IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)
e-ISSN: 2278-1676,p-ISSN: 2320-3331, Volume 7, Issue 1 (Jul. - Aug. 2013), PP 113-118
www.iosrjournals.org

A Mat Lab built software application for similar image retrieval

Akhilesh Mehta[1], Rakesh Pandey[2], Parul Kansal[3]
[1,2](Ece M.Tech,Btkit Dwarahat/ Uttarakhand Technical University, India)
[3](Assistant Prof.,Btkit Dwarahat/ Uttarakhand Technical University, India)

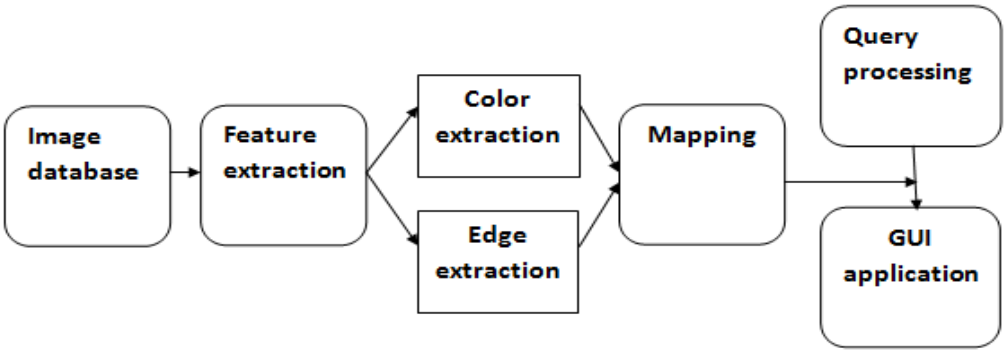
Abstract : *The paper introduced the algorithm of image retrieval which combine the features of color, shape and so on , and then we do sorting in the final result. Firstly, the internal and exterior normalization of multi-features are studied and analyzed, and introduced the normalization algorithm in detail. Secondly, the retrieval methods of the multi-features are showed. Lastly, adopted the Euclid distance metric, and used the image database. Our final result is a* ***Mat Lab*** *built software application, with an image database, that utilized shape and colour features of the images in the database as the basis of comparison and retrieval. The structure of the final software application is illustrated.*

Keywords: *image retreival, feature extraction, indexing, edge point detecion, multi-feature, features.mat*

I. Introduction

In fact, any the image retrieval based on single feature has certain limitation, it cannot the comprehensive descript the image contain content [1]. If the multi features of image make in the integration image retrieval system, the effect of the image retrieval will obtain enormous enhancement [2]. But description methods such as color, shape had reflected the image feature from the different angle, how does organize these characteristics, enables them to defer to the user's request to merge each kind of feature, this is a question which needs to study

Proposed Algorithm



Figure(1)-Block Diagram of Image Retrieval

II. Image Database

In image database we store all the images in which we want to apply our algorithm.

Figure2. Some images from image database

III. FEATURE EXTRACTION AND EDGE DETECTION

AS shown in figure it has two parts. Viz. edge detection and color extraction.

Edge detection: *First the RGB image is reduced to thumbnail size and median filtered, and then we use Four masks as shown in figure(3) to detect edges.*

1	1	1
0	0	0
-1	-1	-1

(a)Horizontal diagonal Mask

1	0	-1
1	0	-1
1	0	-1

(b) Vertical Mask

1	1	0
1	0	-1
0	-1	-1

(c) Diagonal Mask

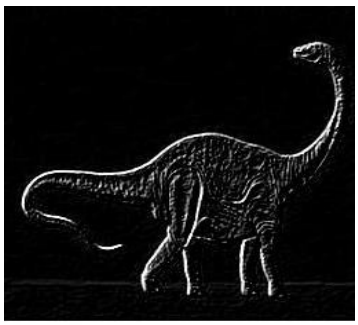
0	1	1
-1	0	1
-1	-1	0

(d) Anti-Mask

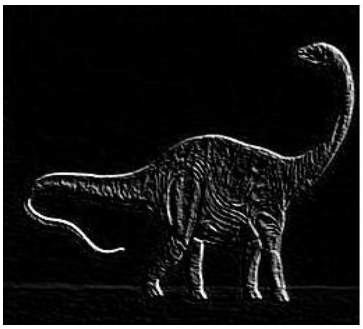
Figure (3)

We have taken an image-

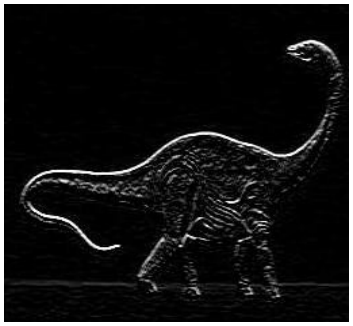
Then the result of four masks is as follow-



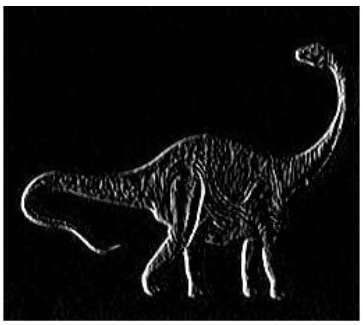
Horizontal edges



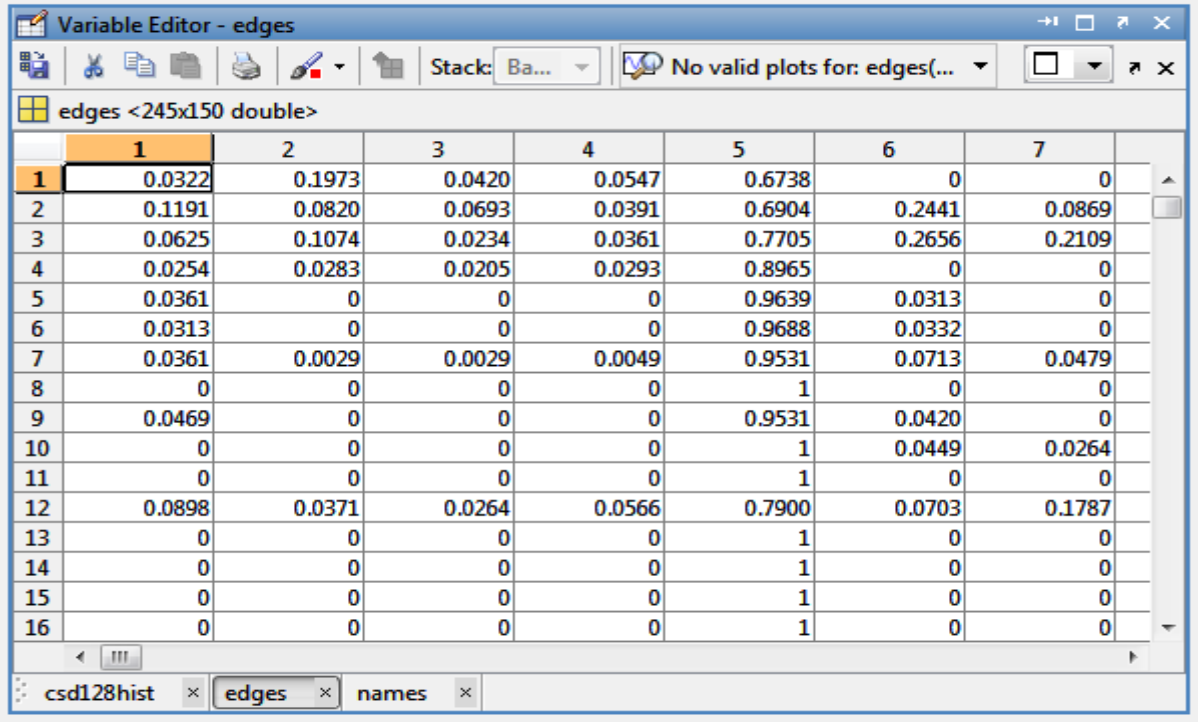
Vertical edges



Diagonal edges



Anti-diagonal edges



	1	2	3	4	5	6	7
1	0.0322	0.1973	0.0420	0.0547	0.6738	0	0
2	0.1191	0.0820	0.0693	0.0391	0.6904	0.2441	0.0869
3	0.0625	0.1074	0.0234	0.0361	0.7705	0.2656	0.2109
4	0.0254	0.0283	0.0205	0.0293	0.8965	0	0
5	0.0361	0	0	0	0.9639	0.0313	0
6	0.0313	0	0	0	0.9688	0.0332	0
7	0.0361	0.0029	0.0029	0.0049	0.9531	0.0713	0.0479
8	0	0	0	0	1	0	0
9	0.0469	0	0	0	0.9531	0.0420	0
10	0	0	0	0	1	0.0449	0.0264
11	0	0	0	0	1	0	0
12	0.0898	0.0371	0.0264	0.0566	0.7900	0.0703	0.1787
13	0	0	0	0	1	0	0
14	0	0	0	0	1	0	0
15	0	0	0	0	1	0	0
16	0	0	0	0	1	0	0

Figure(4)- edge points of images present in database

IV. COLOR EXTRACTION

The color histogram is a method for describing the color content of an image, it counts the number of occurrences of each color in an image. For color extraction we convert RGB image to HSV image.

RGB to HSV

In order to use a good color model for a specific application, conversion between color models is necessary. A good color model for an image retrieval system should preserve the perceived differences in color.

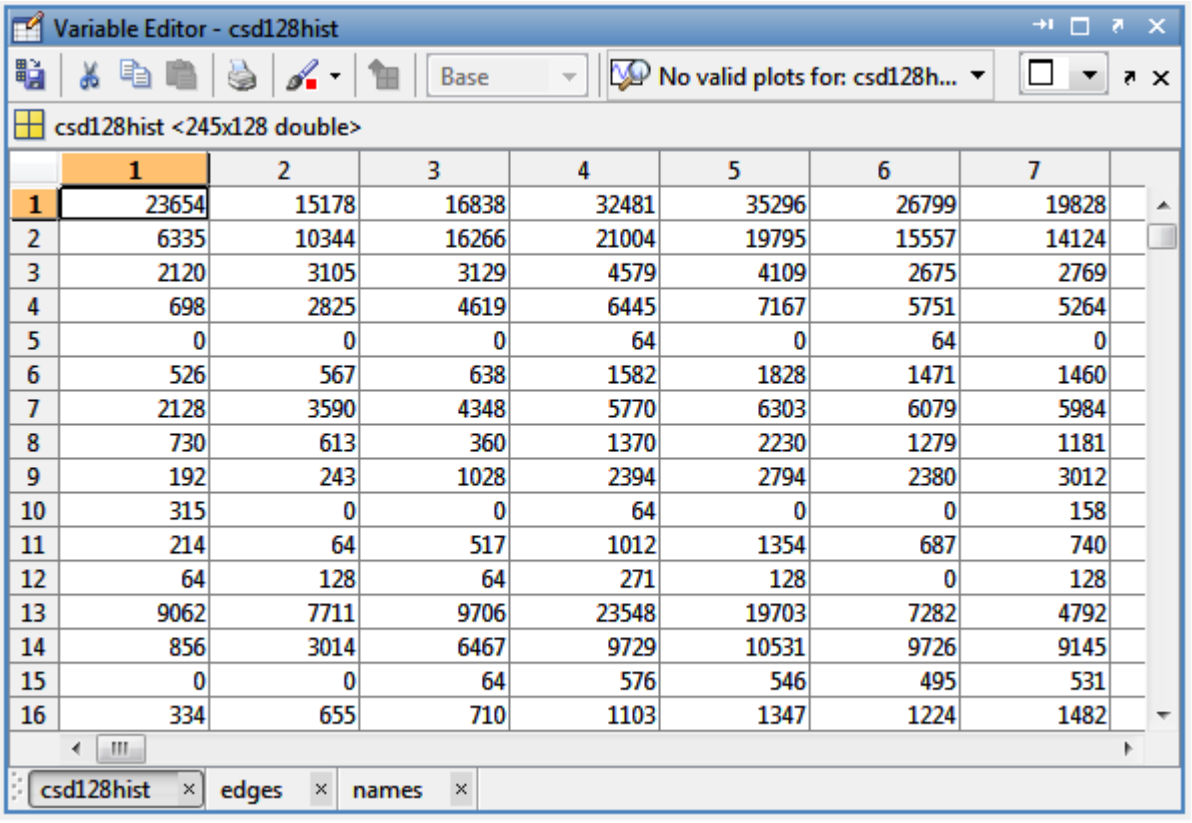
$$H = cos^{-1}\left\{\frac{\frac{1}{2}[(R-G)+(R+B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right\}$$

$$S = 1 - \frac{3}{R+G+B}[\min(R,G,B)]$$

$$V = \frac{1}{3}(R+G+B)$$

Color quantization

In order to produce color histogram, color quantization is often applied. Color quantization is the process to reduce the number of colors employed to represent an image. A quantization scheme is determined by the color model and the segmentation (i.e., split up) of the color model used. As we said before, usually color models represent a color in the form of tuples (generally of three) .By applying a standard quantization scheme to a color model, each axis is divided into a certain number of fractions. When the axes are divided into k, l, and m parts, number (n) of the colors used to represent an image will be: n= k.l.m. A quantization of color model in n colors is often referred to as a n-bins quantization scheme. We have taken 128bin histogram here.



	1	2	3	4	5	6	7
1	23654	15178	16838	32481	35296	26799	19828
2	6335	10344	16266	21004	19795	15557	14124
3	2120	3105	3129	4579	4109	2675	2769
4	698	2825	4619	6445	7167	5751	5264
5	0	0	0	64	0	64	0
6	526	567	638	1582	1828	1471	1460
7	2128	3590	4348	5770	6303	6079	5984
8	730	613	360	1370	2230	1279	1181
9	192	243	1028	2394	2794	2380	3012
10	315	0	0	64	0	0	158
11	214	64	517	1012	1354	687	740
12	64	128	64	271	128	0	128
13	9062	7711	9706	23548	19703	7282	4792
14	856	3014	6467	9729	10531	9726	9145
15	0	0	64	576	546	495	531
16	334	655	710	1103	1347	1224	1482

Figure(5)- color histogram of images prersent in database

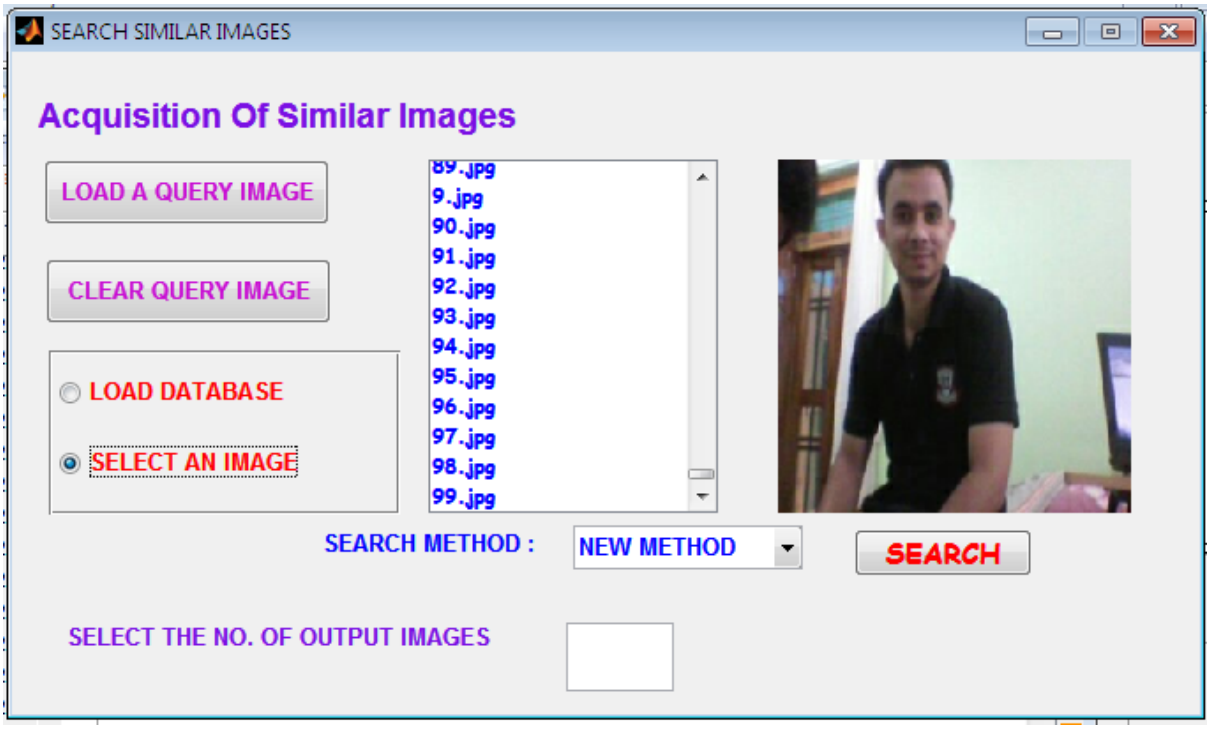
Features.mat: *to save time by calculating image content for each query processing we create a file in mat lab, which saves the edge features and color features of the database.*

5.Query processing

Now we have created a features database in which features of all the image of image database are saved.
Now we input a query image. The system first extract the edge feature and color feature of the query image, then compare it with all the other images present in the database. This is done by using Euclidean distance method. Thus in last we get images which are similar to the query image.
The features saves in a matrix form as shown in figure and then we use a function of mat lab which is used to compare the features of images.

D = pdist(X)

It computes the Euclidean distance between pairs of objects in m-by-n data matrix X. Rows of X correspond to observations, and columns correspond to variables. D is a row vector of length m(m–1)/2, corresponding to pairs of observations in X. The distances are arranged in the order (2,1), (3,1), ..., (m,1), (3,2), ..., (m,2), ..., (m,m–1)). D is commonly used as a dissimilarity matrix in clustering or multidimensional scaling.



Figure(6)- Query image

Now we have the images which are similar to the query image after comparing the features as in array form. The next step is to sort this to get the best matched image in descending order i.e. at the top we get the images which are most similar to the query image.

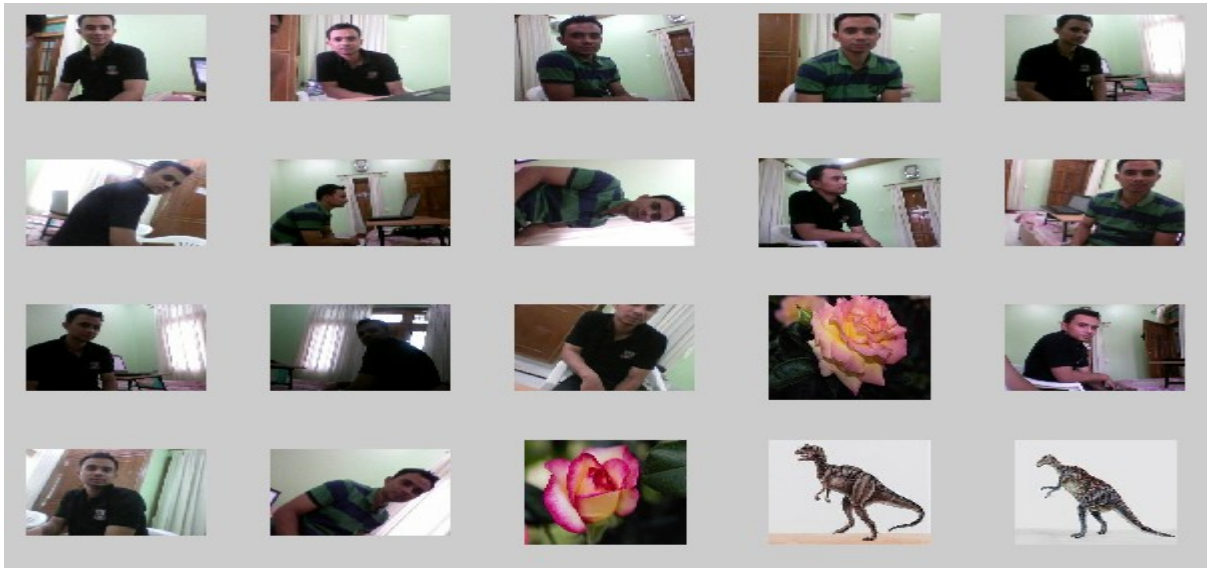


Figure (7) - Final result

V. Conclusion

The paper have analyzed and studied some retrieval questions about multi-feature fusion, for instance processing in multi- feature vector and so on. Finally uses the Euclidean space in the image database to retrieve and confirm the multi-features image retrieval method. From the experimental result, we may see that three kinds of features fuses may obtain the good retrieval effect. We have also created a features.mat file which contains the features of the image database, which save time of calculating features in every operation. But the algorithm also has certain insufficiency, for deficiency of detecting emotions such as emotions of human faces as smile, anger etc. This question is the next research problem.

References

[1] Y.Rui,T.S.Huang,S.F.Chang. Image retrieval: Past, present,and future. Journal of Visual Communication and Image Representation, 1999.10:1~23.
[2] S.Aksoy,R.M.Haralick. Feature normalization and like hood-based similarity measure for image retrieval.PRL.2001.22 (5):563~582.
[3] Guo xiao juan Research on the Content-Based Image Retrieval. A Master degree dissertation of Northwest University .2007.
[4] Chen wei,Xiao Guo-qiang. Image retrieval algorithm based on the multi-features [J]. Computer Engineering & Design,2008(17)
[5] S.Aksoy,R.M.Haralick. Feature normalization and like hood-based similarity measure for image retrieval.PRL.2001.22 (5):563~582.
[6] M.Ortega,et.al. Supporting similarity queries in MARS.ACM Conf.on Multimedia.p:403~413.