and S. F. Chang, "Tools and techniques for color image retrieval," Storage and Retrieval for Still Image and Video Databases Iv, vol. 2670, pp. 426-437, 1996.
- [2] V. Vani, and S. Raju, "A Detailed Survey on Query by Image Content Techniques," Recent Advances in Networking, Vlsi and Signal Processing, C. A. Bulucea, N. Kalamani, N. Mastorakis et al., eds., pp. 204-209, Athens: World Scientific and Engineering Acad and Soc , Feb. 2010.
- [3] F. Long, H.J. Zhang, D.D. Feng, "Fundamentals of content-based image retrieval," in: D. Feng (Ed.), Multimedia Information Retrieval and Management, Springer, Berlin, 2003.
- [4] Datta, R., et al., "Image retrieval: Ideas, influences, and trends of the new age". ACM Computing Surveys (CSUR), 2008. 40(2): pp. 5.
- [5] M.S. Nixon, A.S. Aguado, "Feature Extraction and Image Processing, Academic Press", 2 Ed., January 22, 2008.
- [6] Huang, W., Y. Gao and K.L. Chan, "A review of region-based image retrieval". Journal of Signal Processing Systems, 2010. 59(2): pp. 143--161.
- [7] M. A. Stricker, M. Oregon. "Similarity of Color Images," Proc SPIE Int Soc Opt Eng, vol. 2420, pp. 381-392, 1995.
- [8] C. Shih-Fu, T. Sikora, and A. Purl, "Overview of the MPEG-7 standard," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 11, no. 6, pp. 688-695, 2001
- [9] Won, C.S., D.K. Park and S.J. Park, "Efficient use of MPEG-7 edge histogram descriptor." Etri Journal, 2002. 24(1): p. 23--30.
- [10] Jain A K Vailaya A. "Shape-based retrieval a case study with trademark image databases." Pattern Recognition, 1998 31 (9) 1360-1390.
- [11] Manjunath, B.S., et al., "Color and texture descriptors. Circuits and Systems for Video Technology," IEEE Transactions on, 2001. 11(6): p. 703--715.
- [12] Shapiro L.G. & Stockman G.C. "Computer Vision. London": Prentice Hall, pp. 326, 2001