Using Intrinsic Object Attributes for Incremental Content Based Image Retrieval with Histograms

Jongan Park, Seungjo Han, and Pankoo Kim

Dept of Information & Communications Engineering,
Chosun University, Gwangju, Korea
japark@chosun.ac.kr

Abstract. An incremental Content Based Image Retrieval (CBIR) method is proposed in this paper. This method is based on color histogram. Standard histograms, because of their efficiency and insensitivity to small changes, are widely used for content based image retrieval. We define an algorithm that utilizes the concept of Histogram Refinement [1] and we call it Color Refinement Method. Color Refinement method splits the pixels in a given bucket into several classes just like histogram refinement method. The classes are all related to colors and are based on color coherence vectors. After the calculation of clusters using color refinement method, inherent features of each of cluster is calculated. These inherent features are used for incremental CBIR.

1 Introduction

Content-based image retrieval (CBIR) is regarded as one of the most effective ways of accessing visual data. There are queries that require the comparing of the images on their overall appearance. In such cases, color histograms can be employed because they are very efficient regarding computations. Plus they offer insensitivity to small changes regarding camera position. But the main problem with color histograms is their coarse characterization of an image. That may itself result in same histograms for images with different appearances. We have also proposed color histogram in our algorithm. However, this paper is different because we try to exploit the colors of individual objects.

2 Related Work

Work by Arnold et. al. [2] is an excellent review of content based image retrieval till 2000. Hsu [3] exploits the degree of overlap between regions of the same color. They used a database of 260 images. Paisarn [4] proposed an unsupervised learning network to incorporate a self-learning capability into image retrieval systems. Smith & Chang's method also partitions the image. Histogram back-projection method [5] is used for back projecting set of colors onto the image.

Rickman and Stonham [6] provide a method based on small equilateral triangles with fixed sides. Djeraba [7] tried to add the generalization capability for indexing and retrieval. Stricker and Dimai [8] finds the first three moments of the color

distributions in an image. Huang et al. [9] method is called Color Correlogram and it captures the spatial correlation between colors. Pass and Zabih [1] method called Histogram Refinement partitions histogram bins by the spatial coherence of pixels. Jong-An, Bilal et al. [10] provided shape description based on histogram based chain codes. Choi et. al. [11] proposed SOM based R*-Tree as new indexing method for high dimensional features vector. Xin Huang et. al. [12] presented image retrieval methodology based on color and spatial feature.

3 Feature Extraction Algorithm

3.1 Pre-processing

At the pre-processing stage, the image is converted to grayscale image using threshold. The grayscale image is then quantized to 4 levels from 256 levels, in order to increase the computational speed. Hence, we get 4 bins (levels) after quantization. We used uniform quantization which provides us with 4 separate bins with equal range. The steps in the pre-processing stage can be observed from first 3 blocks in figure 1.

3.2 Features from Quantized Bins

We use color refinement method for feature extraction from the quantized bins based on histogram refinement [1] method. The histogram refinement method provides that the pixels within a given bucket be split into classes based upon some local property and these split histograms are then compared on bucket by bucket basis.

Color histogram buckets are partitioned based on spatial coherence just like computed by Pass and Zabih [1]. A pixel is coherent if it is a part of some sizable similar colored region, otherwise it is incoherent. So the pixels are classified as coherent or incoherent within each color bucket. If a pixel is part of a large group of pixels of the same color which form at least one percent of the image then that pixel is a coherent pixel. Otherwise it is incoherent pixel and the group is incoherent group or cluster.

Then two more properties are calculated for each bin. First the numbers of clusters are found for each case, i.e., coherent and incoherent case in each of the bin. Secondly, the average of each cluster is computed. So for each bin, there are six values: one each for percentage of coherent pixels and incoherent pixels, number of coherent clusters and incoherent clusters, average of coherent cluster and incoherent cluster. These values are calculated by computing the connected components. A connected component C is a maximal set of pixels such that for any two pixels $p, p' \in C$, there is a path in C between p and p' . A pixel is classified as coherent if it is part of a connected component whose size is equal to or greater than τ ($\tau = 5\%$ of the image size). Otherwise it is classified as incoherent.

For each discretized color j , let the number of coherent pixels as α_j , the number of coherent connected components as C_{α_j} and the average of coherent connected component as μ_{α_j} . Similarly, let the number of incoherent pixels as β_j , the number of incoherent connected components as C_{β_j} and the average of incoherent connected component as μ_{β_j} . For each discretized color j , the total number of pixels are $\alpha_j + \beta_j$ and the color histogram summarizes the image as $\langle \alpha_1 + \beta_1, \dots, \alpha_n + \beta_n \rangle$.

3.3 Features from Coherent Clusters

The additional features at this stage are based on the coherent clusters only. We select four features namely, Size of largest cluster in each bin; Size of median cluster in each bin; Size of smallest cluster in each bin; and Variance of clusters in each bin.

Let us denote the largest cluster in each bin as L_{oj} , the median cluster in each bin as M_{oj} , the smallest cluster in each bin as S_{oj} and variance of clusters in each bin as V_{oj} . Since there are 4 bins, so we get additional 4 features for each bin. Therefore, a total of 40 additional features are considered for image retrieval. Considering section 3.2, initially 6 features per bin are selected for image retrieval and later 4 additional features per bin are considered in this section for refining result of image retrieval.

4 Incremental Image Retrieval Approach

Image retrieval is done in two stages hence the name incremental image retrieval approach. In the first stage, the features defined in section 3.2 are considered for retrieval while in stage 2; the features defined in section 3.3 are considered. The first stage gives us a coarse result while stage 2 refines the result obtained in stage 1. Therefore, result is relevant and accurate image retrieval from image databases.

4.1 First Level Retrieval

The features obtained in section 3.2 are used for retrieval at first level. We use the L_1 distance to compare two images I and I' .

$$\Delta_1 = |(\alpha_j - \alpha'_j)| + |(\beta_j - \beta'_j)| \quad (1)$$

$$\Delta_2 = |(C_{oj} - C'_{oj})| + |(C_{\beta j} - C'_{\beta j})| \quad (2)$$

$$\Delta_3 = |(\mu_{oj} - \mu'_{oj})| + |(\mu_{\beta j} - \mu'_{\beta j})| \quad (3)$$

In scheme [1], only equation (1) is used and following is used for comparison:

$$\Delta = |(\alpha_j + \beta_j)| - |(\alpha'_j + \beta'_j)| \quad (4)$$

4.2 Second Level Retrieval

This level of retrieval is used for further refining the result obtained in section 4.1. The additional features obtained in section 3.3 are used at this level of retrieval. Again we use the L_1 distance to compare two images I and I' . Using the L_1 distance, the j th bucket's contribution to the distance between I and I' is:

$$\Delta_4 = |(L_{oj} - L'_{oj})| \quad (5)$$

$$\Delta_5 = |(M_{oj} - M'_{oj})| \quad (6)$$

$$\Delta_6 = |(S_{oj} - S'_{oj})| \quad (7)$$

$$\Delta_7 = |(V_{oj} - V'_{oj})| \quad (8)$$

Fig. 1. Block diagram of the feature extraction algorithm

5 Results and Discussion

The proposed algorithm is tested with the database provided by James S. Wang et. al [14]. After the first the images were preprocessed and converted to grayscale images. Then the images were quantized and the features described in section 3.2 were calculated. Also, the features described in section 3.3 were calculated. These features were stored with each of the images. Figure 2 shows one of the image from the database, its corresponding grayscale image and then the corresponding quantized images.

Consider table 1. Table 1 provides the parameter values related with the incoherent clusters. The parameters include percentage of incoherent pixels (β_j), number of incoherent clusters ($C_{\beta j}$) and average of incoherent cluster ($\mu_{\beta j}$) for each j th bucket or bin. As an example, we show the results for 4 bins of one of the images in table 1. Figure 3 shows the corresponding incoherent clusters.

Fig. 2. One of the image from the database, converted to grayscale & quantized

Table 1. Example of parameter values for Incoherent pixels

	β_i	$C_{\beta i}$	$\mu_{\beta i}$
Bin 1	0.78%	38	1.1053
Bin 2	7.02%	64	5.8438
Bin 3	31.02%	86	19.209
Bin 4	61.18%	105	31.048

Fig. 3. Incoherent clusters in 4 different bins

Consider table 2. Table 2 provides the parameter values related with the coherent clusters. The parameters include percentage of coherent pixels (α_j), number of coherent clusters ($C_{\alpha j}$) and average of coherent cluster ($\mu_{\alpha j}$) for each j th bucket or bin. As an example, we show the results for 4 bins of one of the images in table 2.

Consider table 3. Table 3 provides the additional parameter values related with the coherent clusters. The parameters include size of largest cluster in each bin ($L_{\alpha j}$), size

Table 2. Example of parameter values for coherent pixels

	α_j	$C_{\alpha j}$	$\mu_{\alpha j}$
Bin 1	0	0	0
Bin 2	50.61%	2	26689
Bin 3	41.68%	4	10990
Bin 4	7.71%	3	2712

Table 3. Additional parameter values for coherent pixels

	$L_{\alpha j}$	$M_{\alpha j}$	$S_{\alpha j}$	$V_{\alpha j}$
Bin 1	0	0	0	0
Bin 2	51606	0	1772	1.24E+09
Bin 3	14553	12637	2340	34021226
Bin 4	4996	2025	1115	4119517

of median cluster in each bin (M_{aj}), size of smallest cluster in each bin (S_{aj}) and Variance of coherent clusters in each bin (V_{aj}). As an example, we show the results for 4 bins of one of the images in table 3.

The results were compared with the L_1 distance as described in section 4. First, we used equation (1) to equation (3) for identifying the similarity between images. Then we used equation (5) to equation (8) to further refine the results.

The figure 4 shows the query images and the first 3 results obtained by using the above described algorithm. On inspection of all the images of the database, we found that this was the closest result. Similar query results were obtained for various query images.

Fig. 4. Image Retrieval from the database

6 Conclusions

In this paper, we have proposed an algorithm that is based on color histogram. We have shown that the features obtained using this algorithm are quite useful for relevant image retrieval queries. The feature selection is based on the number, color and shape of objects present in the image. The grayscale values, mean, variance and various sizes of the objects are considered as appropriate features for retrieval. These

features are independent of image orientation. Color refinement method takes care of the color as well as the spatial relation feature. The shape features are extracted in section 3.3 from the color based features defined in section 3.2. This algorithm works well in space domain. We plan to extend it to the transform domain in future.

We have also presented a two stage approach for image retrieval. At first stage, initial set of features described in section 3.2 is used for image retrieval. At next stage, additional features described in section 3.3 are considered for retrieval. Hence, this approach is computationally efficient and provides refined result because of 2-stage retrieval. Results show algorithm presented provides relevant retrieval results.

Acknowledgment: This study was supported by research fund from Chosun University, 2006.

References

1. Greg Pass and Ramin Zabih. Histogram Refinement for content-based image retrieval. In IEEE Workshop on Applications of Computer Vision, pages 96-102, December 1996.
2. Arnol W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, Ramesh Jain, "Content Based Image Retrieval at the end of the early years", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 22, No. 12, pp 1349-1380, 2000.
3. Wynne Hsu, T. S. Chua and H. K. Pung. An integrated color-spatial approach to content based image retrieval. In ACM Multimedia Conference, pages 305-313, 1995.
4. Paisarn Muneesawang and Ling Guan, "Automatic machine interactions for content based image retrieval using a self organizing tree map architecture", IEEE Transactions on Neural Networks, Vol 13, No. 4, pp 821-834, 2002.
5. Michael Swain and Dana Ballard. Color indexing. International Journal of Computer Vision, 7(1):11-32, 1991.
6. Rick Rickman and John Stonham. Content based image retrieval using color tuple histograms. SPIE proceedings, 2670:2-7, February 1996.
7. Chabane Djeraba, "Association and content based retrieval", IEEE Transactions on Knowledge and Data Engineering, Vol. 15, No. 1, pp 118-135, 2003.
8. Markus Stricker and Alexander Dima. Color indexing with weak spatial constraints. SPIE proceedings, 2670:29-40, February 1996.
9. Jing Huang, S. Ravi Kumar, Mandar Mitra, Wei-Jing Zhu, and Ramin Zabih. Image indexing using color correlograms. In IEEE Conference on Computer Vision and Pattern Recognition, pages 762-768, 1997.
10. Jong-An Park, Min-Hyuk Chang, Tae Sun Choi and Muhammad Bilal Ahmad, "Histogram based chain codes for shape description," IEICE Trans. On Communications, vol.E86-B, no.12, pp. 3662-3665, December 2003.
11. K.H.Choi, M.H.Shin, S.H.Bae, C.H.Kwon, I.H.Ra, "Similarity retrieval based on SOM-Based R*-Tree", LNCS 3038, International Conference on Computational Science (ICCS), pp 234-241, 2004.
12. Xin Huang, Shijia Zhang, Guoping Wang, Heng Wang, "Optimal matching of images using combined color feature and spatial feature", LNCS 3991, International Conference on Computational Science (ICCS), vol. 1, pp 411-418, 2006.
13. James Z. Wang, Jia Li, Gio Wiederhold, "SIMPLIcity: Semantics-sensitive Integrated Matching for Picture Libraries," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol 23, no.9, pp. 947-963, 2001.