An Approach: to Find The Similar Color Based Pictures From a Multimedia Database.

Falguni Chakraborty[1]
Student of M.Tech (Operation Research)
National Institute of Technology, Durgapur,
India
E-mail: falguni.durgapur@gmail.com

Rita Ganguly[2]
Department of Computer Application
Dr.B.C.Roy Engineering College, Durgapur-06,
India
E-mail: ganguly.rita@gmail.com

***Abstract*—This paper describes and compares content-based global and local feature image retrieval techniques, which can be applied to estimate similarity between color images. The database of 52 color images with manually specified model similarity of each pair was prepared. The comparison of the analyzed methods was based on examining model and calculated similarities, which should be alike. Presenting experimental results using an image database which contains more than 52 color images. This experiment demonstrate clearly that the proposed weak encoding of spatial information significantly increases the discrimination power of the index compared to plain color indexing techniques**

***Keywords* -CBIR, Multimedia Database, color image**

I INTRODUCTION

Content based image retrieval has already shown its usefulness in many applications [1]. Many systems which retrieve images containing given person faces, fingerprints or objects have been implemented. A lot of them do not use color information contained in images, although color is very important in human vision. Furthermore, in many cases these systems are dedicated only to a single task – for example face recognition. The problem to find similar images to a particular one often occurs in multimedia databases of color images. Unfortunately, there is not one solution for all types of images. It happens because similarity can have different meaning and it depends on interpretation of images. In some cases global color statistics are enough for an estimation of similarity but in others, local features have to be analyzed. Searching in image databases is a challenge, because the results must be reliable and it should be computed fast.

Image may be represented using a set of low level descriptors like intensity, color, texture, shape etc. of them color is the mostly used descriptor for the image indexing and retrieval. Traditionally, color images were not used widely due to the large Storage and high computational requirements. With advances in technology, the computing and storage costs are rapidly decreasing. As a result, color images are increasingly being used in many applications. The motivation of our work is to develop a matching technique so that image retrieval based on color and size (200 X150) can be effective and fast.

Having a large number of images in proposed image database, and giving a user query to select "similar" kind of image which are same in color and size.

This work takes a further step down in selecting the right descriptors for an image, choosing the full image data: the similarities between individual images are computed through RGB by considering the information shared by each couple of objects.

II PROPOSED WORK AREA

Here a large image database is considered. The motivation of proposed work is to develop a matching technique so that image retrieval based on color and size (200 X150) pixels. Therefore the two main attributes are color and size. First just take an image from the image database and then resizing it means height =200 pixels and width = 150 pixels. After that calculating the each image RGB separately and store them in Separate matrix**[4] [5]**. The each matrix is considered as the four region of red , green and blue of each image. Each image has a separate image id**[2].**

Figure 1: Image Database

TABLE 1. CALCULATED RGB

Image Id	*Red(RE)*	*Green(G)*	*Blue(B)*
1	1560990	2585641	3229417
2	2134931	789463	1020959
3	3161404	1692497	1420085
4	2559778	1904457	2673910
5	3983944	2695873	809654
6	3114281	2059710	1929490
7	1612515	1121668	1832943
8	3123698	2688941	1279703
9	2647619	2601889	1905452
10	5283504	5367862	5164605
11	1969695	2527323	3927584
12	4693743	3013206	1954392
13	4769503	2451720	1928413
14	1621046	307327	313435
15	3757169	3016943	2955450
16	2809202	1895832	458977
17	2719604	2333003	2037776
18	4951684	3753636	3712299
19	3081419	3205331	2492556
20	4538592	3489361	3597225
21	4988510	3755489	3717295
22	4549691	3495241	3597858
23	2960090	2048368	1091394
24	2881820	1848008	2148152
25	2890182	1842151	2145841
26	2319685	1690306	1048393
27	2890182	1842151	2145841
28	2777121	1551507	1042851
29	2383734	2080501	2306545
30	3113227	1484827	791495

A Algorithm for finding RGB of an Image

Step 1: Find the index of color (RGB) of the pixel at the specified location of an image
Step 2: Find R = RGB>>16 & 0xFF
Step 3: Find R = RGB>>8 & 0Xff
Step 4: Find R=RGB & 0xFF

By applying the above algorithm in proposed image database by using PHP 5.0.3, getting the value of separate RGB of each image. It is shown in Table 1[6].

TABLE 2. CALCULATED MEAN AND S.D

	RED	*GREEN*	*BLUE*
MEAN	65265.1	54627.52	64943.58
S.D	192195.2	1648019	2179139

Handling of large database with large numbers, is very tedious job .To overcome this problem using a simple arithmetic logic to convert it into a small, easy to handling numbers. Let's the image ids are ($Xi_{=}X_1, X_2, X_3 \ldots\ldots\ldots, X_{52}$). After that, calculating Mean (M) and Standard Deviation (SD) of all 52 images of proposed database **[3]**.

Now applying the following formula getting the value of resultant Red, Green and Blue.

Resultant Red of $X_i = (RE_i - M)/ S.D$
Resultant Green of $X_i = (G_i - M)/ S.D$
Resultant Blue of $X_i = (B_i - M)/ S.D$
Where M= Mean
S.D= Standard Deviation

TABLE 3: RESULTANT RGB

Image Id	*Resultant*	*Resultant*	*Resultant*
1	7.78	1.54	1.45
2	10.77	0.45	0.44
3	16.11	0.99	0.62
4	12.98	1.12	1.20
5	20.39	1.60	0.34
6	15.86	1.22	0.86
7	8.05	0.65	0.81
8	15.91	1.60	0.56
9	13.44	1.55	0.84
10	27.15	3.22	2.34
11	9.91	1.50	1.77
12	24.08	1.80	0.87
13	24.48	1.45	0.86
14	8.09	0.15	0.11
15	19.21	1.80	1.33
16	14.28	1.12	0.18
17	13.81	1.38	0.91
18	25.42	2.24	1.67
19	15.69	1.91	1.11
20	23.27	2.08	1.62

21	25.62	2.25	1.68
22	23.33	2.09	1.62
23	15.06	1.21	0.47
24	14.65	1.09	0.96
25	14.70	1.08	0.95
26	11.73	0.99	0.45
27	14.70	1.08	0.95
28	14.11	0.91	0.45
29	12.06	1.23	1.03
30	15.86	0.87	0.33

III. RESULT AND DISCUSSION

A. Experiment 1

First considering the image id 18 and searching for a similar image from the proposed database. The resultant value of Red , Green , Blue of image id 18 is 25.42, 2.24, and 1.67.

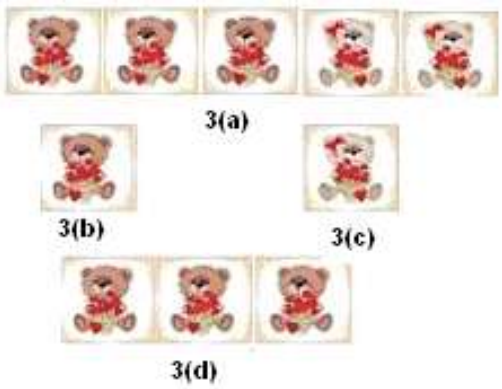
Now considering the Threshold value (0.2) for retrieval of exactly same image. After applying the Threshold value the result will be as follows-

25.22 ≤ Resultant Red ≤25.62
02.04 ≤ Resultant Green ≤02.64
01.47 ≤ Resultant Blue ≤01.87

By using the above formulae, the searching result will give the viewer exactly same of the image id 18, which have image id 21. These images are –

Figure 2. Experimental Image

B. Experiment 2.



In Figure 3(a) all the teddy looks near about same, the only one difference is that (which is shown in figure 3(b) and 3(c)) in fig 3(c) the teddy has bowtie on their left side. Otherwise no difference is found. The Red, Green, Blue of each image is-

TABLE 5.

Image ID	***Red***	***Green***	***Blue***	***Resul-tant Red***	***Resul-tant Green***	***Resul-tant Blue***
46	4964205	4139370	3883431	25.49	2.48	1.75
48	4971737	4142263	3886664	25.53	2.48	1.75
50	4964423	4137651	3882170	25.49	2.48	1.75

TABLE 6.

Image ID	***Red***	***Green***	***Blue***	***Resultant Red***	***Resultant Green***	***Resultant Blue***
47	5119537	4366206	4100837	26.30	2.62	1.85
49	5126690	4369915	4099801	26.33	2.62	1.85

Let the user query is for searching the same teddy looks like fig 3(b). There are three same kind of teddy. The resultant value of Red, Green, and Blue of image 3(b) is 25.49, 2.48 and 1.75.

Now considering the Threshold value (0.2) for retrieval of exactly same image. After applying the Threshold value the result will be as follows-

25.29 ≤ Resultant Red ≤25.69
02.28 ≤ Resultant Green ≤02.68
01.55 ≤ Resultant Blue ≤01.95.

By using the above formulae, the searching result will give the viewer exactly same of fig3 (b), i.e. fig 3(d).

IV CONCLUSION

In this paper we reflect a new scheme for color image indexing and retrieval. Both the color and size features have been utilized to get better accuracy. The advantage of the mean and standard deviation of the color technique over proposed database is that, it's much simpler and gives fast yet reasonably accurate results.

V REFERENCES

[1]. C.Saha, A. Chanda, G.Bag and S.Mitra," A Content Based Algorithm for Indexing and Retrieval of Image Databases", Electronics Research & Development Centre of India Proceedings of IEEE ACE 2002, Kolkata.

[2]. S. Jeong and R. M. Gray, "Image Retrieval using Color Histograms Generated by Gauss Mixture Vector Quantization", Comput. Vision Image Understand., no. 94, pp.44-66, 2004.

[3] A. Torralba, "How Many Pixels Make an mage?", Visual Neuroscience,vol. 2, no.1, pp. 123-131, 2009.

[4] D. Cerra and M. Datcu, "Image Classification using Data Compression Based Techniques", Proceedings IGARSS'08, vol.1, pp. 237-240, 2008.

[5] Healey, G. (1992), Segmenting images using normalized color, IEEE trans. Syst.Man Cybernet. 22 (1), 64-73

[6] Mlsna, P.A, Rodriguez, J.J, "Efficient indexing of multi-color sets for content based image retrieval", The 4th IEEE Southwest Symposium on Image Analysis and Interpretation, Texas, USA, 2000, pp.116.