EDGE COLOR HISTOGRAM FOR IMAGE RETRIEVAL

Seong-O Shim, Tae-Sun Choi, *Senior Member, IEEE*

Department of Mechatronics,
Kwangju Institute of Science and Technology,
1 Oryong Dong, Puk-Gu, Kwangju 500-712, Korea

ABSTRACT

Classical color histogram for image indexing does not take into account shape information of an image. In this paper, color histogram with edge information is studied. Color distributions are found for the pixels of three types of edges (two directional edges and one non-directional edge) and three distance measures are computed on the basis of color distribution of each edge type. Proposed similarity measure obtained by combining these three distance measures could reduce the false match rate in comparison with using only single (non-directional) edge type. Simulation results show improvement of indexing quality as compared to that of traditional color histogram and edge histogram.

1. INTRODUCTION

The main purpose of Content-Based Image Retrieval (CBIR) is to find the relevant image from a database comparing with user provided query image using only the image information. Therefore, it is very important for CBIR to define an efficient and robust image descriptor. Color is considered as the most dominant and distinguishing visual feature. A color histogram describes the global color distribution in an image [1, 2]. While the color histogram is robust to translation of object and rotation about the viewing axis, it does not include any spatial information. Different images can have same color distributions and large appearance changes in an image can easily change the histogram. Thus the use of object-specific information contained in images is essential for efficient image retrieval. Therefore shape-describing features along with color have been intensively researched [3, 4, 5]. Theoretically the best way to extract shape information in images is by segmenting the object in the image. But the segmentation process is very complex and computationally too expensive and if there are no specific objects in an image, the result of segmentation may not be relevant part of the original image.

Incorporating directional edge information to the color information is studied in this article. Edge is local shape feature and it captures the general shape information in the image [6]. Since edges play an important role for image perception, it is frequently used as a feature descriptor in image retrieval. The edge histogram descriptor is an example, which represents the spatial distribution of five types of edges, namely four directional edges and one non-directional edge [7]. Many object feature based approaches use color feature as an object internal feature after segmentation is done [5]. The idea starts from the thought that the color distribution within the edges, or more accurately speaking - the color distribution of pixels that belong to the edges, can be considered as an internal feature of edge shape descriptor.

In this paper, three color histograms are computed: two for directional edges (horizontal and vertical edges), and one for non-directional edge (combination of horizontal and vertical edges). The three color histograms are used as components of feature vector on each images. The proposed approach can be considered as a combination of color histogram and edge histogram methods.

This paper is organized as follows. In section 2, color histogram for the directional edge is studied. Similarity measures and performance measures are discussed in section 3 and 4. Finally the results are compared with that of classical color histogram and edge histogram.

2. COLOR HISTOGRAM FOR DIRECTIONAL EDGE

2.1. Color descriptor for image retrieval

Color is one of the most dominant and distinguishing visual feature. A color histogram is a global feature that captures color distribution in an image. But it does not capture any spatial information in an image. However, color histogram is widely used in image retrieval because of its lower complexity in comparison to traditional techniques of pattern recognition.

Let I be an image quantized to m colors c_1, c_2,..., c_m. For a pixel $p=(x,y)\in I$, we denote $I(p)$ as it's color and $I_c = \{p \mid I(p) = c\}$. Then color histogram is defined by

$$h_{c_i}(I)=n\cdot \Pr_{p\in I}\left[p\in I_{c_i}\right] \tag{1}$$

where n is the total number of pixels in an image. For any pixel in the image, normalized histogram $H_{c_i}(I)=h_{c_i}(I)/n$ gives the probability that the color of the pixel is c_i.

2.2. Edge descriptor for image retrieval

Edge is a local shape feature and it captures the general shape information in the image [6]. Since edges play an important role for image perception, they are frequently used as a shape describing feature descriptor in image retrieval. The edge histogram descriptor is an example, which represents the spatial distribution of five types of edges, namely four directional edges and one non-directional edge [7].

But edge histogram is not rotation invariant. Rotation of the image shifts each of the edge directions by the amount of rotation and also affects the membership in the bins [10]. For example, if a quantization of 45° is used, two edge points with direction of 30° and 40° will fall into the same bin. However, if the same image is rotated by 10° then these two points will fall into adjacent bins.

2.3. Color distribution around the edges

The changes of color in an image occur at the color edges. So the color distribution on the pixels around color edge is very similar to that of entire image. Therefore the color distribution of pixels around edges instead of considering all pixels in an image can be used in describing image feature.

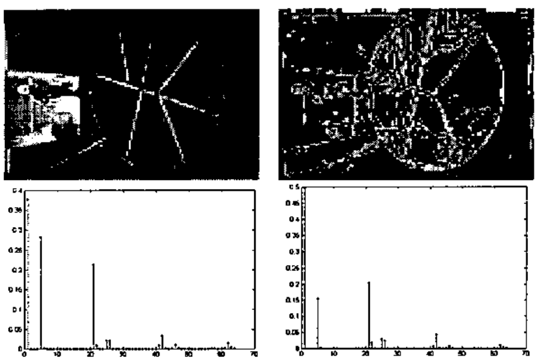


Figure 1. Comparison between the color distribution on the full image and the color distribution on the pixels around color edge (x axis: color bins, y axis: normalized color histogram)

Figure 1 shows that the color distribution on the pixels around edge is quite similar to that on the full image except the small changes in scale. It is because a color conquering a big area in an image also takes big area even when only the pixels around edges are taken into consideration.

2.4. Combining color and edge

In this paper, color histograms are computed for the pixels of two directional edges (horizontal and vertical edges), and of one non-directional edge (combination of horizontal and vertical edges). These three color histograms constitute components of the proposed image feature vector. Therefore the new feature vector has two dimensions (n, m), where n is for three edge bins and m is for color bins corresponding to n^{th} edge bin. We will call it as edge color histogram. In this case, while the component of traditional edge histogram feature vector is the number of pixels for each directional edge bins, the component of proposed feature vector is color distribution for the pixels of each directional edge bins.

2.5. Procedure for edge color histogram

The first step in computing edge color histogram is to find directional edges in an image. To find edges from color image the method from [7] is employed. At first, the color image is transformed to the HIS color space from which the hue channel is neglected. The other two channels are convolved with the two Sobel operators (horizontal and vertical operators). The resulting gradient images are next thresholded to binary images by a proper threshold value for each channel. The threshold values are manually fixed to certain levels which are the same for all images. The thresholded intensity and saturation gradient images are combined by the logical OR operation. The threshold value for the intensity gradient image was manually fixed to 15% of the maximum gradient value and for the saturation image to 30%. In the OR operation, the direction of the larger gradient value is chosen. Next the non-directional gradient edge image is computed by adding the directional gradient edge images. These three binary edge images are dilated one time to take into account pixels on both sides of the edges. Finally three color histograms are computed from original image by considering only the pixels whose intensity value has 1 on the previously computed binary edge images.

Let E_j is the set of pixels which belong to jth edge bin, and

$$n_j = \left|\{p \mid p \in E_j\}\right| \tag{2}$$

where $|\cdot|$ means the number of elements. The color histogram for image I on the jth edge bin is defined as

$$h_I(j,k) = n_j \cdot \Pr_{p \in I}\left[p \in I_{c_k} \mid p \in E_j\right] \tag{3}$$

Here k is the index for quantized colors and edge bins, j s, are lined up in the order of horizontal edge, vertical edge, and non-directional edge. If color is quantized into m colors, it can be written as $k \in [m]$. Still it needs to be normalized to keep scaling invariant property within the image. Normalized final color histogram, the edge color histogram, can be defined as

$$H_I(j,k) = h_I(j,k)/n_j = \Pr_{p \in I}\left[p \in I_{c_k} \mid p \in E_j\right] \tag{4}$$

Since $j \in [3]$, the dimension of the histogram $H_I(\cdot)$ is $3 \times m$. Dimension can be extended to $5 \times m$ or $9 \times m$ by applying four directional edge operators or eight directional edge operators.

3. SIMILARITY MEASURES

In general, the distance between two histograms $H(Q)$ and $H(I)$ in L^p metric can be expressed as

$$d_{L^p}(H(Q), H(I)) = \sum \left|H(Q) - H(I)\right|^p \tag{5}$$

The L^1 (absolute error) and L^2 (square error) distance measures are commonly used when comparing two feature vectors. In practice, the L^1 distance measure performs better than the L^2 distance measure because the former is statistically more robust to outliers [8]. Hafner *et al.* [9] suggested the weighted distance between color histograms of two images, represented as a quadratic form, by trying to capture the perceptual similarity between any two colors. To avoid excessive computation for quadratic form, they proposed low-dimensional color features as filters before using the quadratic form for the distance measure.

In this paper, the similarity $S(Q,I)$ between the user provided query image Q and image I from the image database is defined on the basis of edge color histogram $H(\cdot,\cdot)$. First the distance vector $D(Q,I)=(d_1,d_2,d_3)$ is computed, where d_j is defined as

$$d_j = \left\| H_Q(j,k) - H_I(j,k) \right\| \tag{6}$$

where $\|\cdot\|$ denotes *Euclidean distance*. Finally the similarity measure $S(Q,I)$ is defined as

$$S(Q,I) = \frac{w_1 d_1 + w_2 d_2 + w_3 d_3}{w_1 + w_2 + w_3} \tag{7}$$

where w_1, w_2, and w_3 are the weights assigned to the each components of distance vector d_1, d_2, and d_3. In this case, similar images produce low similarity measure and if two images are identical, similarity measure is zero.

Distance measure d of similar images has low value, but d of perceptually different images may also have low value (false matches). This may affect the rank order of the correct match and hence decrease the accuracy of given matching scheme. By splitting the edge types according to the directions and using the proposed similarity measure (7) reduces the number of false retrievals, as it is unlikely that a pair of perceptually different images have low distance value in all three distance measures.

We assigned same weights ($w_1 = w_2 = w_3 = 1$) to all components of distance vector. Experimental results showed that matching using the distance measure between the color histograms of non-directional edge bins, d_3, has a little higher accuracy than others. So a little high weight can be assigned to d_3 in equation (7).

4. PERFORMANCE MEASURES

Let $\{Q_1,\ldots,Q_q\}$ be the set of query images. For a query Q_i, let Q_i' be the unique correct answer. The following measurements [3] are used to compare the performance of different image features.

1. r-measure: the average rank of the correct answer over all queries, i.e., $r = (1/q)\sum_{i=1}^{q} rank(Q_i')$.

2. p_1-measure: the average precision at recall equal to 1, i.e., $p_1 = (1/q)\sum_{i=1}^{q} 1/rank(Q_i')$.

3. *recall vs. scope*: Let R_1, $R_2,\ldots,R_t$ be the top t retrieved images based on the query image Q and each of them is labeled according to their relevance to image Q. The *recall r* is defined for a *scope*, to be $s > 0$ as

$$r(s) = \left| \{ R_i \mid relevant(R_i) = true, 1 \le i \le s \} \right|$$

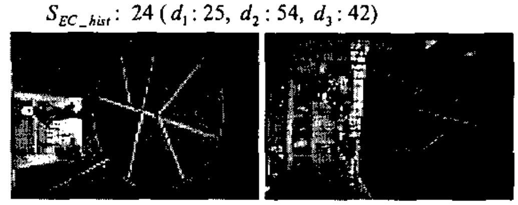
Images ranked at the top contribute more to the p_1-measure but less to the r-measure, and images with low rank contribute more to the r-measure but less to the p_1-measure. In order to have a fair comparison, we have to look at both r-measure and p_1-measure. Note that a method is good if it has low r-measure and a high p_1-measure.

5. EXPERIMENTS

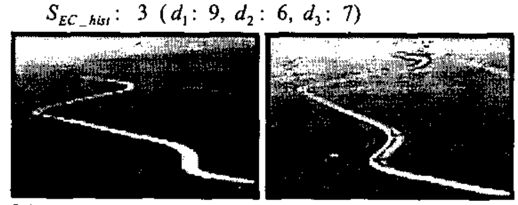
The proposed algorithm using edge color histogram has been implemented and tested with a database of 6,000 images from UC Berkeley digital library project. Database contains 80 queries and their correct answers. The queries are chosen to represent various situations like different views of the same scene, small lighting changes, spatial translations, changes in appearance and size, etc.

Examples of some queries and answers are shown in Figure 2. To compare with previous methods, rankings are calculated using similarity measure based on traditional color histogram method (S_{C_hist}), edge histogram method (S_{E_hist}), and proposed edge color histogram method (S_{EC_hist}). S_{C_hist} and S_{E_hist} are calculated by using *Euclidean distance* between the histogram feature vectors. As we can see, S_{EC_hist} is a function of d_1, d_2, and d_3 ($S_{EC\_hist} = f(d_1,d_2,d_3)$).

[a] S_{C_hist}: 93, S_{E_hist}: 126, S_{EC_hist}: 24 (d_1: 25, d_2: 54, d_3: 42)



[b] S_{C_hist}: 10, S_{E_hist}: 17, S_{EC_hist}: 3 (d_1: 9, d_2: 6, d_3: 7)



[c] S_{C_hist}: 574, S_{E_hist}: 69, S_{EC_hist}: 17 (d_1: 23, d_2: 17, d_3: 30)

Figure 2. Sample queries and answers with ranks for different similarity measures.

In Figure 2, the first image pairs are to investigate indexing quality for the case when the query image is the rotation of the corresponding answer image. Second image pairs are for translation of the query image. And third pairs are for the lighting changes of the query image.
The proposed edge color histogram technique outperformed traditional color histogram and edge histogram methods. It is worthy to note that ranking using d_1, d_2, and d_3 also outperformed traditional methods. Among the ranking using d_1, d_2, and d_3, not one of these considerably outperformed the others. And ranking using the similarity measure S_{EC_hist} outperformed the ranking using d_1, d_2, and d_3 as expected.
The performance of traditional techniques and proposed technique is compared in Table 1.

	r-measure	p_1-measure
S_{E_hist}	99.8	2.48
S_{C_hist}	85.8	2.04
d_1	31.5	3.34
d_2	67.4	2.69
d_3	29.4	3.35
S_{EC_hist}	22.6	3.86

Table 1. Performance of different features.

For 62 out of 80 queries, edge color histogram method performs as well or better than color histogram and edge histograms. In the case where edge color histogram performs better than color histogram and edge histogram, the average improvement in the rank of correct answer is 98 positions, and where color histogram and edge histogram perform better than edge color histogram, improvement is just 9 positions.

Figure 3 shows the comparison based on the number of relevant images retrieved at various scopes. Result shows edge color histogram outperforms color histogram and edge histogram.

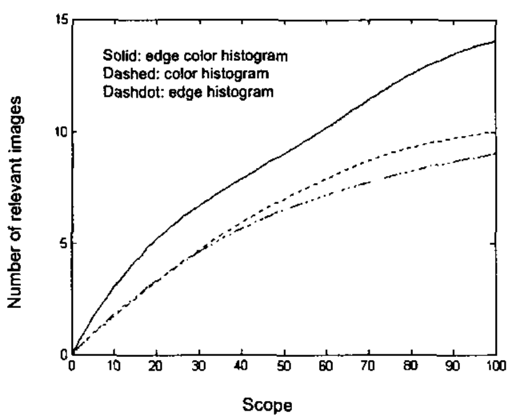


Figure 3. *recall vs. scope* responses of different methods

6. CONCLUSIONS

In this paper, a new image descriptor called edge color histogram is proposed, which investigates color distributions for the pixels of three types of edges (two directional edges and one non-directional edge). Since it considered both the color and the local shape feature, the new image descriptor captured more image information. By splitting edge color histogram according to the types of edges, three distance measures were computed. Our proposed similarity measure associating these three distance measures could reduce false match rate and thereby could further refine indexing quality.
Experimental results show that proposed edge color histogram outperforms traditional color histogram and edge histogram. Performance can be further improved by using four directional edges or eight directional edges.
The proposed edge color histogram is efficient and can be used in many applications where traditional color histogram or edge histogram technique is applied.

Acknowledgement: This research was supported by the International Cooperation Research Program of the Korea Ministry of Science & Technology. The authors wish to thank Abdul Rauf for his reviews which helped them improve this work.

7. REFERENCES

[1] M. Swain and D. Ballard, "Color indexing", *International Journal of Computer Vision*, 7(1):11-32, 1991.
[2] M. Flicker *et al.*, "Query by image and video content: The QBIC system", *IEEE Computer*, 28(9):23-32, 1995.
[3] J. Huang, S. R. Kumar, M. Mitra, W. Zhu, and R. Zabih, "Image indexing using color correlograms", *Proc. Conf. Computer Vision Pattern Recognition*, 1997, pp. 762-768.
[4] Ying Deng, B. S. Manjunath, Charles Kenney, Michael S. Moore, Hyundoo Shin, "An efficient color representation for image retrieval", *IEEE Transactions on Image Processing*, vol. 10, Jan. 2001, pp. 140-147.
[5] Arnold 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, Dec. 2000, pp. 1349-1380.
[6] Sami Brandt, Jorma Laaksonen and Erkki Oja, "Statistical Shape Features in Content-Based Image Retrieval", *Proceedings of 15th International Conference on Pattern Recognition*, vol. 2, 2000, pp. 1062-1065
[7] International Organization for Standardization Organization Internationale De Normalization ISO/IEC JTC1/SC29/WG11 Coding of Moving Pictures and Audio, ISO/IEC JTC1/SC29/WG11 N4031, Singapore, March 2001.
[8] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*, John Wiley & Sons, 1987.
[9] James Hafner *et al*, "Efficient Color Histogram Indexing for Quadratic Form Distance Functions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, No. 7, July. 1995, pp. 729-736.
[10] A. K. Jain, A. Vailaya, "Image retrieval using color and shape", *Pattern Recognition*, 29(8), pp. 1233-1244, 1996