ISSN:0975-9646

D. D. Pukale et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (3) , 2015, 2645-2652

Image Matcher – Content Based Image Retrieval System Using Image Sub Blocks and Indexing

Prof. D. D. Pukale, Varsha H. Sakore, Nilima K. Shevate,. Nutan D. Pawar, Priyanka N. Shendage

Department of Computer Engineering,
Bharati Vidyapeeth's College of Engineering for Women,
Pune, Maharashtra, India

Abstract— **Because of content based techniques, image retrieval is becoming efficient. The goal of CBIR is to extract features of images and to compare them to retrieve similar images. We are focussing on features of images such as colour, shape and texture. In our proposed system, the efficiency is improved by use of different efficient feature extraction algorithms – RGB to Lab conversion for colour feature extraction, Canny edge detection algorithm for shape feature extraction and Framelet transform for texture feature extraction. We applied these feature extraction algorithms on sub blocks of images. For similarity matching of extracted features, Manhattan distance method is used. Also, to improve the speed of image retrieval, indexing and serialization techniques are used. The proposed method is compared with existing systems [1], [2] and we can see an increase in average precision from 45.06% to 64.1%.**

Keywords— **feature extraction, Lab, Canny edge detection, Framelet transform, Manhattan distance, indexing, serialization.**

I. INTRODUCTION

Our paper is divided into 5 sections. First section gives introduction about content based image retrieval process. Second section gives literature review and third describes the proposed system. Fourth section displays the result and fifth section draws the conclusion.

Digital image storage is becoming more popular than other image storage techniques. Due to large number of huge image databases, image retrieval is the very important function that every digital image storage technique must support. There are two main methods of image retrieval: Text Based Image Retrieval (TBIR) and Content Based Image Retrieval (CBIR).

TBIR mainly deals with tags, textual information and captions of images. But this requires manual annotation of images and it is very time consuming process. Also due to different human perceptions, there may be inconsistency in image annotation [3].

Hence, CBIR is gaining more popularity. CBIR is the process of retrieving similar images from a large image database based on contents of images such as colour, texture, shape, etc [4]. Use of these all hybrid features gives better results [5].

Image retrieval process is a process including 3 steps: First step is pre - processing on images. Next step is feature extraction and last step is similarity matching using extracted feature vectors.

II. LITERATURE REVIEW

Colour is a vital feature of colour images. Various colour models are there like, RGB, CMYK, CIE Lab, CIE XYZ, HSV, HSL, etc. Madhura C and Dheeraj D proposed that, converting RGB images into some other colour space like Gray, Lab, YCbCr, CMY, HSV, HIS and then processing them gives good results [6]. Lab colour model contains L factor i.e. Lightness factor, which is similar to human perception of lightness and hence provides very good results.

As we know, edge detection is very important, because good edge detection improves other processing also. There are many edge detection algorithms like Local Threshold, Marr-Hilderth edge detector, Boolean Function based edge detector and canny edge detector. Raman Maini and Dr. Himanshu Aggarwal compared different edge detection algorithms and they reached to the conclusion that Canny's edge detection algorithm performs better than all other edge detecting operators under almost all scenarios [7].

Texture means visual patterns that have properties of uniformity. It is a native property of almost all surfaces, like trees, hair, bricks, fabric, cloud, etc. Framelets are two or more high frequency filter banks, which can produce more sub bands on decomposition. This can achieve better time frequency localization [8].

We need to calculate the distance between various images for finding the similar images. There exist some distance metrics to calculate the distance between images of same size, Euclidean Distance, Manhattan Distance, Modified Euclidean Distance based on histogram index, Vector Cosine Angle Distance, etc. Among these, Euclidean and Manhattan distance metrics give metric of dissimilarity. Modified Euclidean Distance based on histogram index and Vector Cosine Angle Distance metrics give the metric of similarity [9].

III. PROPOSED SYSTEM

From literature review, it is clear that using hybrid feature vectors is always beneficial. Also, we could understand different efficient feature extraction algorithms. So, our proposed system is utilizing the hybrid feature vectors and for extraction of these feature vectors following feature extraction algorithms are used:

- RGB to Lab conversion: Colour feature extraction.
- Canny edge detection algorithm: Shape feature extraction.
- Framelet transform: Texture feature extraction.

For similarity matching, we have used Manhattan Distance method.

Also, for increasing the speed of image retrieval, indexing and serialization is used.

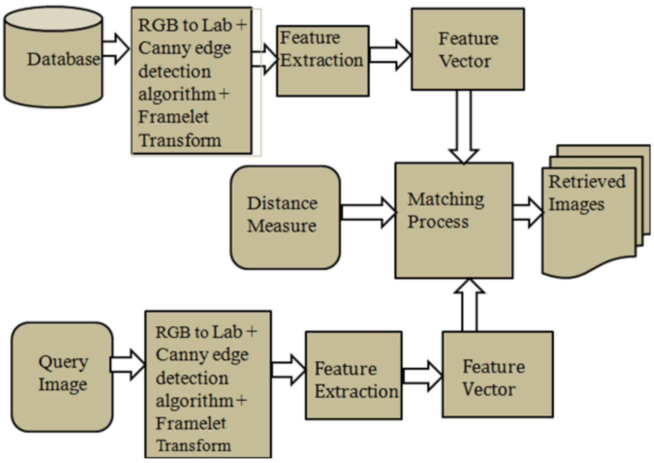

Fig. 1 Basic system architecture

Fig. 1 shows the basic system architecture. From this architecture, we can get the flow of CBIR clearly. Firstly, feature extraction algorithms are applied on database images and feature vectors are stored along with images. While retrieving similar images, same feature extraction algorithms are applied on query image and feature vectors are generated. Finally, extracted feature vectors of query image and stored feature vectors of database images are compared using distance measure. Based on the distance, similar images are retrieved.

A. *Feature Extraction*

Feature extraction is very important step in CBIR. But before applying the feature extraction algorithms, we have resized every image as 300 * 300 resolution image. So we can apply distance measure to these images. Then we divided these images into 100 equal sized sub blocks. Then all feature extraction algorithms are applied and feature vectors of all blocks are stored into database.

1) Colour Feature Extraction: Since, we know converting RGB images to Lab gives better results. Hence, we used RGB to Lab colour conversion for colour feature extraction. CIE Lab colour space is a colour space having dimensions as: L for lightness, a and b are colour - opponent dimensions.

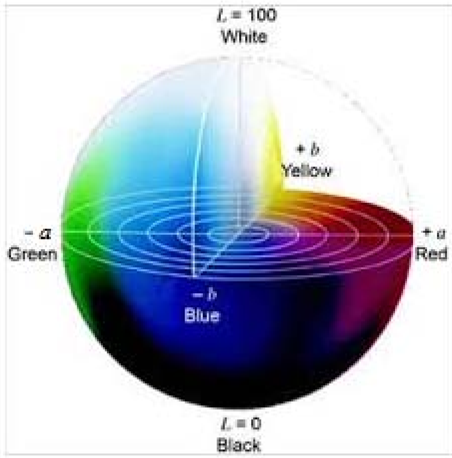

Fig. 2 Lab colour space

As shown in the fig. 2, ‘L’ varies from 0 – Black to 100 – White. ‘a’ varies from negative – Green to positive – Red. ‘b’ varies from negative - Blue to positive – Yellow.

CIE Lab colour space is computed using CIE XYZ colour space, but it has following advantages:

- It is perceptually more uniform. Perceptually uniform means a change of some amount in colour value should produce a change of same visual importance [10].
- It is device independent.

RGB is device dependent and Lab is device independent. So, we cannot convert RGB directly Lab. We have to convert RGB to XYZ and then XYZ to Lab.

Algorithm:

i. Obtain R, G and B values from the image pixels.
ii. Convert these values into X, Y and Z values.
iii. Finally, convert X, Y and Z values into L, a and b values.

We applied RGB to Lab conversion on sub blocks of image. Then calculated sum of L, a and b values of all pixels of that block and stored those values as feature vectors. So we get 3 values for each block and 300 values for an entire image.

100 sub blocks – For each block 3 sum values – L, a and b sums = 300 values for each image.

2) Shape Feature Extraction: The aim of Canny edge detection algorithm is to have good detection (minimum number of false edges), good localization (closeness of the real edge and the detected edge) and minimal response (one edge should be detected only once).

Canny edge detection algorithm:

i. Image smoothing by Gaussian Filter: To filter out any noise in the original image before trying to locate and detect any edges.
ii. Gradient calculation using Sobel operator: To find the edge strength by taking the gradient of the image.

iii. Non-maxima suppression: Non – maximum pixel values are suppressed.
iv. Hysteresis: Hysteresis is used as a means of eliminating streaking. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold.
v. Final binary image: The final output is a binary image.

We applied the Canny edge detection algorithm on the sub blocks of image. Gradient magnitudes and angles are used as feature vectors. So for each block sums of these two values for all pixels is stored as feature vectors. So, 100 blocks with 2 feature vectors yields 200 feature vectors for every image.

3) Texture Feature Extraction: Texture means properties that represent the structure or surface of an object. Two main texture description approaches are there: statistical approach and syntactic approach. We followed statistical approach i.e. Framelet Transform.

The algorithm of Framelet Transform:

a. Decompose every image into Framelet Transform Domain.

b. Calculate the energy, standard deviation of Framelet transform decomposed image using the formulae:

Energy = $1/M*N \sum_{i=1}^{M} \sum_{j=1}^{N} | W_k(i, j) |$

Standard Deviation (σk) = $\sqrt{} 1/M*N \sum_{i=1}^{M} \sum_{j=1}^{N} (W_k (i, j) - \mu_k)$ ^2

μk is the Mean value of the kth Framelet transform sub band.

Wk is the Coefficient of kth Framelet transform sub band.

$M\times N$ is the size of the decomposed sub band.

c. The resulting feature vectors are

f = [σ1, σ2, ..., σn, E1, E2, ..., En].

We have applied this algorithm on sub blocks of images. For each block, sum of energy and standard deviation values of all pixels are stored as feature vectors. So, 100 blocks give 200 feature vectors.

4) Grayscaling: Images are gryascaled with the following formula:

gs = r * 0.33 + g * 0.56 + b * 0.11

On each pixel, this formula is applied and sum of grayscaled values of all pixels in a block is stored as a feature vector. So, for each image 100 blocks and 100 grayscaling feature vectors are stored.

B. Similarity Matching

Manhattan distance calculates sum of difference in every dimension of vector space. It is sum of absolute differences of corresponding components [9].

Manhattan distance is also known as 'City Block Distance' or 'L1 distance'. The City block distance between two points, *a* and *b*, with *k* dimensions is calculated as:

$\sum_{j=1}^{k} | a_j - b_j |$

For matching similarity between query image and database image, same feature extraction procedure is applied on query image also. Then feature vector values are compared block wise. So, first block of query image is compared with first block of database image. As we are using **Manhattan Distance,** the feature vector comparison process is as follows:

Distance = |Feature vector value of query image| - |Feature vector value of database image|.

If this Distance is less than the limit value, then that block is considered to be matched block and total number of matched blocks is nothing but the similarity.

C. Indexing and Serialization

Serialization: In computer science, serialization is the process of converting an object into a sequence of bits so that it can be persisted on a storage medium (such as a file, or a memory buffer). When the resulting series of bits is reread according to the serialization format, it can be used to create a semantically identical clone of the original object.

Here, we are storing each image entry object in serialized form. This reduces the storage and retrieval time.

Indexing: To avoid searching of each and every image for similarity, we have done some indexing like facility to speed – up the process.

Along with the above mentioned feature vectors, we are storing dedicated features for indexing of every image as:

i. Dominant Colour: According to colour dominancy, images are grouped under red, green and blue dominant images.
ii. Grayscale index: Based on range of grayscale values, 5 groups are formed and group numbers 1 - 5 are used as indexes. For example, images having sum of grayscale values less than 50 are in group 1, images having sum of grayscale values between 51 and 100 are in group 2, and so on.
iii. Magnitude index: Based on range of magnitude values, 5 groups are formed and group numbers 1 - 5 are used as indexes. For example, images having sum of magnitude values less than 1000 are in group 1, images having sum of magnitude values between 1001 and 2000 are in group 2, and so on.
iv. Angle index: Based on range of angle values, 5 groups are formed and group numbers 1 - 5 are used as indexes. For example, images having sum of angle values less than 1 are in group 1, images having sum of angle values between 1 and 2 are in group 2, and so on.
v. Energy index: Based on range of energy values, 5 groups are formed and group numbers 1 - 5 are

used as indexes. For example, images having sum of angle values less than 20 are in group 1, images having sum of angle values between 21 and 40 are in group 2, and so on.

vi. Standard deviation index: Based on range of standard deviation values, 5 groups are formed and group numbers 1 - 5 are used as indexes. For example, images having sum of standard deviation values less than 5 are in group 1, images having sum of standard deviation values between 6 and 10 are in group 2, and so on.

Now, query image will first check that both query image and database image fall into same group of particular feature and then start proceeding.

This will save time in processing irrelevant images.
Total feature vector values for an image = 300 for color + 100 for grayscale + 200 for shape + 200 for texture + Color index + Shape index + Grayscale index + Texture index.

IV. RESULTS AND ANALYSIS

We have used the benchmark database i.e. Wang database for comparison of results with other systems and for texture feature, we have used Bortratz database.

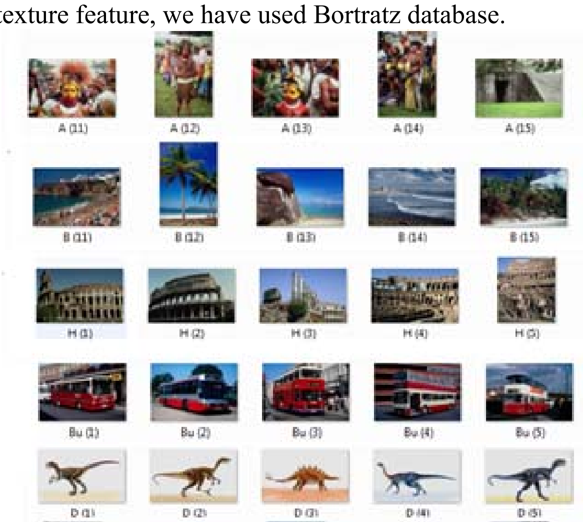

Fig. 3 Sample images from Wang database

Fig. 3 shows the sample images from Wang database. This database contains 1000 images of 10 categories.

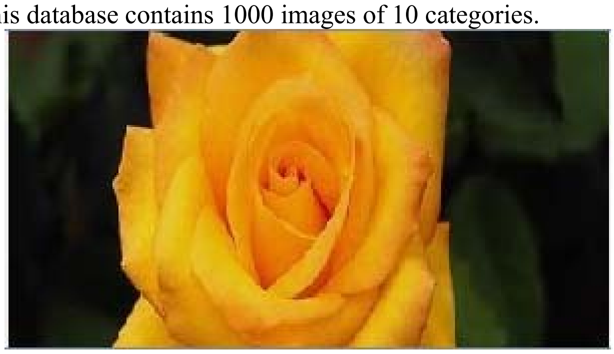
Fig. 4 Original Image

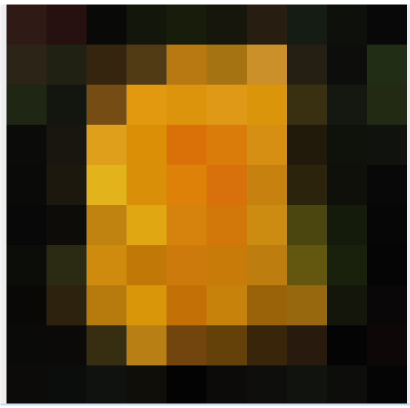
Fig. 5 Colour based signature after dividing it into 100 sub blocks

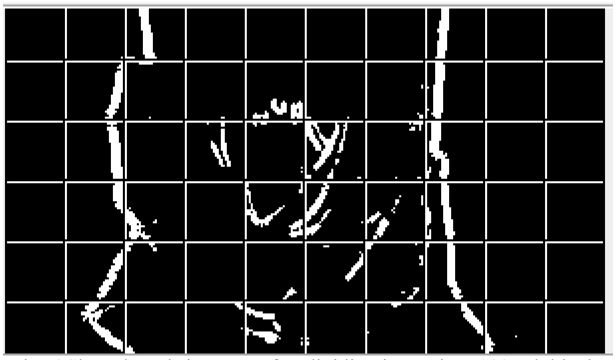
Fig. 6 Shape based signature after dividing image into 100 sub blocks

Fig. 4 shows the original image and Fig. 5 and 6 show colour and shape based signatures of that image when divided into 100 sub blocks.

For CBIR performance evaluation, basically two metrics are used: Precision and recall.

Precision = Number of relevant images retrieved / Total number of images retrieved

Recall = Number of relevant images retrieved / Total number of relevant images in the database.

Various CBIR methods have been proposed. The *Kavitha* methodology is based on CBIR technique that uses the HSV colour space as colour feature and the GLCM's Energy, Contrast, Entropy and Inverse Difference as the texture features. Another methodology of CBIR proposed in [2] is based on colour and texture feature with a new colour feature called Average Colour Dominance (ACD). Hence in this section, we are comparing our system with these two methodologies and the comparison is carried out by comparing precision and recall.

Also, we are comparing average precision and recall of individual feature extraction method like colour, texture, shape etc. and we finally found that combined results are more precise.

We calculated average precision and recall as:

We randomly selected 5 images from each category. Then average precision and recall values are calculated based on top 20 retrieved images.

RESULTS

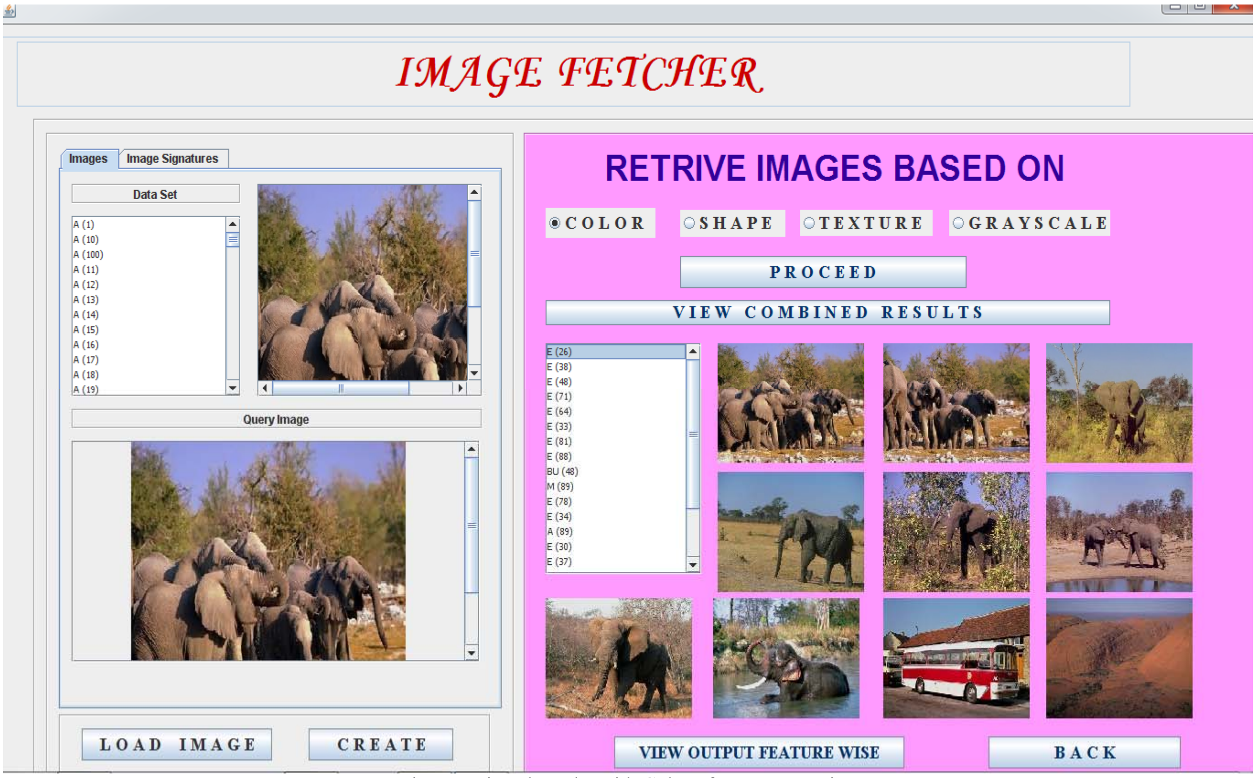

Fig. 7Retrieved results with Colour feature extraction

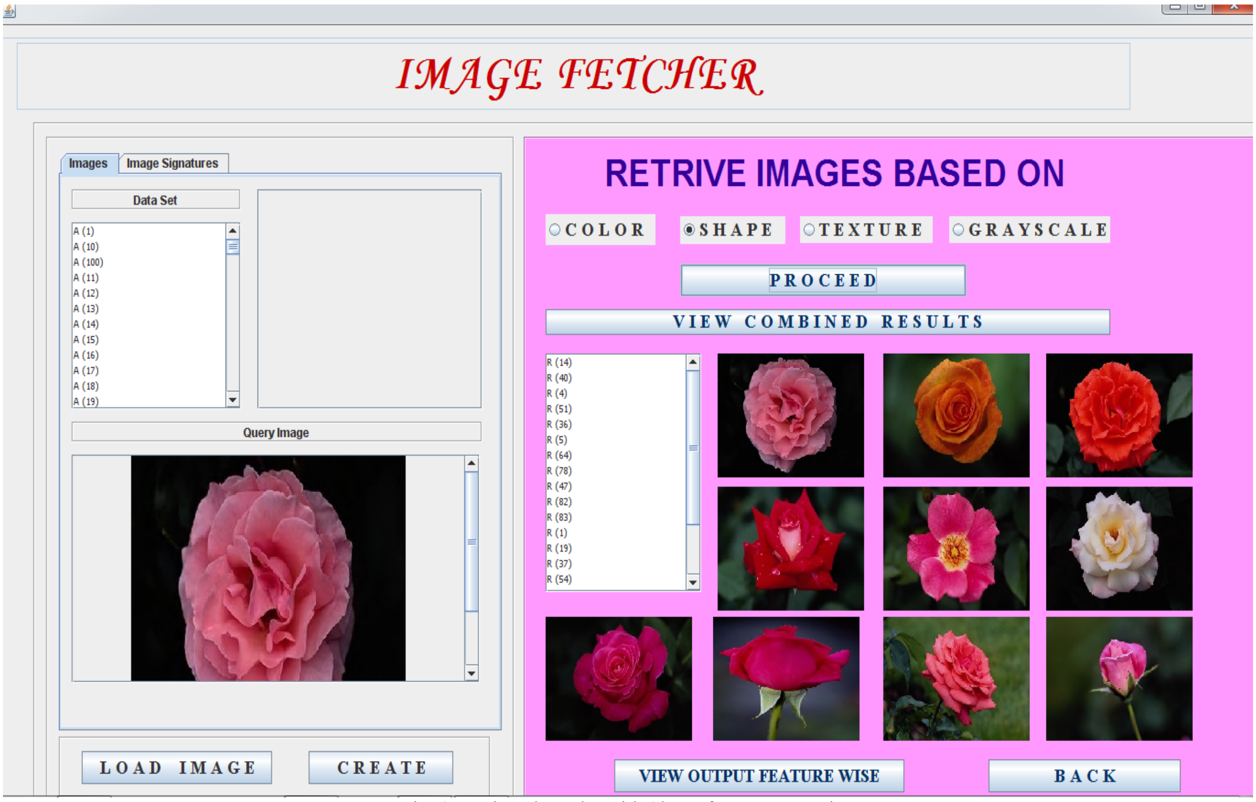

Fig. 8 Retrieved results with Shape feature extraction

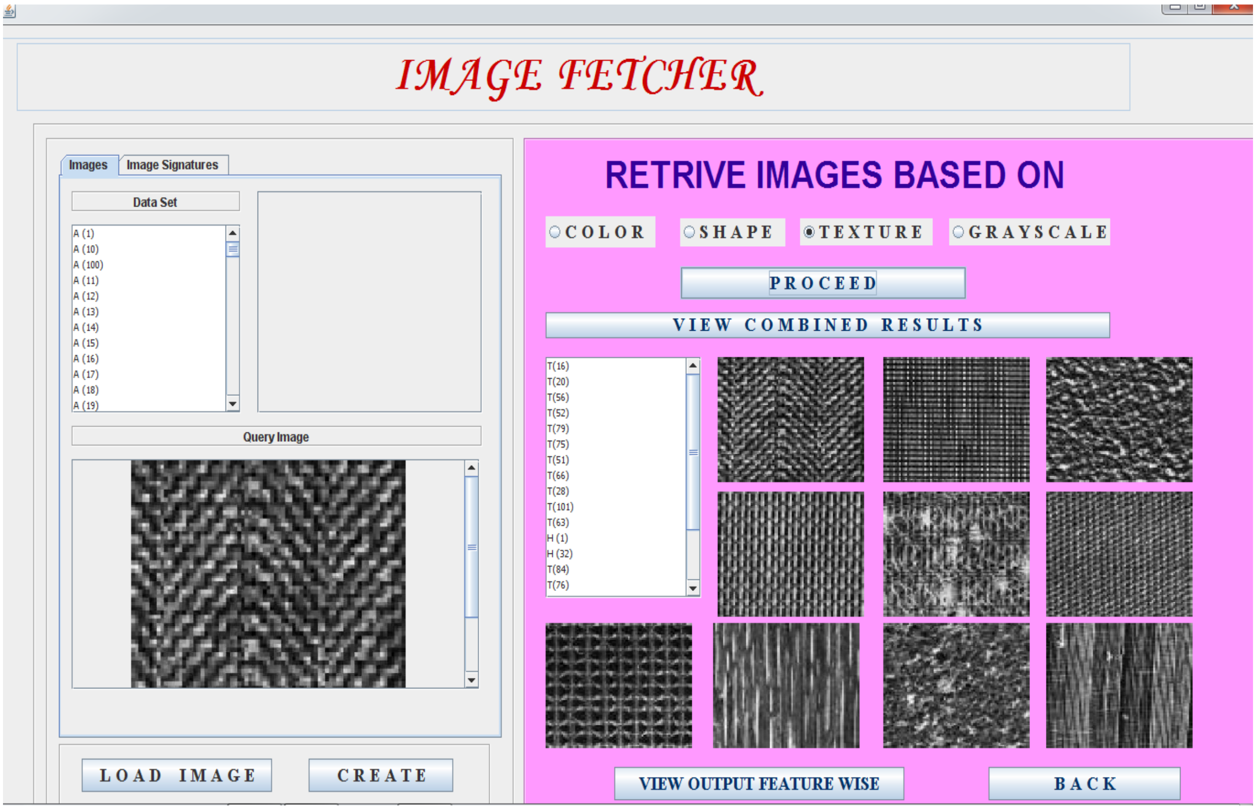

Fig. 9 Retrieved results with texture feature extraction

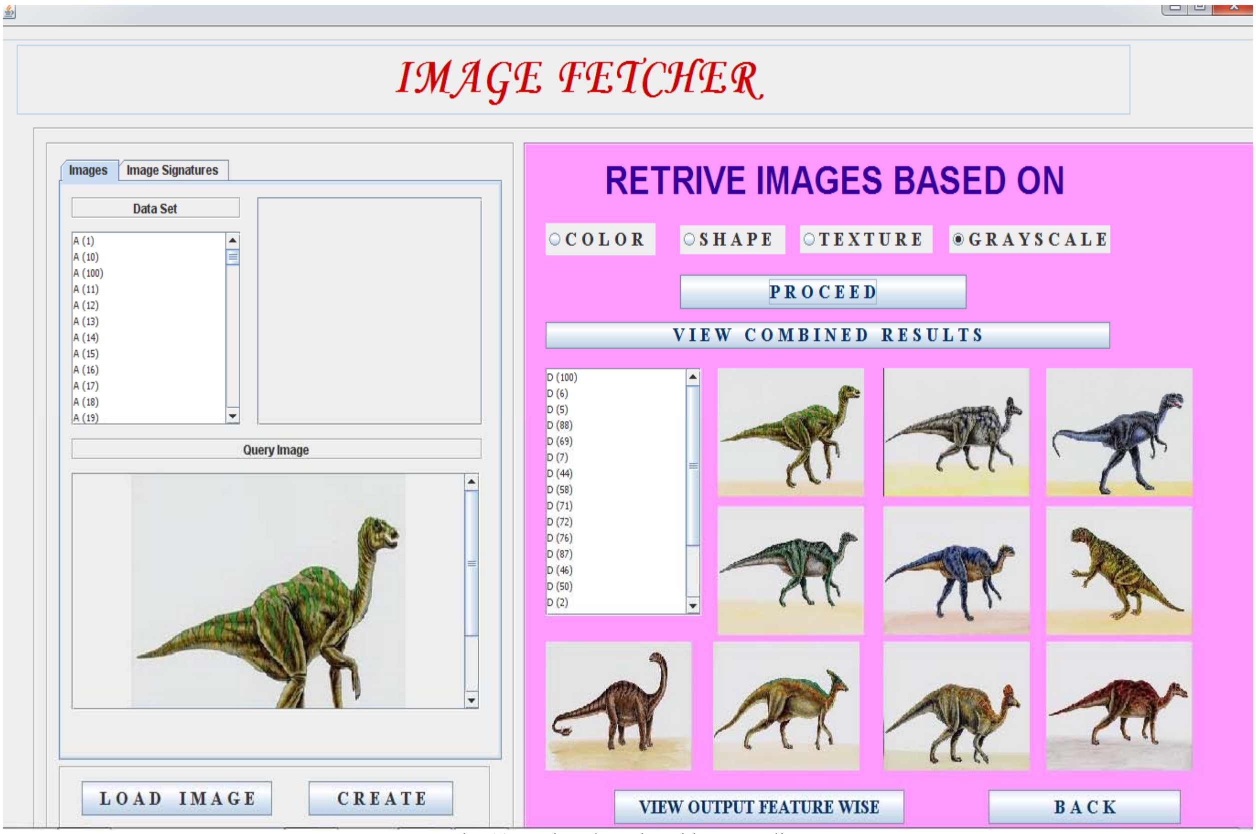

Fig. 10 Retrieved results with grayscaling

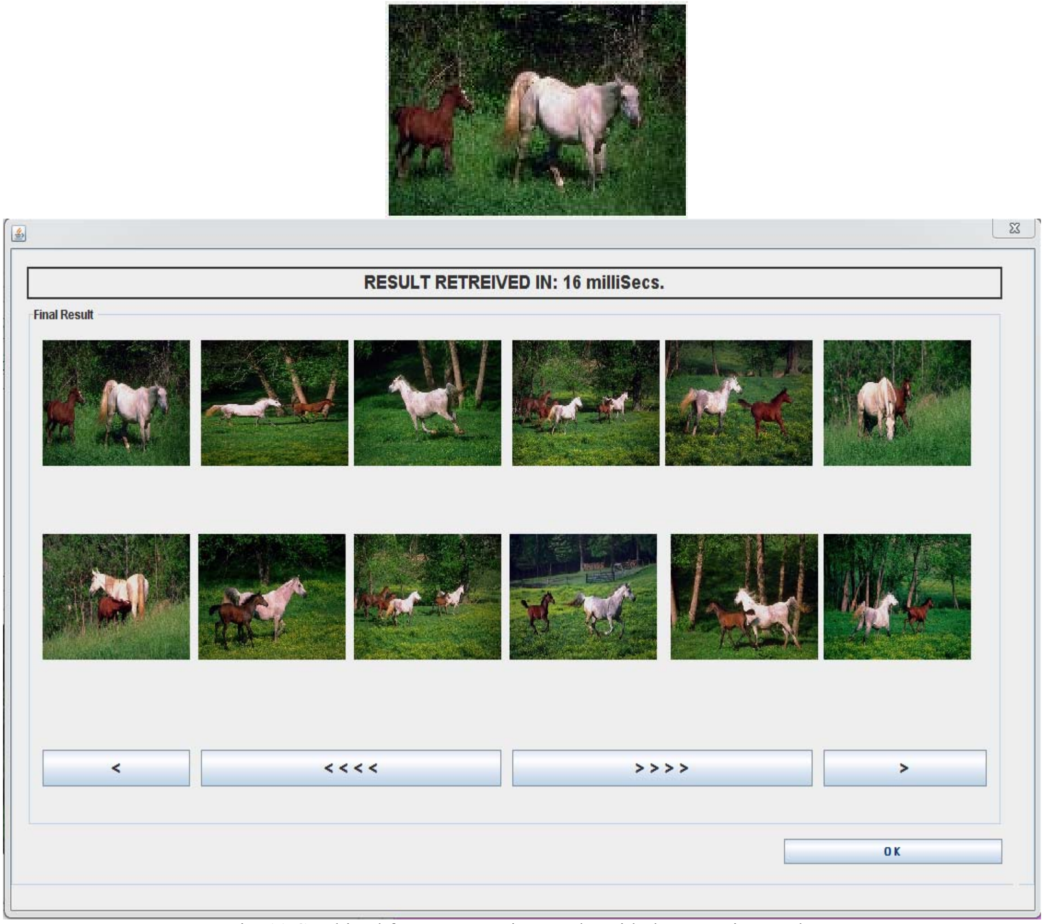

Fig. 11 Combined feature extraction results with the query image shown

Following table shows the average precision and recall using individual feature extraction methods and using combined feature extraction methods.

TABLE I
PRECISION - RECALL ANALYTICAL TABLE

PR – PRECISION AND RE – RECALL

Feature Extraction Method	**Number of images**	**Colour**		**Shape**		**Texture**		**Grayscale**		**Combined**		
Images		**Pr**	**Re**	**Pr**	**Re**	**Pr**	**Re**	**Pr**	**Re**	**Pr**	**Re**	**Time (ms)**
African People	100	39	7.8	26	5.2	35	7	20	4	43	8.6	19
Beaches	100	31	6.2	16	3.2	54	10.8	47	9.4	64	12.8	28
Monuments	100	25	5	24	4.8	44	8.8	27	5.4	43	8.6	31.4
Buses	100	28	5.6	18	3.6	38	7.6	31	6.2	46	9.2	38
Dinosaurs	100	100	20	78	15.6	100	20	100	20	100	20	43.6
Elephants	100	76	15.2	18	3.6	65	13	55	11	61	12.2	46.6
Flowers	100	97	19.4	83	16.6	96	19.2	94	18.8	100	20	37.4
Horses	100	73	14.6	65	13	96	19.2	65	13	97	19.4	25.2
Mountains	100	35	7	17	3.4	38	7.6	31	6.2	43	8.6	22.2
Foods	100	29	5.8	21	4.2	30	6	15	3	44	8.8	19.2
Average / Total	1000	**53.3**	**10.66**	**36.6**	**7.32**	**59.6**	**11.92**	**48.5**	**9.7**	**64.1**	**12.8**	**31.06**

From the above results, it is clear that combined feature extraction shows significant increase in precision and recall. Also, due to indexing and serialization, retrieval time is around only 30 – 40 milliseconds.

TABLE II
COMPARISON RESULTS

PR – PRECISION AND RE – RECALL

Image category	HSV + GLCM -Kavitha methodology		HSV + ACD + GLCM		Proposed System		
	Pr (%)	Re (%)	Pr (%)	Re (%)	Pr (%)	Re (%)	Time (ms)
African People	29.1	5.82	31.0	6.2	43	8.6	19
Beaches	27.9	5.58	37.0	7.4	64	12.8	28
Monuments	21.5	4.3	23.7	4.74	43	8.6	31.4
Buses	36.9	7.38	31.4	6.28	46	9.2	38
Dinosaurs	90.55	18.11	95.2	19.04	100	20	43.6
Elephants	37.7	7.54	37.9	7.58	61	12.2	46.6
Flowers	58.6	11.72	70.1	14.02	100	20	37.4
Horses	59.3	11.86	66.2	13.24	97	19.4	25.2
Mountains	26.9	5.38	32.4	6.48	43	8.6	22.2
Foods	19.9	3.98	25.7	5.14	44	8.8	19.2
Average	**40.8**	**8.16**	**45.06**	**9.012**	**64.1**	**12.8**	**31.06**

It is clear that our proposed system shows increase in precision and recall.

V. CONCLUSIONS

The purpose of this system is to improve precision. Hence we have combined different efficient feature extraction algorithms. Based on the results, it is clear that combined results are far better than individual results of separate feature extraction algorithm. Also, we have compared our methodology with two proposed methodologies showing increase in precision from 45.06% to 64.1%. Because of the use of indexing and serialization techniques, the average image retrieval time is just 30 – 40 milliseconds.

REFERENCES

[1] Ch.Kavitha, Dr.B.Prabhakara Rao, Dr.A.Govardhan, *Image Retrieval Based On Color and Texture Features of the Image Sub-blocks,* International Journal of Computer Applications (0975 – 8887), Volume 15– No.7, February 2011, Pages – 33 – 37.

[2] Tamer Mehyar, Jalal Omer Atoum, *An Enhancement on Content-Based Image Retrieval using Color and Texture Features***,** Journal of Emerging Trends in Computing and Information Sciences, Volume 3, No. 4, April 2012, ISSN 2079-8407, Pages – 488- 496.

[3] Mohd. Danish, Ritika Rawat, Ratika Sharma, *A Survey: Content based image retrieval based on colour, texture, shape and Neuro fuzzy*, International Journal of Engineering Research and Applications (IJERA), ISSN 2248-9622, Volume 3, Issue - 5, Sept – Oct 2013, Pages – 839 – 844.

[4] Schaefer, G., *An introduction to content – based image retrieval,* Digital Information Management (ICDIM), 2013, 8th international conference, ISBN number 978-1-4799-0613-0, 10-12 September 2013.

[5] Yogita Mistry, Dr. D.T.Ingole, *Survey on Content Based Image Retrieval Systems*, International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE), ISSN 2320-9801, Vol. 1, Issue 8, October 2013, Pages – 1827 – 1836.

[6] Madhura C., Dheeraj D., *Feature Extraction for image retrieval using colour spaces and GLCM*, International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN 2278-3075, Volume 3, Issue - 2, July 2013, Pages – 159 -164.

[7] Raman Maini, Dr. Himanshu Aggarwal, *Study and comparison of various image edge detection techniques*, International Journal of Image Processing (IJIP), Volume (3) and Issue 1, Pages – 1 – 12.

[8] Mrs Shweta Singh, Mr Anand Rajavat, *Content based image indexing based on framelet transform and color*, International Journal of Scientific & Engineering Research (IJSER), Volume 4, Issue 9, September-2013, ISSN 2229-5518, Pages – 1490 – 1492.

[9] A. Hasnat, S. Halder, A. Hoque, D. Bhattacharjee, M. Nasipuri, *A Fast FPGA Based Architecture for Measuring the Distance Between Two Color Images Using Manhattan Distance Metric*, International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6472, Volume 4, Issue 3, May – June, 2013, Pages - 1-10.

[10] Mr. Vivek Singh Rathore, Mr. Messala Sudhir Kumar, Mr. Ashwini Verma, *Colour Based Image Segmentation Using L*A*B* Colour Space Based On Genetic Algorithm,* International Journal of Emerging Technology and Advanced Engineering (IJETAE), Pages – 156 – 162.