Computer Graphics, Imaging and Visualization
School of Mathematical Sciences, USM (2004)

Removing Edge Defect in Color-based Image Retrieval

Putra Sumari, Mala Vili, Hailiza Kamarulhaili, Rosalina Abd. Salam
Multimedia Computing Research Group
School of Computer Science,
Universiti Sains Malaysia
Minden 11800 Penang
Email: {putras, mala, hailiza, rosalina}@cs.usm.my

Abstract

growth of digital image archive is increasing rapidly and the demand for effective and efficient tools for searching through large amounts of visual data is great. One of the contributing factors toward an efficient tool is the quality of images being handle. Due to this we have identified that many images in the database have black boundaries at the edge of the image in which significantly affects the values for extraction and similarity process. In this paper we devised an algorithm to remove this edge defect and yet make a good similarity measurement between two images. We use edge histograms to detect the edge defect and then remove it from color histogram for histogram intersection in similarity measurement. We show in our preliminary prototype experiment that free edge defect can produce better results.

1. Introduction

With the advances in computer technologies and the advent of the World Wide Web, there has been an explosion in the amount and complexity of digital data being generated, stored, transmitted, analyzed, and accessed. The use of images in human communication is common. Educators and writers use images for illustrations, designers and engineers use images for recording finished projects, doctors use images for diagnosis and monitoring, entertainers use images for making stories, newsmen use images for enhancing text information, to name just a few. An efficient searching and retrieval method indeed is an important issue for better image retrieval result of existing content-based image retrieval system.

We observed that many images in the database have black boundaries. We called it as edge defect and these black boundaries are perceptually insignificant (usually they are not even noticed as part of the image content) and yet they may significantly affect our values (e.g. it produces strong fake edges) during retrieval process. Specifically the color histogram is not represented the pattern of color feature of original image. This will affect the similarity color histogram comparison and yet degrade the final results. In this paper we devised an algorithm to remove this edge defect and yet make a good similarity measurement between two images. We use edge histograms to detect the edge defect and then remove it from the final color histogram for histogram intersection in similarity measurement. In our prototype we have shown that the absence of edge defect as within image can produce better results.

The organization of this paper is as follows: section 2 presents related studies. Section 3 presents the new algorithm to remove the edge defect. Section 4 presents the prototype architecture system the finding. Finally section 5 is the conclusion.

2. Related studies

In order to access the huge amount of information there should be means to browse, search, and retrieve the images. There have been two major approaches in the study of image retrieval system [1, 7, 12. The traditional approach is to annotate each image manually with text in which describes the content of the image; some image attributes such as number and data can also be included in the annotation. Another approach is to index images directly based on the visual image content such as QBIC, Visual Seek [8, 12], etc. There are several color spaces exist in image retrieval system. The HSV color space was developed to be more intuitive in manipulating color and was designed appropriately the way humans perceive and interpret color. Three characteristics of color: hue, saturation and value are defined to distinguish color components. Hue describes the actual wavelength of the color by representing the color name, for example: green, red or blue. Saturation is a measure of the purity of a color. For instance, the color red is 100% saturated color due to the amount of white color. Value embodies the lightness of the color. It ranges from black to white.

There are two types of color histograms [2, 3, 4, 5, 9, 10, 11]: Global color histograms (***GCH***s) and Local/region color histograms (***LCH***s). A GCH represents one whole image with a single color histogram. An LCH divides an image into fixed blocks and takes the color histogram of each of those blocks. LCHs contain more information about an image but are computationally expensive when comparing images. Both techniques are very useful for retrieval of images but are suited for different types of queries. However, since the LCH only compares regions in the same location, when the image is translated or rotated, it does not work well. Color indexing by global distribution is most useful when user provides a sample image for the query.

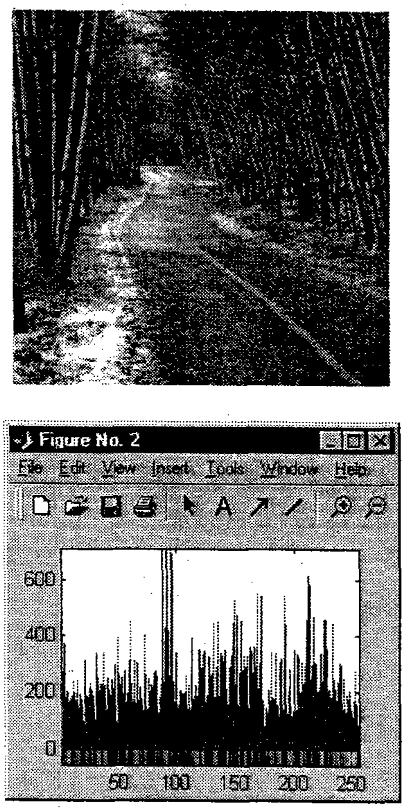

Figure 1: Sample image and its corresponding histogram

Edge defect is one of the criteria that has not been focused in this field. Several works identified this as black boundaries resulted from the transform task in the compression stage. Most of the works mentioned the edge defect and has skewed their experiments work.

3. Removing edge defect for similarity measurement

The main and most common method of representing color information of images in content-based retrieval system is through color histograms. A color histogram is a type of bar graph, where each bar represents a particular color of the color space being used. In MatLab for example you can get a color histogram of an image in the RGB or HSV color space. The bars in a color histogram are referred to as bins and they represent the x-axis. The number of bins depends on the number of colors there are in an image. The y-axis denotes the number of pixels there are in each bin as shown in figure 1.

3.1 Edge histogram

Many images in the database have black boundaries. These boundaries are perceptually insignificant (usually they are not even noticed as part of the image content) and yet they may significantly affect our values (e.g. it produces strong fake edges) in representing its color histogram. This insignificant boundary will be removed in feature extraction task so that it can give more accurate similarity measurement of images. The percentage of image with free edge defect will be in the range of 5%~95% in each dimension of every image. We first detect the edge defect of the image. We use a 4-bin edge histogram to represent the strength of the edge in 0, pi/4, pi/2, 3*pi/4 directions.

1) Image gradients Gx and Gy are computed using sobel operators;
   sobel_y = [1 0 -1; 2 0 -2; 1 0 -1];
   sobel_x = [1 2 1; 0 0 0; -1 -2 -1];
2) Compute edgemap with sobel operator using matlab function edge.m, the cut-off percentage is determined using an RMS estimate of image noise
3) For each edge pixel, compute the edge direction ***theta* = *atan(Gy/gx)***;
4) Edge direction ***theta*** is then uniformly quantized to 4 bins (*0, +/-45, 90 degrees*) using the decision values *+/- 22.5, +/-67.5* degrees.

Finally the edge histogram is then normalized w.r.t. the image size, i.e. each bin value represents the percentage of a certain edge direction in an image. Then it will use later in the histogram intersection to eliminate the edge defect.

3.2 Histogram intersection and similarity measurement

The aim of a content-based retrieval system is to retrieve images from a repository that are most similar to the user's perception using a similarity measurement. The distance metric can be termed as similarity measure, which is the key component in content-based image retrieval. It is important to explore different similarity measures to find the best distance metric for content based image retrieval.

In the Histogram Intersection technique, two normalized histograms are intersected as a whole, as the name of the technique implies. The similarity between the histograms is a floating point number between 0 and 1. Equivalence is designated with similarity value 1 and the similarity between two histograms decreases when the similarity value approaches to 0. Both of the histograms must be of the same size to have a valid similarity value.

Let *H1 [1...n]* and *H2 [1...n]* denote two histograms of size n, and $S_{H1,H2}$ denote the similarity value between H1 and H2. Then, $S_{H1,H2}$ can be expressed by the distance between the histograms H1 and H2 as:

$$S_{H_1,H_2} = \frac{\sum_i^n \min(H_1[i], H_2[i])}{\min(|H_1|, |H_2|)}.$$

We use Minkowski-form distance metric [6] to eliminate the edge defect derived from section 3.1. It is a simple distance metric involving the subtraction of the number of pixels in the edge histogram to the color histogram of query image.

4. Experimental results

Figure 2 shows the outline of the prototype system in our work. The system has four main components namely image database component, feature extraction component, feature database component and similarity measurement component. The image database component contains 8-bit uncompressed bit maps ***BMP***s that have been randomly selected from the World Wide Web. The database system for manipulating images is SQL server 2000. The image files was sequentially named and saved. Each image will have unique ID. User can get all the images available in the database in the user interface as shown in figure 3. The database consists of 300 images. The images are approximately 192 X 128 pixels in size. The feature extraction module, we chose HSV color space because of its similarity and perceptibility. Color histogram is created in this component. The color histogram is freed from edge defect of the image. The feature database component is simply storing all image color features with its corresponding original image in the database. The similarity measurement component is responsible for comparing color histograms of two images and able to detect the most similarity one. The highest histogram intersection is used in deciding the best results.

Figure 4 shows the result of the query image ID 174.png (top left corner of figure 4) and we have extra one image (the fourth image from the left of figure 4) resulted from our prototype. Our system has capable to retrieve another extra image and this is due to the absence of black boundary of the query image. We can see that the fourth image has little amount of black color that resulted form the absence of black defect in the query image and in similarity measurement task.

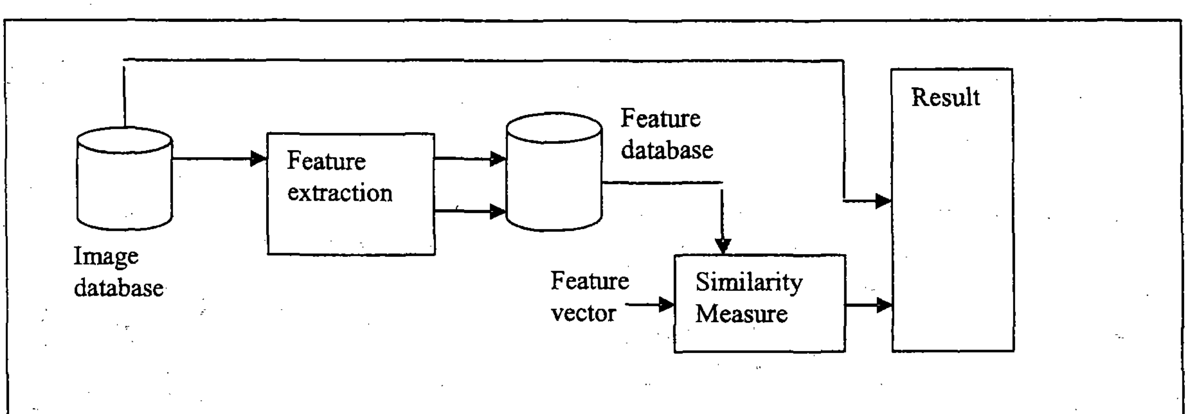

Figure 2: The architecture of the prototyped.

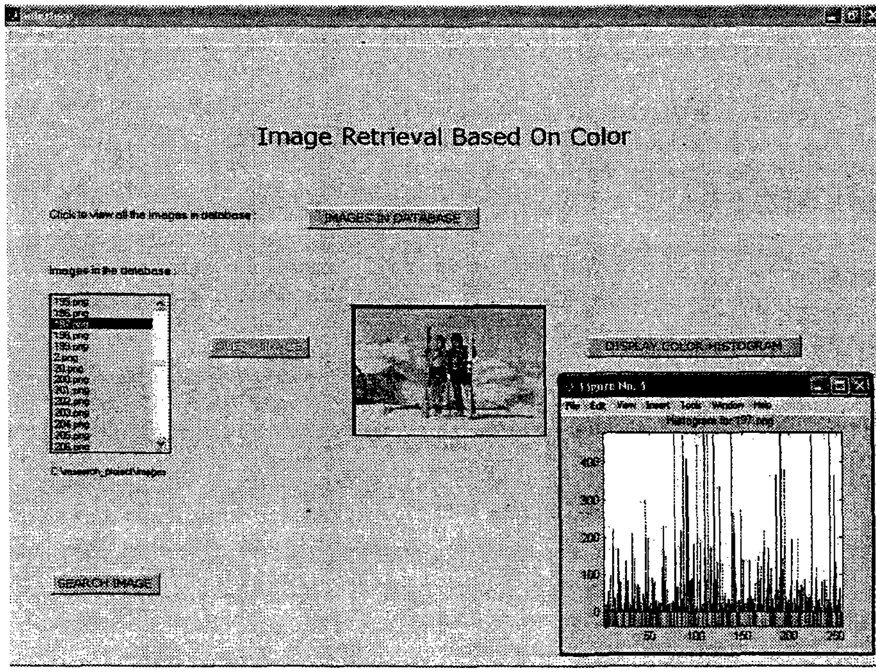

Figure 3: Query Interface and Relevant Histogram for the Query Image

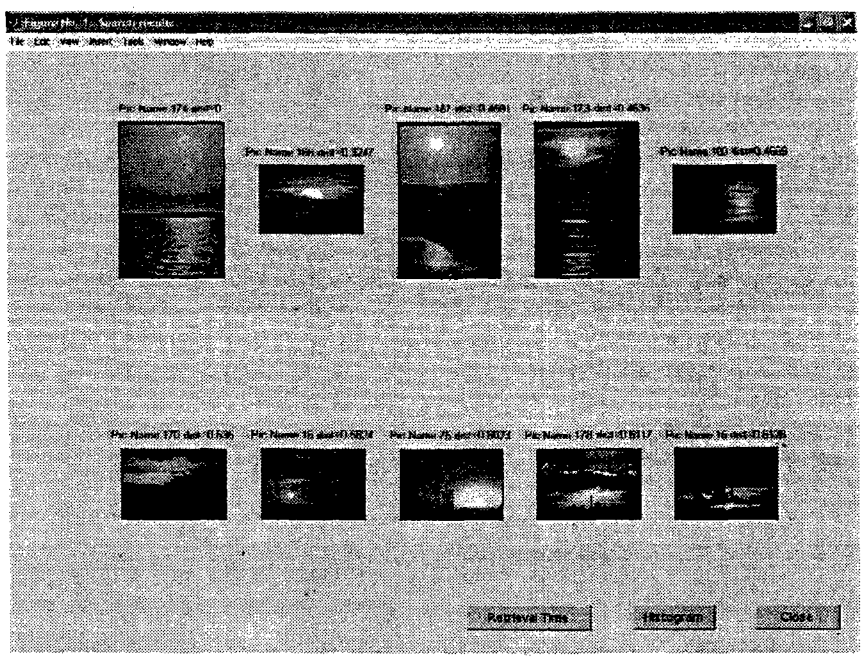

Figure 4: Query Results for Image 174.png

5. Conclusion

In this paper we present our prototype of image retrieval system. The prototype has a capability of removing the edge defect in the image during the similarity process. We observed that many images in the database have black boundaries. This we referred as edge defect and it has significantly affects our values (e.g. it produces strong fake edges) during color histogram intersection in similarity measurement process. We devised an algorithm to remove this edge defect and yet make a good similarity measurement between two images. We use edge histograms to detect the edge defect and then remove it from the final color histogram for histogram intersection in similarity measurement. We show in our prototype experiment that the absence of edge defect within image can produce better results.

6. 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 cations*, vol. 2, pp. 133–156, 1996.

[3] ..., 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] Barbeau Jerome, Vignes-Lebbe Regine, and Stamon Georges, "A Signature based on Delaunay Graph and Co-occurrence Matrix," Laboratoire Informatique et Systematique, Universiyt of Paris, Paris, France, July 2002, Found at: http://www.math-info.univ-paris5.fr/sip-lab/barbeau/barbeau.pdf

[7] 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, http://www.vldb.org/conf/1997/P506.PDF

[8] 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, Found at: http://citeseer.nj.nec.com/wang01robust.html

[9] 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

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

[11] 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.

[12] Niblack W., Barber R., and et al., 1994, "The QBIC project: Querying images by content using color, texture and shape", In Proc. SPIE Storage and Retrieval for Image and Video Database.