

## COMPUTATIONAL SIMILARITY BASED ON CHROMATIC BARYCENTER ALGORITHM

Shiuh-Sheng Yu, Jinn-Rong Liou, and Wen-Chin Chen  
 Communications & Multimedia Laboratory  
 Department of Computer Science and Information Engineering  
 National Taiwan University  
 Taipei, Taiwan, R.O.C.

### Abstract

We propose in this paper a novel histogram-based Chromatic barycenter algorithm for computing the similarity between images. The algorithm is simple, intuitive, and thus easy to implement. With its high precision and recall rates, the algorithm can be used as retrieval and scene change detection methods in very large video and image databases. When being used to process DCT based compressed data, the algorithm is even more efficient, as some of the time consuming decompression procedures can be skipped. This makes the algorithm an ideal tool for realtime and interactive multimedia applications with JPEG and MPEG data.

keywords: Similarity, Chromatic barycenter, Histogram-based, Multimedia, Database, DCT, Scene change.

### 1. Introduction

With the rapid growing of Internet, people are facing digital libraries containing tremendous amount of multimedia documents. The information in these documents may contain image or video data. To rapidly access these digital information one needs efficient methods to query and retrieve image and video data by their contents.

The computational measurement of *perceptual similarity* is the key component to query image and video by their contents. A good measurement must be able to deal with image distortions that are usually introduced when image and video data are acquired, transmitted, and processed. The distortions one encounters most often are *resolution mismatch*, *color shift*, and *object movement*. We call that two images have resolution mismatch if they only differ in size. Color shift refers to the difference in color tone, contrast, or brightness, which result from film development or printing process. Object movement is due to content objects or camera moving when one takes the picture or video.

Most popular similarity measurements are based on *segmentation* and *region detection* methods [2,3,4,5]. These measurements suffer the following difficulties:

- Segmentation is never perfect.
- They are unable to identify high frequency images.
- They have high computational overhead.
- They are unable to process compressed images.

These difficulties constrain segmentation based

methods to some special domains, for example, face identification.

To deal with the afore-stated difficulties, an alternative *histogram-based* similarity measurement is proposed in [4,6]. Rather than extracting the contents of images, the histogram-based method tries to distinguish images by analyzing the color distribution statistics (histograms) of images. However, these measurements have the drawback of low precision and recall rates, which results from that it extracts only low level color features of images. This drawback makes the measurements infeasible for huge amount of data. To overcome the problem, we propose a novel histogram-based *Chromatic barycenter algorithm*. The algorithm enjoys the properties of very high precision and recall rates, low computing overhead, and the ability to directly process DCT based compressed data. Throughout this paper, the algorithm will be referred for short as Barycenter algorithm.

This paper is organized as follows. Section 2 introduces the Chromatic barycenter algorithm. Section 3 discusses the advantages of the algorithm. Section 4 discusses how the algorithm be applied efficiently to DCT-based compressed images. Section 5 describes how the algorithm can be used for scene change detection of video data. Finally, we conclude the paper in Section 6.

### 2. Barycenter Algorithm

*Chromatic barycenter* is the average value of pixels of an image region in the color space. For example, the barycenter of a green region in true color RGB space is (0, 256, 0). The similarity between two regions is determined by the distance  $d$  between their barycenters. The similarity function  $s$  can formally be defined as:

$$s(d) = \begin{cases} 100 & \text{if } d \leq T_1; \\ 100\left(\frac{T_2 - d}{T_2 - T_1}\right) & \text{if } T_1 < d \leq T_2; \\ 0 & \text{if } d > T_2, \end{cases}$$

where  $T_1$  is determined by the sensitivity of human eyes and  $T_2$  is determined by the precision requirement. To mimic human eyes, we calculate the barycenter in YCrCb domain and define  $T_1$  and  $T_2$  as 1/128 and 1/8 of the color space.

As the barycenter of an image only represent the average color distribution, it is too rough to distinguish images with only a barycenter. To calculate the similarity

of two images, we then compute the similarities of their corresponding subregions and take the average. The similarity between image A and B can formally be defined as

$$Sim_1(A, B) = \frac{1}{n} \sum_{i=1}^n s(d_i(A, B)),$$

where

$$d_i(A, B) = \|C_i^A - C_i^B\|$$

is the distance of  $C_i^A$  and  $C_i^B$ , the barycenters of the  $i$ th subregions of A and B. It is easy to see that the more subregions we divide, the more accuracy we obtain. However, too many subregions may cost too much computing power and makes the algorithm intolerable to object movement. Experiments suggest that 3x3 subregions division is practical and yields good results for most images.

Barycenter algorithm tries to compensate color distortions by adjusting barycenters before computing the similarity between subregions. Assume  $C_0^A$  and  $C_0^B$  are the barycenters of image A and B, then the shift vector  $T(A, B)$  of A and B for adjusting the barycenters and their modified distance  $d_i'$  are defined as:

$$T(A, B) = C_0^A - C_0^B;$$

$$d_i'(A, B) = \|C_i^A - C_i^B - T(A, B)\|.$$

The similarity between A and B after adjusting is defined as:

$$Sim_2(A, B) = \frac{1}{n} \sum_{i=1}^n s(d_i'(A, B)).$$

Finally, the similarity between image A and B for the Chromatic barycenter algorithm is defined by taking the maximum of  $Sim_1$  and  $Sim_2$ :

$$Sim(A, B) = \begin{cases} 0 & \text{if } d_0(A, B) \geq T_3; \\ \max\{Sim_1(A, B), Sim_2(A, B)\} & \text{otherwise,} \end{cases}$$

where constant  $T_3$  is the threshold that human eyes can distinguish from different images. We currently choose  $T_3$  as 1/8 of the color space.

### 3. Applying Barycenter Algorithm To Daily Life Images

Recall and precision rates are two major factors that are used to justify the effectiveness of an algorithm for computing the similarity between images. In this section, we shall carry out extensive simulations to show that the proposed Barycenter algorithm is quite effective in practical applications in terms of recall and precision rates.

For the recall rate, we shall do the experiment with a number of daily life pictures which have resolution mismatch, color shift, and object movement.

Since the barycenters are the same for images that have resolution mismatch, the Barycenter algorithm can easily overcome the resolution mismatch. The algorithm can also overcome the color shift with the barycenter adjusting step. To test the performance of the barycenter adjusting step, we enhance the power and RGB values of an image from 10% to 100% and then compute the similarities between the enhanced and the original image. Figure 1 shows that these images have high similarities even though there are heavy color shifts.

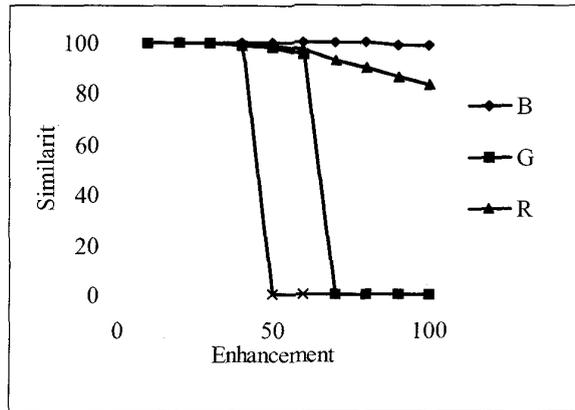


Figure 1. Similarities versus color shift effect

The performance of Barycenter algorithm for images is shown in Figure 2. The image on the left side is scanned from a magazine; the other two are scanned from the original photograph but with two different scanners. Although there are color shifts, the Barycenter algorithm still shows that they all have high similarities.

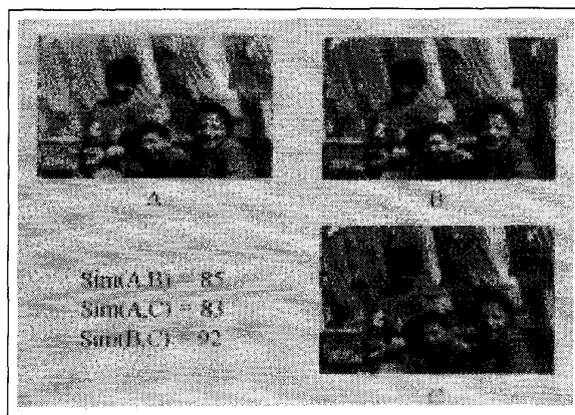


Figure 2. Images with color shift

As for the object movement, only those subregions containing moving objects are different. Consequently, the algorithm can still pinpoint certain similarities in these images. To evaluate the performance on images with slight object movement, we compute the similarities between consecutive frames of a video which shows a salesman advertising his product by hands. The simulation results in Figure 3 show that these frames all have similarity scores above 92.

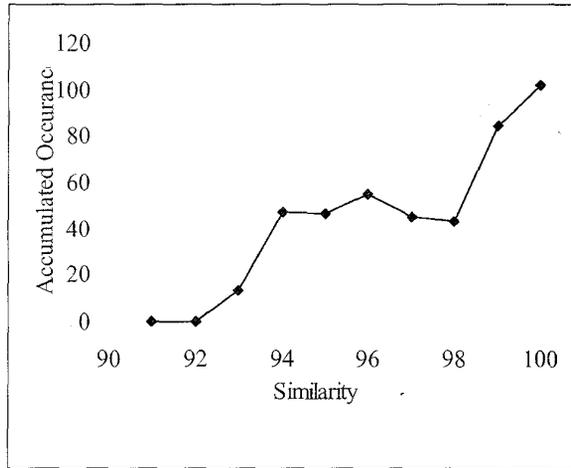


Figure 3. Similarities between frames of a video

Applying Barycenter algorithm to images with more object movement also results interesting outcomes. In figure 4, although the model has different gestures, and the trademark at the upper-right corner has also been moved to the lower-left corner, Barycenter algorithm still returns a similarity score of 81. Experiments shows that Barycenter algorithm can overcome various distortions that occur in daily life pictures. In other words, the algorithm yields very high recall rate.

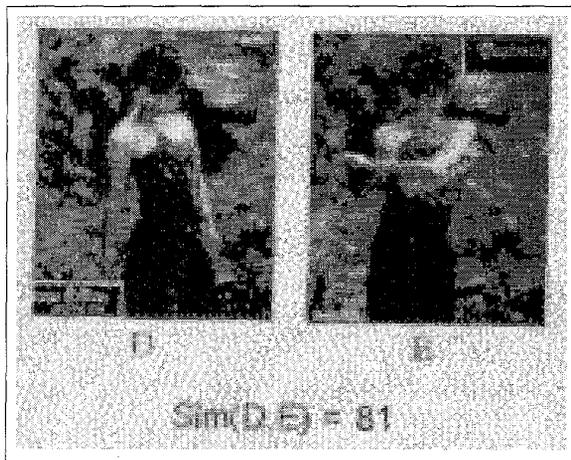


Figure 4. Images with object movement

To evaluate the precision rate, we first select by human eyes a set of 133 daily life images that includes paintings, photo scenes, human faces, rendering images, and so on. We then compute their similarities pairwise using Barycenter algorithm. Figure 5 gives the simulation results, which shows that only 17 pairs (0.2%) have similarity scores above 80. With such a high precision rate, we believe that Barycenter algorithm can be used as the retrieval tool for very large image databases.

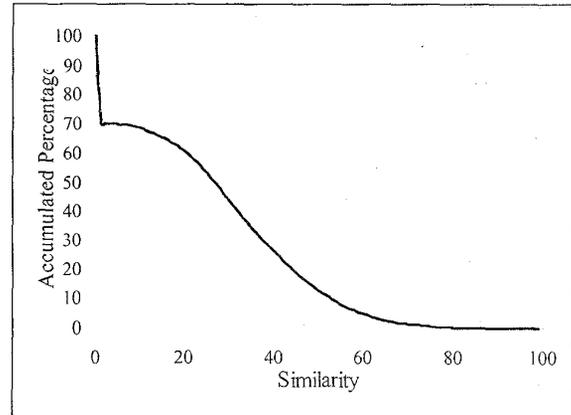


Figure 5. Similarities between different images

#### 4. Applying Barycenter Algorithm To DCT Based Compressed Images

Most digital images are stored in compressed formats to save storage space. With the popularity of JPEG and MPEG compression standards, computational similarity algorithms which can effectively process DCT based compressed data will be very desirable. As the DC value of DCT coefficients is the barycenter itself, Barycenter algorithm can compute the similarity between compressed images without having to fully decompress them. Consider the JPEG decoding procedures shown in figure 6, Barycenter algorithm can skip the most time consuming IDCT process, and process only the DC component. When applying Barycenter algorithm to JPEG images, the processing and I/O time is about 50% less than raw PPM images.

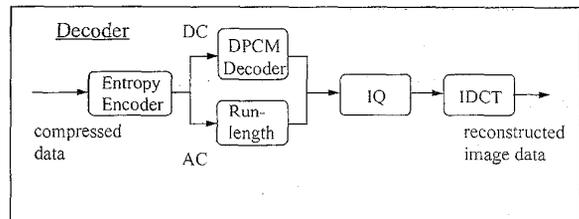


Figure 6. JPEG decoder's block diagram

JPEG uses 8x8 macro blocks as the compression units which may not fit exactly into one subregion. Using only DC component without fully decompressing the whole macro block will introduce slight difference between the accurate barycenters and those obtained from the DC values. As the Barycenter algorithm can overcome the color shift and object movement phenomena, it can tolerate these slight errors. Experiments shows that the Barycenter algorithm yields the same query results whether on PPM or JPEG images.

### 5. Applying Barycenter Algorithm To Video Scene Change Detection

Adjacent frames of the same video shot usually have slight differences result from object movement. These frames are normally captured at about the same time and the same places, so that we can assume that there is no color shift. To detect the scene changes in a video stream, we can apply Barycenter algorithm to each consecutive pair of adjacent frames. If the similarity of two adjacent frames is lower than a threshold, we may consider there is a scene change.

Since there is no optimal threshold suitable for every cases, we adopt the following strategy. We use two thresholds. When the similarity of two adjacent frames is lower than the lower threshold, it is assumed that there is a scene change. When the similarity of two adjacent frames is higher than the higher threshold, there is no scene change. For those pairs having similarities between these two thresholds, we ask the users if there is a scene change. After experimenting with various video streams, we found setting thresholds at 80 and 90 makes Barycenter algorithm excellent for scene change detection.

To show the performance of Barycenter algorithm, we use two video streams as illustrating examples. The first one is cut from a famous Japanese cartoon "Totoro." The video stream has six scene changes as shown in Figure 7. The data in Figure 8 indicates that the similarities of these scene changes are all below 70 and the similarities of the rest are all above 90. Choosing thresholds 80 and 90 makes Barycenter algorithm perfect for this video.

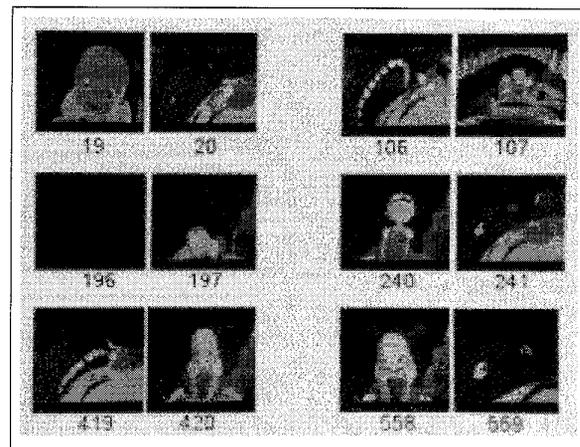


Figure 7. Scene changes of video 1

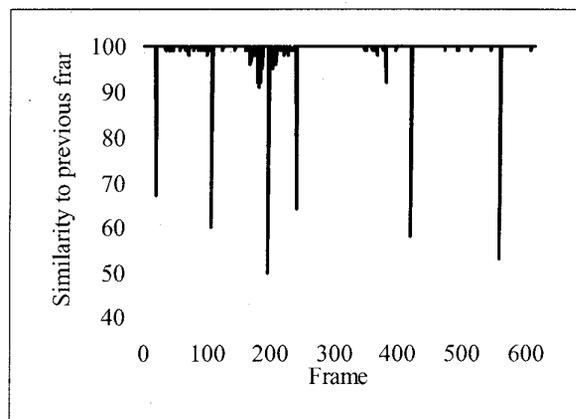


Figure 8. Scene change detection results for video 1

The second video stream shown in Figure 9 is taken from a TTV news report. The video contains an anchor man followed by an assembly report. The assembly report contains many shots taken in the same room, which makes the detection of scene changes difficult. Barycenter algorithm detects out nine correct scene changes. It also reports three ambiguous scene changes with one is selected and the other two are rejected by human eyes.



Figure 9. Scene changes of video 2.

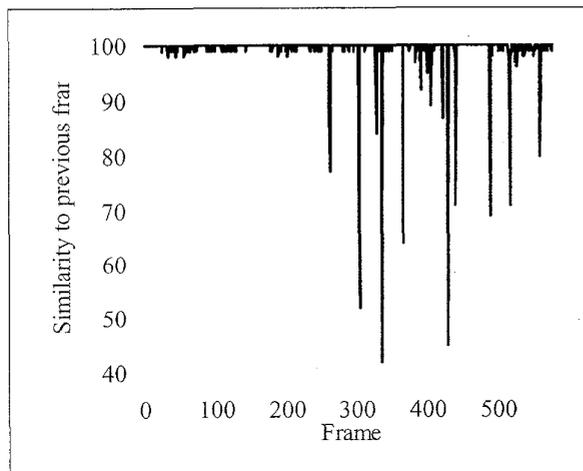


Figure 10. Scene change detection results for video 2

In addition to the very high precision rate, the ability to detect scene changes without having to fully decompress the MPEG video streams makes the algorithm suitable for realtime and interactive applications. With the fast growing of MPEG video volumes, the algorithm can be a good tool for very large databases to parse video streams.

## 6. Conclusions

We propose Chromatic Barycenter algorithm as an efficient tool for computing the similarity between images. The algorithm is simple, intuitive, and thus easy

to implement. With its high precision and recall rate, the algorithm can be used as the retrieval and scene change detection methods in very large video and image databases. Moreover, the algorithm's ability to process the DCT based compressed data makes the algorithm even more efficient and enable it to serve as an ideal tool for real time and interactive multimedia applications.

## 7. References

- [1] T. C. Chiueh, "Content-Based Image Indexing," Proc. of the 20th Int'l Conf. on Very Large Data Bases, 1994.
- [2] M. H. O'Docherty, "A Multimedia Information System with Automatic Content Retrieval," Technical Report Series UMCS-93-2-2, Dept. of Computer Science, Univ. of Manchester, England, 1993.
- [3] A. Yoshitaka, S. Kishida, M. Hirakawa, and T. Ichikawa, "Knowledge-Assisted Content-Based Retrieval for Multimedia Databases," Proc. of IEEE Int'l Conf. on Multimedia Computing and Systems, Boston, 1994.
- [4] Y. H. Gong, H. J. Zhang, H. C. Chuan, and M. Sakauchi, "An Image Database System with Content Capturing and Fast Image Indexing Abilities," Proc. of IEEE Int'l Conf. on Multimedia Computing and Systems, Boston, 1994.
- [5] J. K. Wu and A. D. Narasimhalu, "Identifying Faces Using Multiple Retrievals," IEEE Multimedia, Vol. 1, No. 2, Summer 1994.
- [6] M. J. Swain, "Interactive Indexing into Image Database," SPIE, Vol. 1908, San Jose, CA, 1993.
- [7] W. B. Pennebaker and J. L. Mitchell, JPEG Still Image Data Compression Standard, Van Nostrand Reinhold, New York, 1993.
- [8] W. I. Grosky, "Multimedia Information Systems," IEEE Multimedia, Vol. 1, No. 1, Spring 1994.
- [9] M. Sakauchi, "Database Vision and Image Retrieval," IEEE Multimedia, Vol. 1, No. 1, Spring 1994.
- [10] L. Teodosio and W. Bender, "Salient Video Stills: Content and Context Preserved," Proc. of First ACM Int'l Conf. on Multimedia, Anaheim, California, Aug. 1993.
- [11] S. W. Smoliar and H. J. Zhang, "Content-Based Video Indexing and Retrieval," IEEE Multimedia, Vol. 1, No. 2, Summer 1994.
- [12] R. M. Haralick and L. G. Shapiro, Computer and Robot Vision, Vol. 1, Addison Wesley, Reading MA, 1992.