Color Feature Embedded for Content-based Image Retrieval System

Putra Sumari, Hailiza Kamarulhaili
Department of Computer Science,
University Sciences of Malaysia,
11800 Minden, Penang, Malaysia
e-mail {putras, hailiza}@cs.usm.my

ABSTRACT

Retrieving images from a large image databases using image content as a key is an important issues. Due to the tremendous increase of digital images recently, there is an urgent need of image content retrieval to be processed quickly. The image feature extraction task as in the existing content-based image retrieval system contributes to high processing task during retrieval process. In this paper we present the new image format in which the color feature is embedded in the image. Color histogram data is stored in the high-level syntax of the image. The new format image has eliminated color extraction task as in existing content-based image retrieval. We have shown in this paper that readily color histogram in the image has reduced the processing task in the content-based image retrieval system.

1. Introduction

With the advances in computer technologies and the advent of the World Wide Web, there has been an explosion in the amount of digital image being generated. Many people access these data frequently for various purposes. The goal of content-based is to develop techniques that provide ability to store and retrieve images based on their contents[9]. Some of the applications of image retrieval are: multimedia information system, digital libraries, remote sensing and natural resources management, movie industry and video on demand [1]. Traditional databases use keywords as labels needs a large amount of manual processing and entails extra storage and the retrieval result may not be satisfactory. Several content-based image retrieval systems have been proposed in the literature [1]. A block schematic of a typical image archival and retrieval system is shown in figure 1. The major processing task

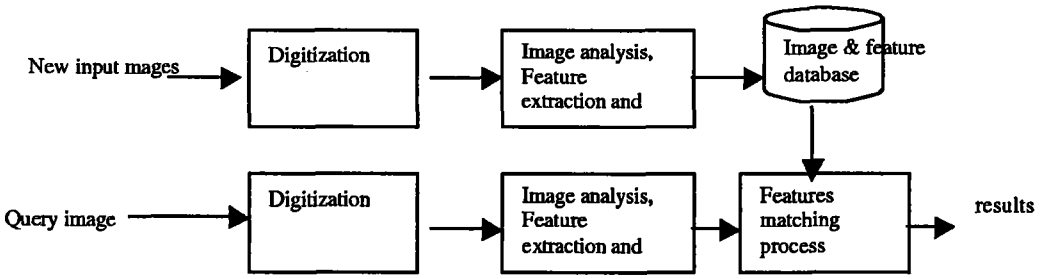

Figure1 : Schematic of an image archival and retrieval system

in content-based retrieval is feature extraction module. Feature extraction is a multidimensional feature vectors derived from color, texture, shape, sketch, others used for image comparison and similarities on the feature vectors of query image and to those image in the database. Many content-base retrieval works in literature such as QBIC, VisualSeek, are used color, texture and shape features for retrieving images. The processing of these features to huge amount of images can be length process. Therefore a fast and quick processing of feature extraction is needed in content-based image retrieval to handle data in the large database.

In this paper we look into the structure of the images especially the high-level syntax (image bit-stream) to embedded the feature inside it. The motivation is the processing of color extraction is no longer needed since the

color feature (we chose color histogram in our initial work) sit inside the image bit-stream. This gives new dimension to the existing architecture of content-based image retrieval. Since color histogram is embedded in the image then the feature extraction can be done quickly. The system just needed to peel the high-level syntax to obtain color histogram of the image. This approach has decreases the computational task of content-based image retrieval tremendously.

The organization of this paper is follows: section 2 presents the new image format. Section 3 present an experiment and results. In the experiment, we developed a new modified content-based image retrieval system to adapt the new image format. Finally section 4 is conclusion.

2 Image construction for content-based image retrieval

In this section we present a construction of a new image that color feature is embedded in the high-level syntax bit-stream of the image. The new image gives new dimension to the existing architecture of content-based image retrieval. Since color feature is embedded in the image then the feature extraction task as in the content-based image retrieval is eliminated. It decreases the computational task of content-based image retrieval. In our work, we have chose color feature (color histogram) as a candidate for embedding in the image syntax bit-stream simply due its simplicity. The following section, the color histogram computation and extending the high level of syntax of the image are presented.

2.1 Color histogram

We use color histogram [2, 7, 9, 10] for characterization image content. The color histograms represent the color distribution in an image by counting the number of occurrences of each unique color on the image. Since an image is composed of pixels and each pixel has a color, the color histogram of an image can be computed easily by visiting every pixel once. Color histogram of an image with color in the range $[0, L-1]$ is a discrete function $p(i) = n_i/n$, where i is the color of the pixel, n_i is the number of pixel in the image with color i, n is the total number of pixels in the image, and $i = 0, 1, 2, ..., L-1$. In general $p(i) = n_i/n$ gives an estimate of the probability of occurrence of color i. Figure 2 shows an example of image with its corresponding color histogram.

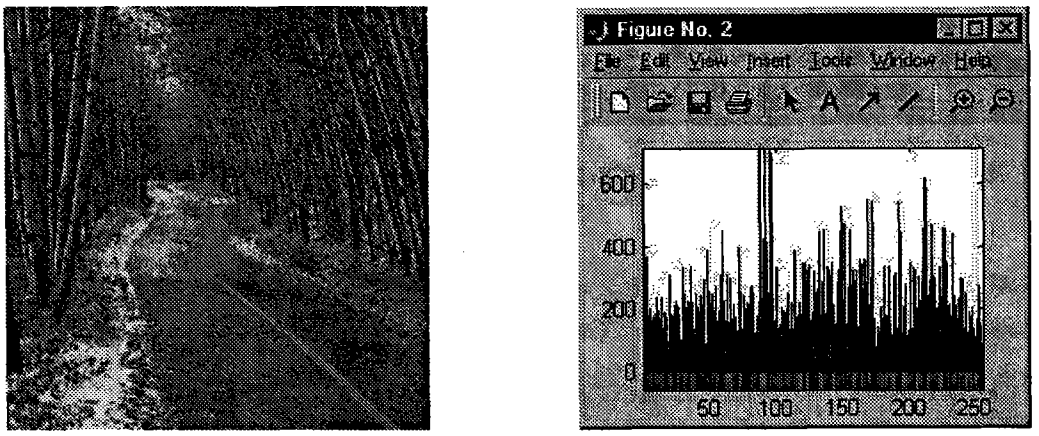

Figure 2: Image and its corresponding Histogram

2.2 Embedding color histogram in image syntax bitstream

Figure 3 shows the order of high-level constituent parts of the interchange file format. A more specific example using the image specified in CCITT Rec. T.81|ISO/IEC 10918-1. The part consists of file header directory, image data and indirect data as shown in figure 3a. The header then consists of many parameters (MN, HLEN, IDENT, and etc). These parameters represent values such as header length, version, id, image height, image width,

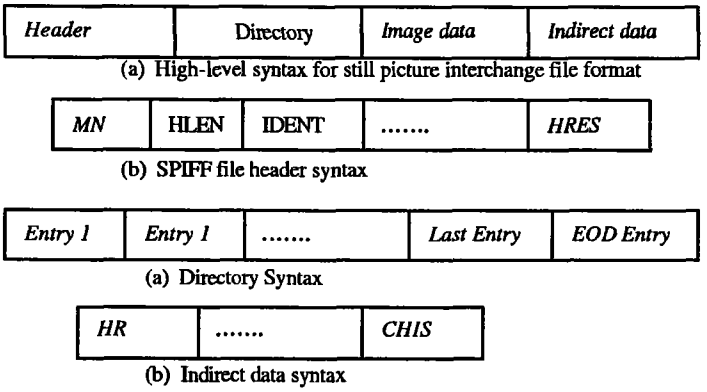

Figure 3 : High-level syntax for an image

compression type, vertical and horizontal resolution, color space, and etc. as shown in figure 3b. The directory component consists of parameters such as entry 1, 2, and etc. Finally the indirect data is an optional and consist of extra information of directory entry. We store the color histogram in the indirect data syntax. CHIS represents the histogram format.

3 Experiment and results

First of all we provided 100 images with embedded color histogram in its syntax bit stream. Then the image retrieval system was developed as shown in figure 4 using MatLab programming. The raw and query images are new extended format (described in section 2.2). The database system for manipulating images is SQL Server 7.0. Figure 4 show that to those images, we peeled the image syntax bitstream to pick the color histogram and store in the database. Given user image, the color histogram is peeled from the bitstream and pass to matching module.

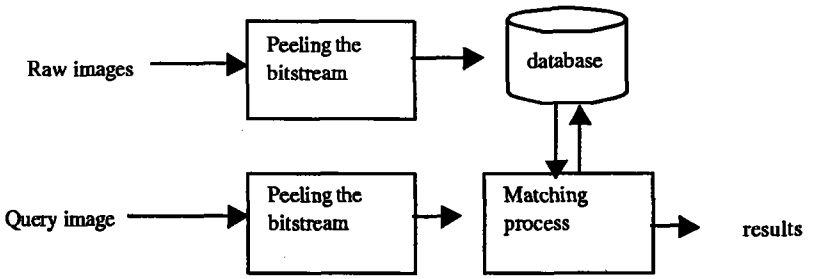

Figure 4: The modified content-based image retrieval system structure

The matching module compares the histogram of query image and histogram in the database. Comparison is made by histogram intersection. The intersection between histogram h and g is given by:

$$d_I(h,g) = \frac{\sum_{m=0}^{M-1} \min(h[m], g[m])}{\min(\sum_{m_0=0}^{M-1} h[m_0], \sum_{m_1=0}^{M-1} g[m_1])}$$

Figure 5 shows the results of query of image in the top. The results show that we managed to retrieve similar images to the query image. Our proposed method also eliminated some computational component in general content-based image retrieval.

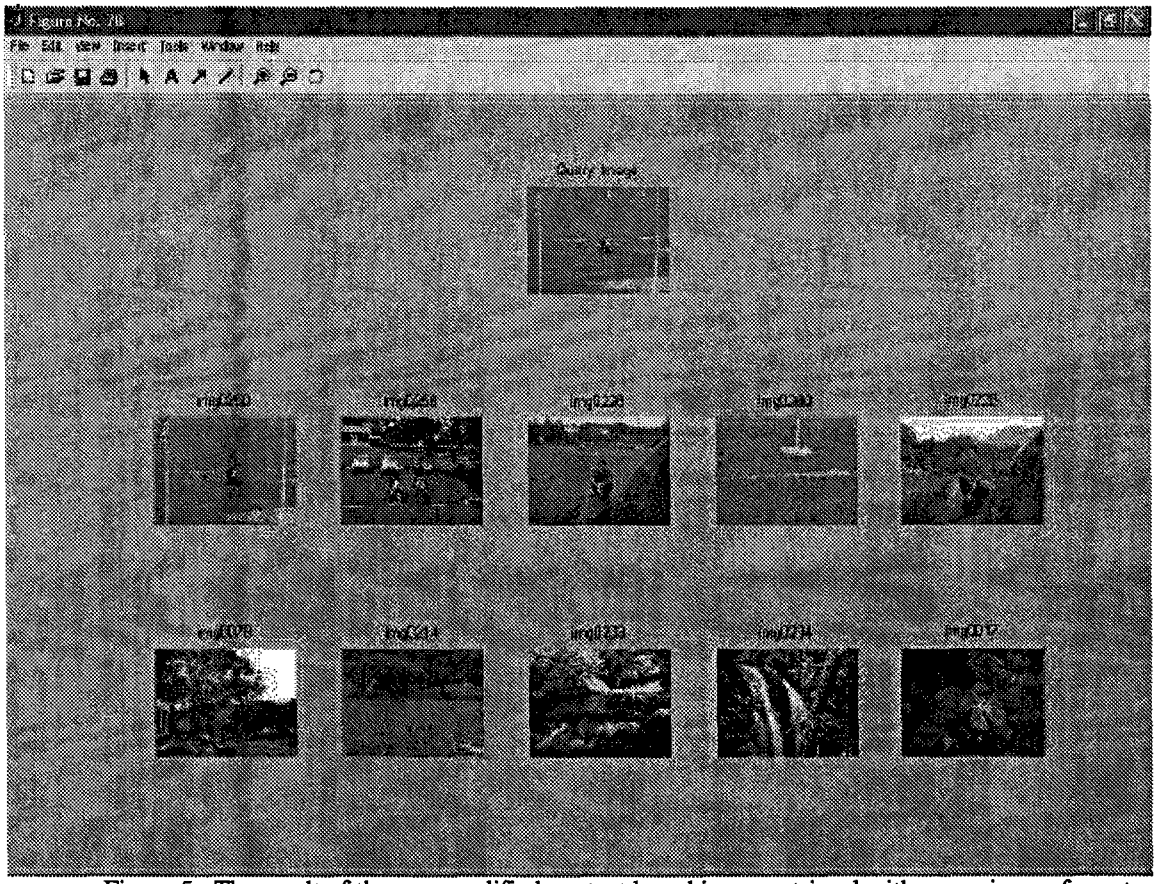

Figure 5 : The result of the new modified content based image retrieval with a new image format

4. Conclusion

In this paper we present a construction of a new image that color feature is embedded in the image. Color histogram is chose to be embedded in the high-level syntax of the image. The significant processing reduction is the retrieval system is color extraction task. Extraction of feature in existing system is eliminated. The system just needed to peel the high-level syntax of the image to obtain the color histogram data. We carried and experiment and we developed a new modified content-based image retrieval to adapt the new image format. Future work we will develop a unified form to handle color, texture, shape. We also look into the compress domain indexing techniques such wavelet , DCT and other signal processing in image retrieval.

5. References

[1] J. P. Eakins and M. E. Graham, "Content-based image retrieval: A report to the JISC Technology Applications Programme, Institute for Image Data Research, University of Northumbria at Newcastle, available online at http://www.unn.ac.uk/iidr/research/cbir/ report.html," January 1999.

[2] Y. Gong, C. Chuan, and G. Xiaoyi, "Image indexing and retrieval using color histograms," *Multimedia Tools and Applications*, vol. 2, pp. 133–156, 1996.

[3] Kim, I.-J., Lee, J.-H., Kwon, Y.-M & Park, S.H. (1997), Content-based image retrieval method using color and shape feature, in 'Proceedings of International Conference on Information, Communication and Signal Processing', IEEE, Singapore, pp. 948-952.

[4] M. A. Nascimento, E. Tousidou, V. Chitkara, and Y. Manolopoulos. Color based image retrieval using signature trees. In *Technical Report TR-01-02, Department of Computing Science, University of Alberta,* Edmonton, Alberta, Canada, 2001.

[5] The theory and practice of color - from a technical point of view. URL: http://xata.lviv.ua/~color/technical.htm.

[6] Thomas Seidl and Hans-Peter Kriegel, "Efficient User-Adaptable Similarity Search in Large Multimedia Databases," in Proceedings of the 23rd International Conference on Very Large Data Bases VLDB'97, Athens, Greece, August 1997.

[7] Shengjiu Wang, "A Robust CBIR Approach Using Local Color Histograms," Department of Computer Science, University of Alberta, Edmonton, Alberta, Canada, Tech. Rep. TR 01-13, October 2001.

[8] B.S. Manjunath, J.-R. Ohm, V.V. Vasudevan, A. Yamada, "Color and Texture Descriptors", IEEE Transactions on Circuits and Systems for Video Technology, Vol 11, No. 6, 2001, pp. 716-719

[9] Swain M. and Ballard D., 1991, Color Indexing. Int. J. Comput. Vis., 7 (1): 11-32

[10] Ioka M., 1998, A method of defining the similarity of images on the basis of color information. Technical Report RT-0030, IBM Research, Tokyo Research Laboratory.