

Color Descriptor for Image Retrieval in Wavelet Domain

Ariya Utenpattanant
Faculty of Engineering,
King Mongkut's Institute of
Technology Ladkrabang,
Bangkok, Thailand
ariyau@yahoo.com

Orachat Chitsobhuk
Faculty of Engineering,
King Mongkut's Institute of
Technology Ladkrabang,
Bangkok, Thailand
kcoracha@kmitl.ac.th

Amnach Khawne
Faculty of Engineering,
King Mongkut's Institute of
Technology Ladkrabang,
Bangkok, Thailand
kkamaj@kmitl.ac.th

Abstract — This paper presents an approach to manage a large database using a compact color descriptor and a statistical method for accurately pruning the database. A compact color descriptor adopted in the proposed content-based image retrieval system is 63-bit binary Haar color histogram, which is very compact and can be effectively used for fast image search. In addition to fast searching using this compact descriptor, we further improve retrieval time by applying pruning technique, which looks for the candidate images similar to the query image from the database and ignore the rest that are not likely to the query image. The descriptors of the candidate images are then matched with that of the query. The most similar images will be retrieved and ordered according to their distance to the query. The proposed retrieval system can efficiently retrieve the most similar images from the database while can help reducing the retrieval time and the storage space.

Keywords — Compact descriptor, image retrieval, pruning.

1. Introduction

As the availability of images has been significantly increasing, this creates a need for efficiently organizing and managing of image storage systems. Several low-level image content descriptions are proposed [1] [2]. Among the basic low-level features used to index each image in the database, color is the most effective feature for representing visual information since it is local and independent of view and resolution. Color histogram is the most widely used color descriptor to represent the color content of an image [3]. However, it is not suitable for some tasks in image such as search, filtering and retrieval since color histogram need a large number of bits to represent color information which becomes a difficult problem and requires large amount of processing time in matching images.

This paper presents an image retrieval system using a compact color descriptor obtained from Haar transform coefficient of the color histogram [4]. Each coefficient is quantized to a binary number. This results in a color descriptor of 63 bits long, which is more compact than regular color histogram. The binary nature of the descriptor allows simple matching metric based on Hamming distance. This compact descriptor is useful for applications where the

computational resources are limited and there is a need to stream the descriptors through a bandwidth-limited channel. In addition, to further reduce the searching time, the hierarchical search using statistical pruning technique is proposed in this paper. The objective of pruning is to remove images from the database that are not likely to be of interest to the user while retaining all or most potentially candidate images [5]. This is useful since the reduced size of the pruned database makes it easier and more effective to search through the database. Then, the descriptors of the candidate images are compared to the descriptor of the query. The most similar candidate images are retrieved and ordered according to their distances to the query.

The organization of this paper is as followed. Section 2 presents a process to generate a compact binary Haar color descriptor. Section 3 details the proposed image retrieval system followed by experimental results in Section 4. Finally, summarize and conclusion is presented in Section 5.

2. Binary Haar descriptor

In this paper, a compact color descriptor called binary Haar descriptor [4] is adopted as an index of an image.

The algorithm used to compute the binary Haar descriptor can be expressed as followed

2.1 Obtain the color histogram using a suitable color space and quantization table. The histogram can be one, two or three dimensional histogram. (In this paper, one dimensional uniformly quantized HSV color histogram is adopted.)

2.2 Compute the Haar transform coefficients of the histogram by using Haar wavelet function (1). Since the Haar wavelet function [6] contains only values of +1 or -1. Therefore, the computation does not involve any multiplication.

$$\psi(x) = \begin{cases} 1 & 0 \leq x < 0.5 \\ -1 & 0.5 \leq x < 1 \\ 0 & \text{elsewhere} \end{cases} \quad (1)$$

The Haar transform coefficients are obtained by taking the inner product of the basis functions with the given histogram. For example, the second coefficient is positive if the sum of

the left half of histogram bins is greater than the right half and negative otherwise (from the basis function 2 in figure 1). Similarly, the third coefficient is positive if sum of first quarter of histogram bins is greater than the second quarter and negative otherwise (from the basis function 3 in figure 1). Also note that the first coefficient is always positive. Therefore, this coefficient is not used in similarity matching.

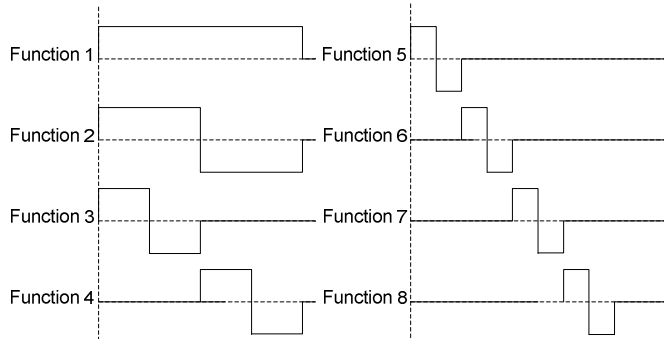


Figure 1. The basis functions for Haar wavelet transform of input 8 bits long

2.3. After that each of Haar transform coefficients is quantized to binary, the Haar coefficient is quantized to '1' if it is positive and is quantized to '0' otherwise.

The Haar transform coefficients are hierarchically computed. The first level, the 64 bins of histogram is divided into two halves. If the sum of the histogram values in the left half is greater than the sum of the histogram values in the right half then the first bit of descriptor is '1' else it is '0'. The second level, the 64 bins of histogram is divided into four halves. If the sum of histogram values in first(third) half is greater than the sum of histogram values in second(forth) half then the second(third) bit of descriptor is '1' else it is '0'. This is repeated recursively at third, forth, fifth and sixth levels resulting in 4,8,16 and 32 coefficients. Therefore, the 63(1+2+4+8+16+32) bits binary Haar descriptor is obtained. After calculating the binary Haar histogram, this descriptor will be used as an index of each image.

3. The Proposed Image Retrieval System

The proposed image retrieval system is presented in Fig. 3. To retrieve the similar images compared to the query from the database, first, a binary Haar histogram of the query image is computed. Then, the candidate search is performed using pruning technique. The objective of pruning is to remove images from the database that are not likely to be of interest to the user while retaining all or most potentially desirable images [5]. We look for some specific bits in the binary Haar histogram to prune the database. The specific bits from the 32nd - 63rd bits of binary Haar histogram, in the sixth level of the Haar transforms, are computed by comparing the histogram value of adjacent color histogram bin. For example, the 32nd bit is computed by comparing the color histogram value of 1st bin with 2nd bin. The bit 33rd is computed by comparing the color histogram value of 3rd bin with 4th bin and so on as shown in Fig 2.

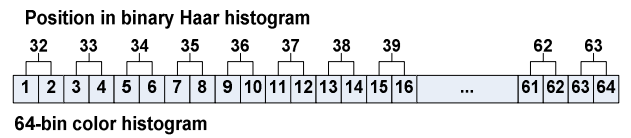


Figure 2. The bits 32nd - 63rd of binary Haar histogram

Bits used for filtering the candidate images are selected based on the most important color in the query image as followed

Let H is the binary Haar color histogram and j is the j^{th} bin of the binary histogram while i is the i^{th} bin having maximum histogram value.

$$i = \arg(j) \max(H(j)) \quad (2)$$

The rank (i) of bin which has maximum histogram value is used to define which bit in the descriptor is fixed. For example, the 32nd bit of descriptor is fixed to '1' if the rank of bin is the first bin or fixed to '0' if the rank of bin is the second bin. The 33rd bit of descriptor is fixed to '1' if the rank of bin is the third bin or fixed to '0' if the rank of bin is the forth bin and so on. The images in the database having the same specific bits as those of the query are defined as the candidate images. The binary Haar histogram of each candidate is then compared with that of the query. The distance between each candidate and the query is calculated using the Hamming distance. The Hamming distance is computed by taking the XOR of the two 63-bit binary Haar descriptors and counting the number of '1' bits in the result. The smaller value of the result, the closer the image to query image. The most similar images will be retrieved and ordered according to their distance to the query.

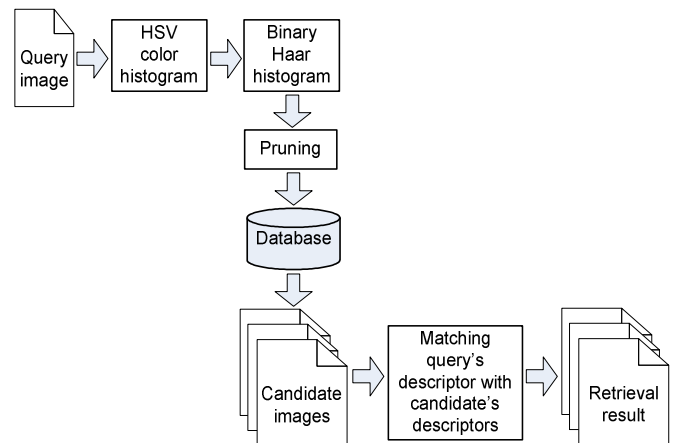


Figure 3. The process to retrieve image from the database

4. Experimental results

The image retrieval experiment using the binary Haar descriptor was performed using a database of 2,997 images. The images are chosen from many different sources [7] and can be categorized into 7 groups (e.g., human, animal, natural

view, etc.). The binary Haar descriptors of images in database are pre-computed and stored. In this experiment, we use a 64-bins color histogram in HSV color space. Figure 4 shows the process to compute the binary Haar descriptor then stored in the database.

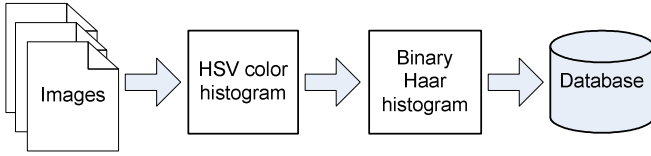


Figure 4. The process to compute the binary Haar descriptor

When an image is selected as the query image, its 64-bins color histogram is calculated. The histogram of query image is transformed to the binary Haar descriptor. The rank (i) of bin which has maximum histogram value is used to define which bit in the descriptor is fixed as described in section 3.

Searching images from the database as candidate images is performed by considering these bit positions and values as the criteria. As a result, some images in database are pruned. After that, the descriptor of query image is compared to the descriptor of the candidate images using the Hamming distance. The retrieval results are ordered as ascending sort and presented to the user. The retrieval performance using the pruning technique is as efficient as the performance of matching the descriptor of the query to the descriptors of all the images in the database. Moreover, the proposed image retrieval system can reduce the retrieval time while saving storage space. Figure 5 shows the sample results of 2 queries.



(a) Query image I



(b) A result of image retrieval using binary Haar descriptor for Query image I



(c) A result of image retrieval using binary Haar descriptor with pruning technique for query image I



(d) Query image II



(e) A result of image retrieval using binary Haar descriptor for Query image II



(f) A result of image retrieval using binary Haar descriptor with pruning technique for query image II

Figure 5. Sample results of image retrieval

To evaluate the performance of the proposed image retrieval system, we use an objective measure called ANMRR suggested by MPEG-7 [4]. A set of ground truth images that are most relevant to query were identified. The query and ground truth sets were chosen manually. For each query a set of ground truth images were specified a rank. This measure combines the precision and recall measure to obtain a single objective value.

The average rank $AVR(q)$ for query q is computed as follows:

$$AVR(q) = \sum_{k=1}^{NG(q)} \frac{Rank(k)}{NG(q)} \quad (3)$$

where

- $NG(q)$ is the number of ground truth images for query q
- $Rank(k)$ is the retrieval rank of the ground truth image.
- K is the top ranked retrievals are examined where $K = \min(4 * NG(q), 2 * GMT)$ and $GMT = \max\{NG(q)\}$ for all q 's of a data set.
- The ground truth images are not in the first K retrievals is assigned to $(K+1)$

The modified retrieval rank is computed as follows:

$$MRR(q) = AVR(q) - 0.5 - \frac{NG(q)}{2} \quad (4)$$

Note that $MRR = 0$, when the $NG(q)$ ground truth images are the top retrievals.

The normalized modified retrieval rank is computed as follows:

$$NMRR(q) = \frac{MRR(q)}{K + 0.5 - 0.5 * NG(q)} \quad (5)$$

Then compute average of $NMRR$ over all queries

$$ANMRR = \frac{1}{Q} \sum_{q=1}^Q NMRR(q) \quad (6)$$

The average normalized modified retrieval rank $ANMRR$ is used as the objective measure to evaluate the retrieval

performances. The *ANMRR* not only indicates how many of the correct items are retrieved, but also how highly they are ranked among the retrieved items. The *ANMRR* is always in range of 0 to 1 and smaller the value of this measure is the better quality. Table1 shows the *ANMRR* and the reducing time from the experiment (50 set of query images and ground truth images).

Table 1. The comparison of performances and reducing time

	ANMRR	% Reducing time
Binary Haar descriptor	0.41	-
Binary Haar descriptor with pruning	0.44	46.69

From the results, we have proved that using a compact binary Haar descriptor with statistical pruning technique gives the retrieval performance as close as without any pruning technique while can save searching time by 46.69 %.

5. Conclusion

This paper presents the image retrieval system using a binary Haar descriptor and pruning search algorithm. The proposed retrieval system performs candidate search using pruning technique, which considering the bit position and value corresponding to maximum histogram value.

Then, the binary Haar descriptors of these candidate images will be compared with that of the query using hamming distance. The most similar images will be retrieved and ordered according to their distance to the query. The experiments show that the proposed retrieval system can efficiently retrieve the most similar images from the database while can help reducing the retrieval time and the storage space.

REFERENCES

- [1] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, "Efficient and effective querying by image contents," *J. Intell. Inform. Syst.*, vol. 3, 1994, pp. 231–262.
- [2] M. Swain and D. Ballard, "Color indexing," *Int. J. Comput. Vis.*, vol. 7, no. 1, 1991, pp. 11–32.
- [3] B. S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan, and Akio Yamada, "Color and Texture Descriptors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, 2001, pp. 703–715.
- [4] S. Krishnamachari, and M. Abdel-Mottaleb, "Compact color descriptor for fast image and video segment retrieval", *Proceedings of IS&T/SPIE Storage and Retrieval of Media Databases 2000*, San Jose CA, Jan. 2000.
- [5] J-Y. Chen, C.A. Bouman, and J. Dalton, "Active Browsing using Similarity Pyramids", *SPIE/ IS&T conference on Storage and Retrieval for Image and Video Databases VII*, San Jose CA, Jan. 26-29 1999, Vol. 3656, pp. 144-154.
- [6] Rafael C. Gonzalez, and Richard E. Woods, "Digital Image Processing", Prentice Hall, 2001.
- [7] Britton I. "Free Pictures" [Online]. <http://www.freefoto.com> . 2002.