3-1-2002

Histogram based image indexing and retrieval.

Sung-Hyuk Cha

SCHOOL OF COMPUTER SCIENCE AND INFORMATION SYSTEMS

TECHNICAL REPORT

Number 173, March 2002

Histogram Based Image Indexing and Retrieval

Sung-Hyuk Cha

PACE UNIVERSITY

Sung-Hyuk Cha is Assistant Professor of Computer Science at Pace University, based in Westchester. Dr. Cha holds baccalaureate and masters degrees from Rutgers University and a doctorate in computer science from the State University of New York at Buffalo. He joined the faculty at Pace University in September, 2001.

Dr. Cha's research interests lie in the areas of distance measure and pattern matching algorithms, pattern recognition, image analysis, and machine intelligence and data mining.

Histogram based Image Indexing and Retrieval

Sung-Hyuk Cha
Computer Science Department, Pace University
861 Bedford Road, Pleasantville, New York, 10570 USA
scha@pace.edu

Abstract

In content-based image indexing and retrieval (IIR), hue component histograms of images are widely used for indexing the images in an image database. It is to retrieve all colour images whose distance between hue distributions are within some threshold distance of the query image. Edit distance has been successfully used as a similarity measure. Our earlier $O(b^2)$ algorithm computing the edit distance between two angular histograms, where b is the number of bins in the hue histogram, tends to be too slow for users to wait for the outputs when applied to every image in the database. For this reason, we design two filtration functions that eliminate most colour images from consideration as possible outputs quickly and exact edit distances are only computed for those remaining images. We are still guaranteed to find all similar hue distributions and the filtration technique gives significant speeds-ups.
Key Words: Colour image, Filtration, Hue, Image indexing and retrieval, Similarity Measure

1. Introduction

In content-based image indexing and retrieval (IIR) systems, a user is interested in retrieving all images similar to a given query image from a database of images. This challenging problem has received great attention due to many applications (see the extensive survey [10]). The histogram of hue values (the hue component of the Hue-Value-Chroma (HVC) colour space) is widely used to retrieve images, the city-block or Manhattan distance, Euclidean, intersection [7,9,11], cross-entropy [6] measures are often used to measure the distance or similarity between image hue histograms. These distances regarding only the overlap between two histograms have the disadvantage that they do not take into account the similarity of the non-overlapping parts of the two distributions [2,3,4].

To overcome this disadvantage, the new measure using the notion of the Minimum Difference of Pair Assignments, called Edit distance was introduced [2,3,4]. Simultaneously, Rubner and et al.introduced the similar distance called an Earth Mover's Distance [8]. Albeit this distance is very promising measure, when the measurement type is angular or modulo, our earlier $O(b^2)$ algorithm computing the edit distance between two angular histograms, where b is the number of bins in the hue histogram, tends to be too slow for users to wait for the output. If we have n images in the image dictionary, retrieving similar images to a query image takes $O(nb^2)$.

For this reason, we design a fast algorithm which quickly eliminates most colour images from consideration as possible matches. Suppose we have another distance measure, d' such that $d'(I_i,q) \le d(I_i,q)$ where d is the edit distance, and d' can be computed very fast $o(nb^2)$. Then we can use d' to filter the colour images, eliminating from consideration if $d'(I_i,q) > t$ where t is the threshold value.

This candidate selection and filtration algorithm was applied successfully when the measurement type, or histogram level type is linear or ordinary [1]. When the measurement type is angular like hue values, designing such a filtration technique is not as straight-forward as in linear type histogram. This paper introduces two levels of filtration functions for angular type histogram.

1.1. Organization

The rest of this paper is organized as follows. In section 2, the concept of edit distances for linear and angular type histograms are reviewed. Section 3 discusses designing the filtration functions for each type of histogram. Finally, section 4 concludes this work. The major contribution of this paper is designing the filtration function for angular histograms generated from colour images.

2. Edit Distance

In this section, we briefly discuss the edit distance between histograms and how it can be used for image indexing and retrieval system (see [1] for detailed description and proofs on measuring distance between histograms). We first show the linear type histograms generated from grey scale images because the algorithm for the distance measure between angular histograms is based on them.

2.1. Linear type histogram & edit distance

Consider the 1760x1760 Computed Radiography images, CR in short, shown in Figure 1. Medical images are represented by a rectangular array of picture elements called pixels, which are linear numerical values ranging from 0 to 1023. An intensity histogram represents the frequency of each grey level in a grey level image as shown in Figure 2.

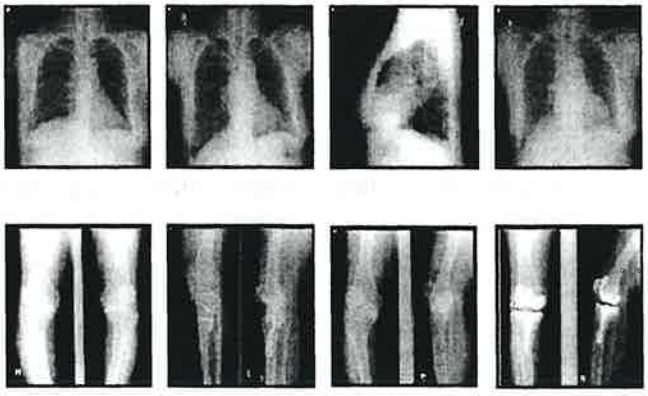


Figure 1. 1760x1760 CR chest and knee images.

Similar CR images whose body parts and view positions are the same tend to have similar intensity histograms according to human visual system. Good distance measure between intensity histograms must return the similar result. In the earlier work [2], it is well described why the conventional distance measures such as Euclidean, city-block, cross-entropy, or K-L distance fail and the new edit distance is better.

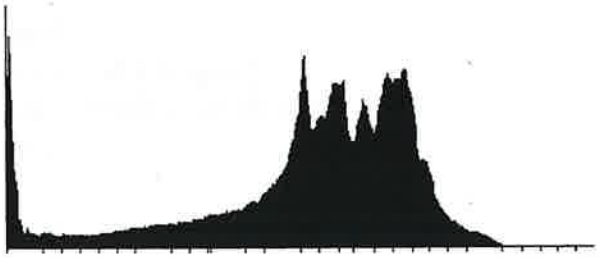


Figure 2. a sample intensity histogram from a CR image.

Figure 3 briefly explains how the distance between two linear type histograms is computed. It was originally developed to expedite the image template matching problem [].

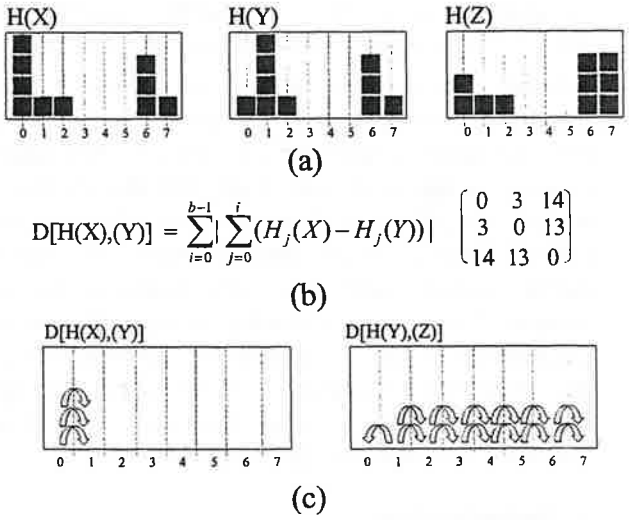


Figure 3. Illustration of computing the distance between linear type histograms: (a) three sample linear type histograms (b) the computing distance formula and (c) the arrow representation of the cell movement necessary to transform one histogram to another.

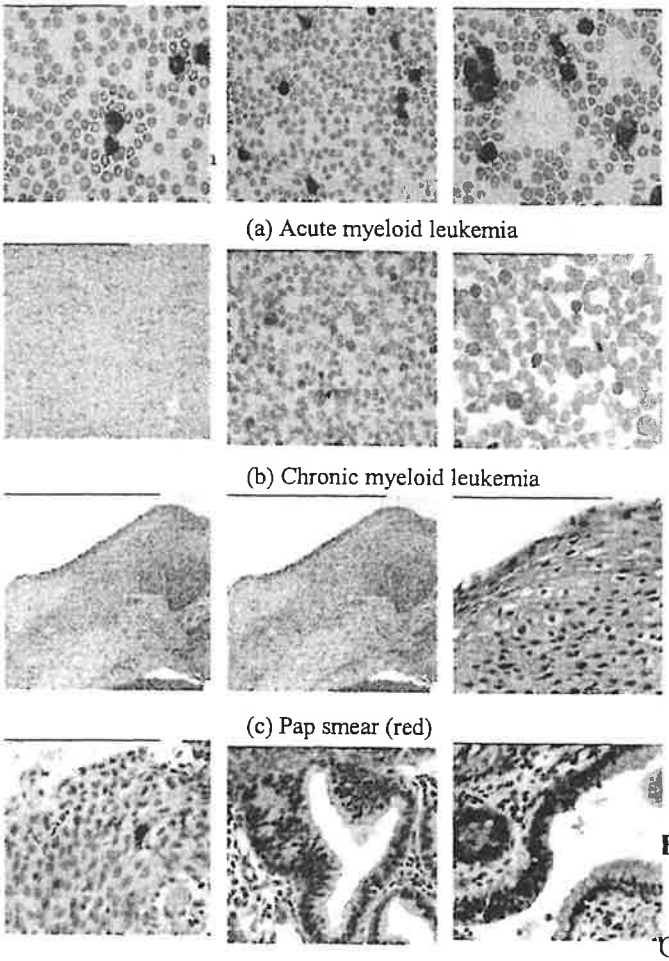
Consider the three histograms and a histogram H(X) can be transformed into H(Y) by moving elements to left or right and the total of all necessary minimum movements is the distance. The computational time complexity for computing the distance formula in Figure 3 (b) is $O(b)$. Table 1 shows the distance matrix of sample CR image intensity histograms and

we can observe the similarities between the same body part and view position CR images.

Table 1. Distance Matrix of Sample CR image Intensity Histograms

-	c1	c2	c3	c4	k1	k2	k3	k4
c1	0	26	451	68	266	275	232	270
c2	26	0	447	60	281	271	240	266
c3	451	447	0	403	214	196	262	187
c4	68	60	403	0	250	235	221	222
k1	266	281	214	250	0	56	91	64
k2	275	271	196	235	56	0	71	30
k3	232	240	262	221	91	71	0	89
k4	270	266	187	222	64	30	89	0

2.2. Hue Histograms & Edit Distance



(a) Acute myeloid leukemia

(b) Chronic myeloid leukemia

(c) Pap smear (red)

(d) Pap smear (purple)

Figure 4. Sample Bloodsmear color images

Consider sample bloodsmear color images shown in Figure 4 and there are four categories. Although the intensity histograms may be used for indexing after converting the color images to grey scale images, hue component histograms are more valuable than the intensity histogram. The HVC space is most suited for identifying regions of a particular color in an image and is insensitive to lighting variations in the image. Hue histograms are quite different from the intensity histogram, i.e., the measurement type of hue histograms is angular instead of linear. The angular measurement ranges from 0° to 360°. Figure 5 shows the corresponding hue histograms in circular histogram representation.

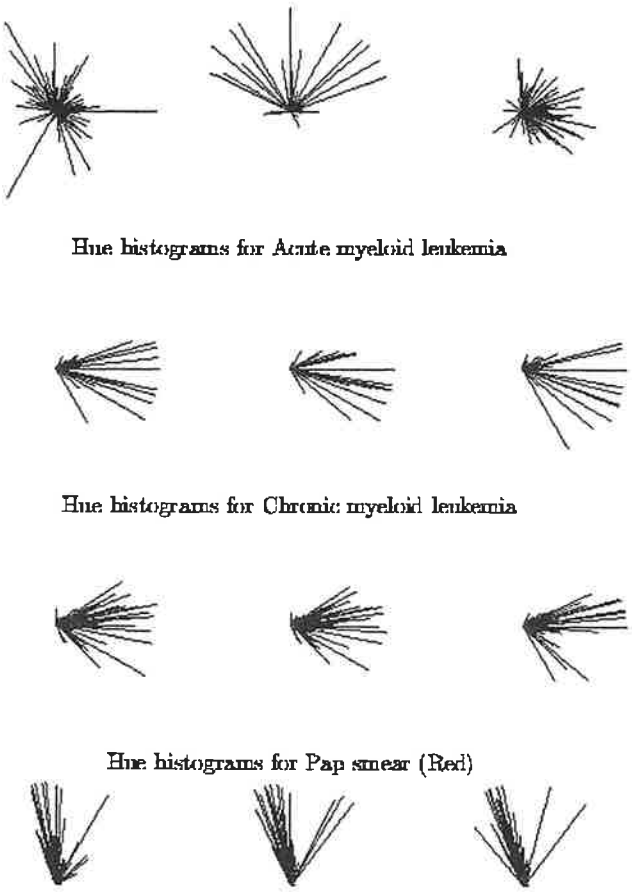


Figure 5. Corresponding Circular Hue histograms of Bloodsmear color images in Figure 4.

Computing the edit distance between two angular type histograms is more challenging than that between linear type histograms. Figure 6 illustrates the algorithm counting the clockwise and counter-clockwise cell movement necessary to transform one angular histogram to another.

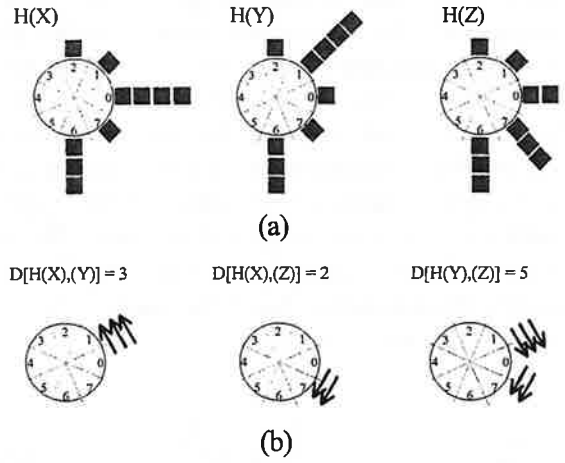


Figure 6. Ilustration of computing the distance between angular type histograms: (a) three sample angular type histograms and (b) the arrow representation of the cell movement necessary to transform one histogram to another.

Our algorithm to compute the distance between two angular histograms takes $O(b^2)$ in time, where b is the number of bins in the hue histogram. It tends to be too slow for users to wait for the output. If we have n images in the image dictionary, retrieving similar images to a query image takes $O(nb^2)$. Hence, in the following section, we give an efficient algorithm to compute the distance quickly.

Table 2 shows the distance matrix of sample bloodsmear colour image hue histograms and we can observe the similarities among the same categories. The measure satisfies conditions for being a metric[2].

Table 2. Distance Matrix of Sample bloodsmear colour image Histograms

-	aml1	aml2	aml3	cml1	cml2	cml3	psr1	psr2	psr3	psp1	psp2	psp3
aml1	0	8057	10783	17159	17597	16467	15026	15012	15283	12283	14115	14114
aml2	8057	0	17690	21836	22318	21846	20157	20080	20065	5125	6511	6414
aml3	10783	17690	0	7748	7629	6581	6238	6194	6806	18571	21337	21799
cml1	17159	21836	7748	0	753	1267	2238	2254	2211	20835	23784	24454
cml2	17597	22318	7629	753	0	1276	2621	2643	2647	21333	24274	24967
cml3	16467	21846	6581	1267	1276	0	2159	2176	2338	21203	24152	24774
psr1	15026	20157	6238	2238	2621	2159	0	156	782	19474	22402	23070
psr2	15012	20080	6194	2254	2643	2176	156	0	789	19424	22353	23019
psr3	15283	20065	6806	2211	2647	2338	782	789	0	19485	22434	23088
psp1	12283	5125	18571	20835	21333	21203	19474	19424	19485	0	2960	3745
psp2	14115	6511	21337	23784	24274	24152	22402	22353	22434	2960	0	1056
psp3	14114	6414	21799	24454	24967	24774	23070	23019	23088	3745	1056	0

3. Filtration Function

This section presents designing a fast algorithm which quickly eliminates most images from consideration as possible matches. Suppose we have another distance measure, d' such that $d'(I_i,q) \le d(I_i,q)$ where d is the edit distance, and d' can be computed very fast $o(nb^2)$. Then we can use d' to filter the colour images, eliminating from consideration if $d'(I_i,q) > t$ where t is the threshold value. Similar images according to the measure d are still guaranteed to be retrieved.

The filtration algorithm has two stages: a candidate selection stage and a candidate verification stage. Let L and M be a set of all images in the database and a set of all matches according to the histogram edit distance measure, respectively. Let C be a candidate set, which is a subset of the entire image database and contains all matches. The candidate selection stage is taking all images from the image database and ouputs C. There are three essential properties about good candidate sets.

- C must contain all matches to the query: $M \subseteq C \subseteq L$.
- C must be computed in $o(nb)$ and $o(nb^2)$ for linear and angluar type histograms, respectively.
- The expected $|C|$ must be small.

The first property guarantees the correctness of the algorithm and the second and third ones guarantee the speedup.

3.1. Linear Intensity Histogram Case

One of good candidate selection function is the difference between sums of all pixel values in the grey level images. The sums of all pixel values of an image X equals to the sum of all frequencies of all

levels in the intensity histogram $H(X)$ and can be denoted as $\sum_{i=0}^{b-1} i \times H_i(X)$. This function satifies all three aforementioned properties.

Theorem 1. The candidate set filtered by the difference between the sums of all pixel values contain all matches.

Proof: Consider two images X and Y. Then the difference between the sums of all pixel values is $|\sum_{i=0}^{b-1} i \times H_i(X) - \sum_{i=0}^{b-1} i \times H_i(Y)|$ and the edit distance is $\sum_{i=0}^{b-1} |\sum_{j=0}^{i} (H_j(X) - H_j(Y))|$. Clearly $|\sum_{i=0}^{b-1} i \times H_i(X) - \sum_{i=0}^{b-1} i \times H_i(Y)| \leq \sum_{i=0}^{b-1} |\sum_{j=0}^{i} (H_j(X) - H_j(Y))|$ by triangle inequality. ■

Computing the difference between sums of all pixel values is constant if sums are precomputed, whereas the computing the edit distance is $O(b)$ even thought the histograms are already built.

Hence, when we have a query image, one can simply compute the differences between the sum of the query image pixel values and that of each image from the image library. This candidate selection stage is done in $O(n)$ which is clearly $o(nb)$ satisfying the second property. Next, only for those selected candidate images, one computes the edit distance against the query image intensity histogram because of the theorem 1;

$$M \subseteq C \subseteq L \text{ implies } |M| \leq |C| \leq |L|.$$

3.2. Angular Hue Histogram Case

When histograms are angular in type, designing the filtration function is not as easy as in the linear type case. The difference between sums of all hue values fails to satisfy the first property. To design a function, the angular histogram is first converted to a linear type histogram. To convert it, we introduce a folded histogram.

Consider the three angular histograms shown in Figure 6 (a). The bottom half of the circle is folded to the top and frequencies of hue values are added in the corresponding position as shown in Figure 7 (a).

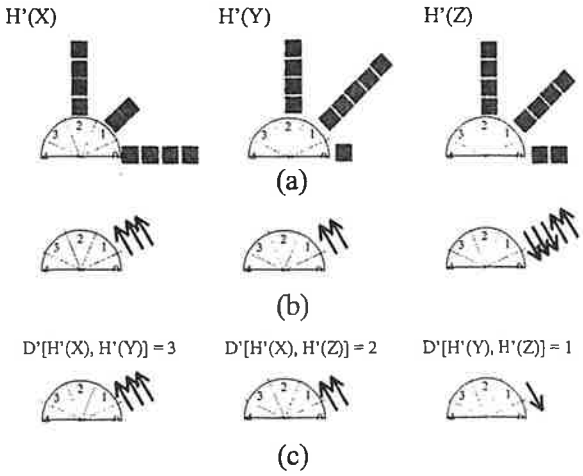


Figure 7. Illustration of designing a filtration function for angular histograms: (a) folded histograms for three angular histograms shown in Figure 6 (a), (b) edit distances between two angular histograms, and (c) the edit distances between two folded histograms.

The folded histograms are no longer angular but linear in type. Thus if we take the edit distance between linear folded histograms, it takes only $O(\frac{b}{2})$ in time complexity which is must faster than $O(b^2)$ in computing the edit distance between two angular histograms.

Figure 7 (b) and (c) gives the intuitive correctness that the distance between two folded histograms is equivalent to or smaller than that between two angular histograms. The number of arrows in Figure 7 (c) is the distance between two folded histograms and that in Figure 7 (b) is the distance between two angular histograms. This is because when the arrow representation between two histograms is folded, the opposite arrows on the same level are cancelled out and that is the edit distance between two folded histograms. Clearly, one can use the edit distance between folded histograms as a candidate selection function for the angular histograms as it is much faster to compute and guarantees to include all the matches.

Since folded histograms are linear, the combined filtration function can be designed for the angular type histograms. After building the folded histograms, take the sum of these folded histograms

and have these sum values as an index as shown in Figure 8. When the query colour image is present, use the difference between sums as a first filtration function, which results in the first candidate set denoted C1.

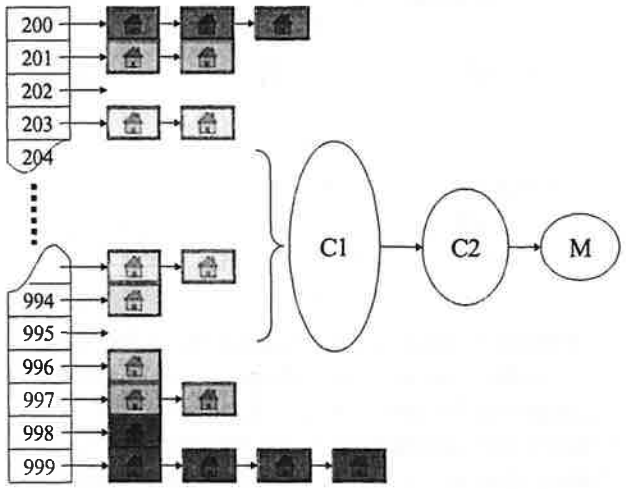


Figure 8. Color Image Database and Indexing.

Next, compute the linear type edit distances only for those images in C1 and create a smaller candidate set called C2. Then, we have $M \subseteq C2 \subseteq C1 \subseteq L$. Finally, compute the expensive angular type edit distance measures for those images in C2. The total running time complexity is $O(n+|C1|\times b+|C2|\times b^2)$ which is much faster than the original $O(nb^2)$ because $|M| \leq |C2| \leq |C1| \leq |L|$

4. Conclusion

In this paper, a fast algorithm to compute the edit distance between histograms using several filtration functions. This technique has two stages: candidate selection and veridation stages. Three important properties when designing a filtration function were discussed. The major contribution of this paper is designing the filtration functions for angular histograms such as hue histograms generated from color images.

In this paper, we have explored only the hue component of the HVC color space. The value and chroma components should also be used in indexing and retrieving images. The histograms of these two types are linear. Measuring the distance between multidimensional histograms of these heterogeneous types is an open problem.

Acknowledgements

The author would like to thank Dr. David J. Foran at the University of Medicine and Dentistry of New Jersey (UMDNJ) for providing the bloodsmear images.

Reference

[1] S.-H. Cha. *Efficient Algorithms for Image Template and Dictionary Matching, Journal of Mathematical Imaging and Vision*, Vol 12-1, 2000, pp 81-90

[2] S.-H. Cha, and S. N. Srihari. *On Measuring the Distance between Histograms, Journal of Pattern Recognition*,Vol 35-6, 2002, pp 1355-1370

[3] S.-H. Cha, and S. N. Srihari, *Distance between Histograms of Angular Measurements and its Application to Handwritten Character Similarity,* In Proceedings of 15th ICPR, Barcelona, Spain, p21-24, 2000

[4] S.-H. Cha, and S. Munirathnam, *Comparing Colour Images using Angular Histogram Measures,* In Proceedings of 5th JCIS, Vol II, CVPRIP, p139-142, 2000

[5] T. Chiueh. *Content-based image indexing, in Proceedings of the 12th International Conference on Very Large Databases*, Santiago, Chille 1994, pp. 582-593

[6] S. Munirathnam. *Image Indexing and Retrieval using the Cross-Entropy Measures, in Proceedings of the HKK*, Waterloo, Ontario, 1999

[7] G. Pass, R. Zabih, and J. Miller, *Comparing images using color coherence, in* ACM International Multimedia Conference, pp. 65-73, 1996

[8] C.T. Y. Rubner and L. J. Guibas, *A metric for distributions with applications to image database, in* International Conference on Computer Vision, IEEE, pp. 59-66, 1998

[9] H. S. Sawhney and J. L. Hafner, *Efficient color histogram indexing, in* International Conference on Image Processing, vol 1, pp. 66-70, 1994

[10] A. W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, *Content-Based Image Retrieval at the End of the Early Years,* IEEE *Trans. on Pattern Analysis and Machine Intelligence,* Vol. 22 No. 12 pp. 1349-1380, IEEE Computer Society, 20

[11] M. J. Swain, and D. H. Ballard, *Color Indexing,* International Journal of Computer Vision, vol. 7, pp. 11-32, Nov. 1991