Region-based Image Retrieval System Using Efficient Feature Description

ByoungChul Ko, Hae-Sung Lee, Hyeran Byun
Dept. of Computer Science, Yonsei University, Korea, 120-749
{soccer1, geneel, hrbyun}@aipiri.yonsei.ac.kr

Abstract

In this paper, we introduce a region-based image retrieval system, FRIP. This system includes a robust image segmentation scheme using scaled & shifted color and shape description scheme using Modified Radius-based Signature. For image segmentation, by using our proposed circular filter, we can keep the boundary of object naturally and merge small senseless regions of object into a whole body. For efficient shape description, we extract 5 features from each region: color, texture, scale, location, and shape. From these features, we calculate the similarity distance between the query and database regions and it returns the top K-nearest neighbor regions.

1. Introduction

As the use of digital video information grew rapidly in recent years, it became more important to manage multimedia databases efficiently. Furthermore, the dramatic improvements in the hardware technology have made it possible in the last few years to process, store and retrieve huge amounts of data in multimedia format [1].

Many researchers and institutions are currently involved in providing tools and methods to efficiently manage pictorial digital libraries. Query-by-image is perhaps the most popular way in the content-based image retrieval (CBIR) system. But, if the CBIR system only use the global properties of image, it may be easy to miss many similar images. So, most recent CBIR systems have been focused on object or region based image retrieval.

Several systems have been developed recently to search object or region through image database using color, texture, and shape attributes (e.g. QBIC [2], Netra [3], VisualSEEk [4], Blobworld [5]).

Most of these systems, basically, work in the same way: (1) images are segmented into several regions, (2) some feature vectors are extracted from each region in the database and (3) the set of all feature vectors is organized into a database index. At the query time, features are extracted from the query image, a user-provided sketch or a region from a segmented image and is matched against the feature vectors in the index.

In this paper, we present FRIP(Finding Region In the Pictures), a retrieval system based on region. FRIP uses color for region segmentation and uses color, texture, location, and shape information for indexing regions in the database.

This paper is organized as follows. In the next section, we describe the image segmentation process. In section 3, the feature extraction techniques are introduced. Section 4 is dedicated to similarity matching algorithm. The experimental results and evaluations are explained in section 5.

2. Image segmentation process

Image segmentation refers to partitioning an image into different regions that are homogeneous or similar in image characteristics. In this work, we segment an image into regions using scaled and shifted color coordinate and our circular filter.

2.1. Scaling and shifting color coordinate

The human visual system can be modeled as using three perceptual attributes; hue, saturation, and luminance. Here, the hue is usually preserved to not disturb the natural coloring of the image. So, to enhance an image quality, we need to modify not the hue component but the saturation and luminance component. Normally, color image is transformed from RGB to YIQ color to modify the saturation and luminance. After transformation, modified color coordinates are converted back to RGB color space again. But, this process is computationally not efficient. So, in this system, we use a hybrid technique for image enhancement [6].

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = k\frac{L'}{L}\begin{bmatrix} R \\ G \\ B \end{bmatrix} + (1-k)\left(\begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} L'-L \\ L'-L \\ L'-L \end{bmatrix}\right) \quad (1)$$

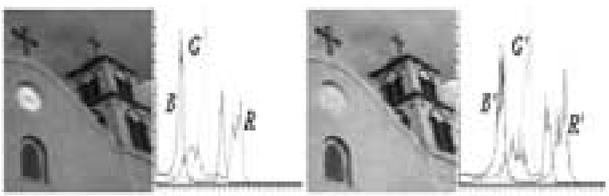
where ,

R', G', B' : *transformed RGB color*

L' : *Contrast stretched luminance from L*

k : *scaling and shifting factor*

The luminance is defined from LHS color system: $L=0.299R+0.587G+0.114B$. Here, we define desired luminance L' as the contrast-stretched luminance. The contrast of a luminance is the distribution of light and dark pixels of image. Images with good contrast exhibit a wide range of pixel values. So, we apply this method to L value in order to expand the image histogram to cover all ranges of pixels. If k=0.5, then ($R'G'B'$) is the average of the processed vectors of scaling and shifting in the RGB cube. But, through experiment, we found out that the scaling of saturation is more important than the shifting of luminance. So, we decided 0.7 as the value for *k*. This hybrid method alleviates the over-saturation and under-saturation problem in the scaling and shifting techniques. By avoiding the conversion of RGB color space to any other color space, we reduce the computation time in addition to obtaining enhanced color image.



(a) (b)

Figure 1. (a) Original image (b) scaled and shifted image

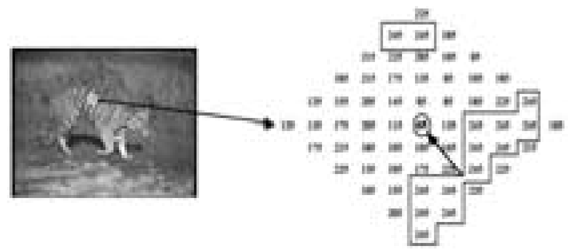
2.2. Region segmentation by the proposed method

In this paper, image segmentation is divided into two parts: (1) first-level segmentation step using circular filter and region merging, and (2) iterative step using region labeling and iterative region merging.

2.2.1. First-level segmentation. First, modified colors in the image are coarsely quantized with significant color values. For example, if the quantization interval is 10 and the color values of $R'G'B'$ are 18, 32, and 251, these values are quantized with the middle value of the interval, 15, 35, and 255. By this step, the number of color bins are reduced from 256 to 25. Secondly, median filtering is applied to eliminate noise. Thirdly, image averaging is applied. By averaging filter, peak spots or small different color regions in the whole body region are converted to blurred appearance which makes it easer to merge into a neighboring big region.

After image averaging, we apply our proposed circular filter. This circular filtering step is essential to merging some senseless regions which cannot be merged to whole region by normal region merging algorithm. In our algorithm, we use two kinds of different sized filter. The big one is an 11×11 window, as shown in Figure. 2-(b), and the small one is a 7×7 window. Here, the window of big one is useful for determining a set of small initial areas by removing the senseless regions for region merging, in the first level segmentation step and the window of small one is useful for removing the remaining small senseless regions, in the iterative merging step.

From Figure 2-(b), if the number of pixel value, '245', is more than the number of any other pixel values and satisfy the equation (2), the center point, '65', is changed into '245'. Here, we use $\boldsymbol{T_1=200}$ as a threshold for color.



(a) (b)

Figure 2. Circular filtering step

For a given color image, above processes are computed in each of the three color bands.

$$|C(a,b)-M_c| \le T_1 \tag{2}$$

where,

$C(a,b)$: *Color value of center point*

M_c : *Color value with maximum number of pixels in the color region*

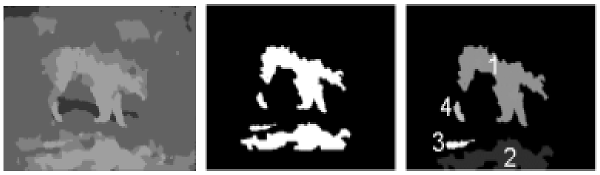
By using circular filter, we can keep the boundary of object naturally and merge stripes or spots of objects into body region.

Finally, we use region merging algorithm to merge small regions into adjacent regions which have the most similar color property based on the mean of the region, R_m^i and homogeneity function (3).

$$\sum_i^N \left| R_{R'G'B'} - R^i_{m\,R'G'B'} \right| \le T_2 \tag{3}$$

where, N : *Number of neighbor regions*

2.2.2. Region labeling and iterative merging. At the iterative step, we use connected-component algorithm to label regions. By using region-labeling, we can protect over-segmentation problem. This process can be obtained by combining the three color channel as $I=(3*R'+4*G'+2*B')/9$. Each region which has the same color value is projected to buffer as a value of 255. This binary image is raster scanned from left to right and from top to bottom. In this step, each region is labeled by a different region number. In addition, regions that are N (30) pixels or less in size are considered to be non-significant region like noises and are not labeled.



(a) (b) (c)

Figure 3. Region labeling (a) segmented regions by first-step (b) projected same color regions to buffer (c) labeled regions as a different number by connected component.

After the region labeling, if the number of region is over 30 and the threshold ***T2*** for region merge is below 100, we repeat the circular filtering with a 7×7 window and region merging step with increased ***T2*** until the two con conditions are satisfied. Here, to the save storage space of shape description for region-based image retrieval application, we restrict the maximum region number to 30.

2.3. Segmentation results

Experiments have been carried out with this method on a set of four color spaces, RGB, HSI, CIE-XYZ and R′G′B′ and database consists of dynamic kinds of images described in chapter 5. The average segmentation time requires approximately 35 second per one image using a Pentium PC, 450 MHz. Here, we found that the scaled and shifted color system, R′G′B′ provides the best segmentation results than others, even though the image intensity is small or partially changed by shadow and surface curvature.

3. Feature extraction

From segmented image, we need to extract features of each region. That is, the semantics of image should be extracted and stored as index.

Features (color, texture, scale, location, shape) of each region are used to describe the content of image. These five features are:

1. Color : We extract average color of ***RGB*** color space from each region.
2. Texture : We choose the Biorthogonal Wavelet Frame(BWF). By BWF, we can obtain a fast and precise directional feature compared with multi-resolution method.
3. Scale : We describe a scale as the size of the region; that is, scale is the number of pixels in the region.
4. Location : After region segmentation process, the centroid is calculated from each region. Location of each region is defined by this centroid.

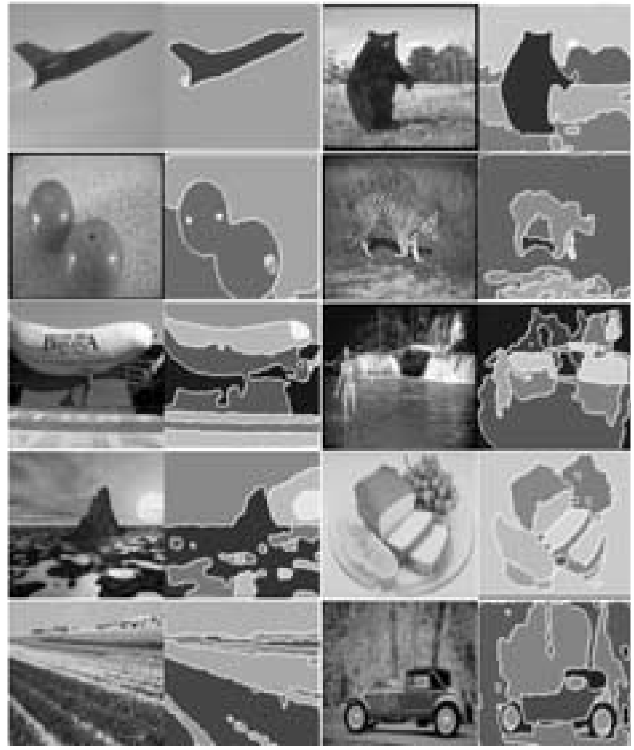
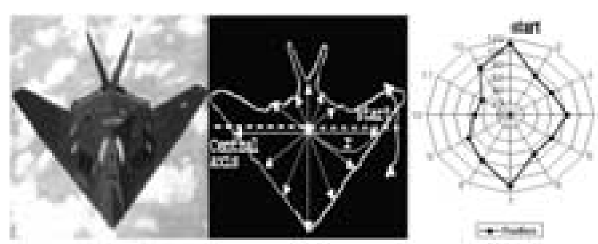


Figure 4. The results of the segmentation by our algorithm

5. Shape : For efficient shape matching, we use two properties. One is the eccentricity of region and the other is our Modified Radius-based Signature (MRS). At the first step, if the eccentricity is quite different between query and database region, these database regions are removed. So, at the second step, we can significantly reduce comparison time for shape matching. At the second step, with major-axis and centroid of region, we estimate the same starting point and radii of region regardless of rotation and scale change. By our MRS, we can get the invariant results about small distortion as well as rotation and scale changes of region because it provides local information of shape in addition to global information. Here, to reduce index size, we extract 12 radius distance values per 30 degrees and add these to the index for the region.



(a) (b) (c)

Figure 5. Region signature: (a) original image (b) region boundary (c) polygon by our signature

Proceedings of the International Conference on Pattern Recognition (ICPR'00)
1051-4651/00 $10.00 @ 2000 IEEE

Finally, we save 12 statistical properties of each region of the image as an index to database.
The properties are shown in below:

```
Image
  keys : imageNo
 {
  attribute int RegionNo;
  attribute int AverRed(AR);
  attribute int AverGreen(AG);
  attribute int AverageBlue(AB);
  attribute int NumberOfPixels(NP);
  attribute int CenterOfX(Cx);
  attribute int CenterOfY(Cy);
  attribute int MajorLength(Rmax);
  attribute int MinorLength(Rmin);
  attribute Array<int> Signature[12];
  attribute float AmplOfYDirect(Yd);
  attribute float AmplOfXDirect(Xd);
 }
```

4. Stepwise matching strategy for region comparison

The actual matching process is to search for the k elements in the stored region set closest to the query region. After regions are segmented, the user selects a region that he/she wants to search. Finally, by the user specified constraints, such as (1) scale-care/don't care, (2) shape-care/don't care, (3) location-care/don't care, the overall matching score is calculated by linear combination of each features.

In order to determine the 5 weights $(a_1 \sim a_5)$, we normalize all the feature vectors to 0 ~ 1 value, and their associated weightings are adjusted to *1/f*, where *f* is the number of feature vectors selected by user. The system carries out image retrieval using K-nearest neighbor search, based on current weightings to compute the similarity between the query region and database regions, and finally, it returns the top K-nearest images.

During the matching, linear combination of five distances is used to carry out final score according to the user constraints.

5. Experimental results and evaluation

We have performed a variety of queries using a set of 2,600 images from the *WWW* and Corel photo-CD, containing various categories such as natural images (e.g. landscape, animals) and synthetic images (e.g. graphics, drawing). This system is developed in Visual C++ 6.0 language as an off-line system. The retrieval results are accessible at http://vip.yonsei.ac.kr/Frip. Since there is no standard criterion for the performance, we performed tests only on some specific domain data such as, sunset, eagle, and tiger. The users can choose color, texture, scale, shape, and location in order to search regions more precisely. Figure 6 shows the retrieval results of sunset region. We obtained an average retrieval rate of 78% on these 3 domains.

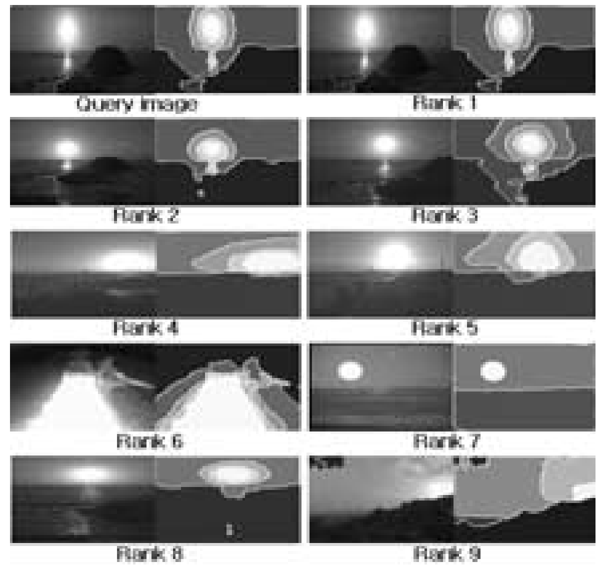


Figure 6. Retrieval results of sunset (left: original image, right: segmented image)

6. References

[1] L. Cinque, S. Levial, K.A. Olsen, and A. Pellicano, "Color-Based Image Retrieval Using Spatial-Chromatic Histograms", *Int. Conf. on Multimedia computing and Systems,* June 7-11, 1999, Florence, Italy

[2] Faloutsos, M. Flickner, W.Niblack, D. Petkovic, W. Equitz, R. Barber. "Efficient and Effective Querying by Image Content", *Research Report #RJ 9203 (81511)*, IBM Almanden Research Center, San Jose, Aug.

[3] W.Y.Ma, B.S. Manjunath. "Netra: A toolbox for navigating large image database", *IEEE International Conference on Image Processing*, 1997.

[4] J.R. Smith and S.F. Chang. "VisualSEEk: A Fully Automated Content-Based Image Query System", *ACM Multimedia* , Boston MA, 1996.

[5] Carson, M.Thomas, S. Belongie, J.M. Hellerstein, and J. Malik. "Blobworld: A system for region-based image indexing and retrieval", *In Proc. Int. Conf. Visual Inf. Sys*, 1999.

[6] Christopher C. Yang, Jeffrey J. Rodriguez, "Efficient Luminance and Saturation Processing Techniques for Color Images", *Journal of visual communication and image representation,* vol. 8, pp. 263-277, 1997.

Proceedings of the International Conference on Pattern Recognition (ICPR'00)
1051-4651/00 $10.00 @ 2000 IEEE