# Image Retrieval from Databases: an Approach using Region Color and Indexing Technique

**M V Sudhamani**
Asst. prof, Dept. of ISE
Siddaganga Institute of Technology
Tumkur, pin: 572 103,
Visvesvaraya Technological University
Karnataka, India

**Dr. C R Venugopal**
Professor, Dept. of E & C
Sri Jayachamarajendra College of Engineering
Mysore, pin: 570 006
Visvesvaraya Technological University
Karnataka, India

## Summary
The paper focuses on a low-dimensional color-based indexing technique for achieving efficient and effective retrieval performance. We have implemented the technique of retrieval of images from databases based on color feature extracted from each region/regions of an image represented by dominant or representative color descriptor. Each image is segmented using Mean Shift algorithm, a robust clustering technique to extract color feature of a region of interest. Image databases have been designed and the image region features are ingested into them. Indexing data structures are designed to support basic operations on image databases such as search, insertion, and deletion. Search efficiency is considered as the most important aspect because search is normally carried out on-line. The new algorithm cluster-based R*-tree indexing is proposed and compared the efficiency with R* -tree and sequential search. The accuracy of the retrieval system is measured and compared the results. Experimental results show the high performance of the proposed method. The toolbox is developed in JAVA to incorporate all these modules and retrieve images based on query-by-example.

***Key words:***
Image retrieval, Segmentation, Databases, Search efficiency, Indexing.

## 1. Introduction

With the advances in multimedia technologies and the increasing emphasis on multimedia applications, the production of image and video information has resulted in large volumes of images and video clips. This trend is likely to continue, and to cope effectively with the explosion of multimedia information, image-based systems have been proposed to properly organize and manage this information for rapid retrievals [1]. Among low-level visual features color is one of the most dominant and distinguishing visual feature used for content-based image retrieval (CBIR). The representative color descriptor is compact and captures the representative color or colors in a given region.

### 1.1 Previous Work

Current CBIR systems such as IBM's QBIC, allow automatic retrieval based on simple characteristics and distribution of color, shape and texture. But they do not consider structural and spatial relationships and fail to capture meaningful contents of the image in general. Also the object identification is semi-automatic. The Chabot project integrates a relational database with retrieval by color analysis. Textual meta-data along with color histograms form the main features used. VisualSEEK allows query by color and spatial layout of color regions. Text based tools for annotating images and searching is provided. A new image representation which uses the concept of localized coherent regions in color and texture space is presented [15], [16]. Recently, additional systems have been developed at IBM T.J. Watson [2], VIRAGE [3], NEC C&C Research Labs [4], Bell Laboratory [5], Interpix (Yahoo), Excalibur, and Scour.net. In academia, MIT Photobook [6], [7] is one of the earliest. Berkeley Blobworld [8], CMU Informedia [9], University of Illinois MARS [10], University of California at Santa Barbara NeTra [11], the system developed by University of California at San Diego [12], Stanford WBIIS [13], and Stanford SIMPLIcity [14], [15] are some of the recent systems. Some of the popular methods to characterize color information in images are color histograms, color moments and color correlograms. Though all these methods provide good characterization of color, they have the problem of high-dimensionality. This leads to more computational time, inefficient indexing and performance. To overcome these problems, use of SVD, dominant color regions approach, and color clustering have been proposed. In this paper, we focus on region-based retrieval of images obtained through segmentation by clustering.

### 1.2 Region-based retrieval

Region-based approach has recently become a popular research trend. Region-based retrieval systems attempt to overcome the deficiencies of color histogram and color layout search by representing images at the object-level. A

Manuscript received January 5, 2008
Manuscript revised January 20, 2008

region-based retrieval system applies image segmentation to decompose an image into regions, which correspond to objects if the decomposition is ideal. The object-level representation is intended to be close to the perception of the human visual system. Many retrieval systems match images based on individual regions. Such systems include for e.g., the Netra system and the Blobworld system. To query an image, a user is provided with the segmented regions of the image, and is required to select the regions to be matched e.g., color, of the regions to be used for evaluating similarity. Blobworld is a CBIR system that fragments an image into regions (blobs), homogeneous with respect to color and texture, by using an Expectation-Maximization clustering algorithm. The Blobworld index-based query resolution algorithm uses an *R*-tree like structure to index color descriptors of blobs. This uses nearest neighbors query on the index. Netra system used the edge-flow algorithm for image segmentation and colors in a region are indexed using hexagonal lattice structure. Most of the existing work focused on the effectiveness of retrieval mechanisms (i.e., precision and recall). While there is some work done to address the efficiency issue (i.e., speedy retrieval), most of these are compared against the brute force approach of scanning the entire feature representation. Our proposed method uses Mean shift robust clustering algorithm for segmentation of images and interested region/regions are indexed using cluster-based $R^*$-tree. This is compared with $R^*$-tree and sequential scan. This algorithm is described in section 5 and uses range query. The retrieval accuracy is compared with [18], [19]. The remainder of the paper is organized as follows. In Section 2, the proposed architecture is given. Section 3 explains feature extraction process. Section 4 discusses the database design. In section 6, experimental results are provided and discussed. Section 7 concludes the paper and future extension of the present work.

## 2. Architecture

Fig. 1 shows architecture of our proposed Region-based image retrieval system. Two main functionalities are supported: Data insertion and Query processing. The data insertion subsystem is responsible for extracting appropriate features from images and storing them into the image database. This process is performed off-line. The query processing, intern, is organized as follows: the interface allows a user to specify a query by means of a query pattern and to visualize the retrieved similar images. The query-processing module extracts a feature vector from a query pattern and applies a metric as the Euclidean distance to evaluate the similarity between the query image and the database images. Next, it ranks the database images in a decreasing order of similarity to the query image and forwards the most similar images to the interface module. The database images are indexed according to their feature vectors to speed up retrieval and similarity computation. Both the data insertion and the query processing functionalities use the feature vector extraction module.

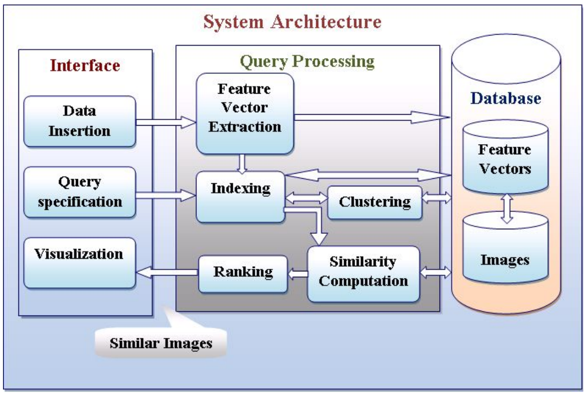



Fig.1 Architecture of a proposed system

## 3. Feature Extraction

### 3.1 Clustering Algorithm

The steps of the algorithm are as follows [17]:

1. Tessellate the space with $k << n$ spheres (windows).

A set of $k$ points $x_1......x_k$ called the sample set is randomly selected from the set of $n$ data points to reduce the computational load. The points in the sample set are imposed with two constraints. The distance between any two neighbor points should not be smaller than the radius of the sphere, and they should not lie in sparsely populated regions. The later constraint is required to avoid low-density clusters. The number of points inside the sphere is less than threshold $T_{d1}$ and then it is a sparsely populated region. The constraints, distance and density automatically determine the size $k$ of the sample set. The spheres centered on the sample set points cover most of the data points.

2. Apply the mean shift procedure to the sample set points in parallel.

The $k$ cluster center candidates in the sample set are defined by the points of convergence of the $k$ mean shift procedures. The computational complexity decreases which is now $O(k \times n)$, and the computation of the mean shift vectors is based on the entire data set. Therefore, with sampling, the quality of the density gradient estimate is not reduced.

3. Reapply the mean shift procedure on perturbed cluster center candidates.

Since a local plateau can prematurely stop the iterations, each cluster center candidate is perturbed by a random vector of small norm and the mean shift procedure is allowed to converge again.

4. Derive the cluster centers $y_1......y_p$ from the cluster center candidates.

A cluster center is defined as any subset of cluster center candidates, which are sufficiently close to each other. The cluster center is the mean of the cluster candidates in the subset. Note that $p \leq k$ .

5. Validate the cluster centers.

The presence of the valley is tested for each pair $(y_i, y_j)$ because a significant valley should occur in the underlying density between any two cluster centers $y_i$ and $y_j$ . The sphere is moved with step equal to radius along the line defined by $(y_i, y_j)$ and the weighted number the data points lying in the sphere is counted at each position, i.e., the density is estimated with Epanechnikov kernel $K_E$ along the line. Whenever the ratio between *min* $[\hat{f}(x_i), \hat{f}(x_j)]$ and the minimum density along the line is larger than a threshold $T_{d2}$, a valley is assumed between $y_i$ and $y_j$ . If no valley was found between $y_i$ and $y_j$, the cluster center of lower density ( $y_i$ or $y_j$) is removed from the set of cluster centers.

6. Clusters delineation.

Here each sample point is associated with a cluster center. To allocate the data points a $k$ - nearest neighbor technique is employed, i.e., each data point belongs to the cluster defined by the majority of its $k$ - nearest sample points.

## 3.2 Color Image Segmentation

The local color feature extraction starts with color image segmentation. The feature extraction subsystem, which uses mean shift segmentation algorithm [17] to segment an image based on color. After segmentation, color feature of the interested region/regions could be extracted. This segmentation algorithm is based on the feature space analysis. In this paradigm the pixels are mapped into $L*u*v*$ color space and clustered, with each cluster delineating a homogeneous region in the image. For color image segmentation, the mean shift is employed in the spatial and range domain. Only two parameters, the resolution in the spatial and range domain will control the segmentation quality. Fig. 2 shows the flow of the procedure.

Here, an image is represented as a 2-dimensional lattice of q dimensional vectors (pixels), where q is 1 for gray level images, and 3 for color images. The gray level, or color information is represented in the range domain while the space of the lattice is known as the spatial domain. After a proper normalization, with $h_s$ and $h_r$ are the global parameters in the spatial and range domains, the location and range vectors can be concatenated to obtain a spatial range domain of dimension $d = q + 2$ . Each data point becomes associated to a point of convergence which represents the local mode of the density in the $d$ - dimensional space. Both spatial and range information is taken into account simultaneously by the process. Example is given in Fig. 3.

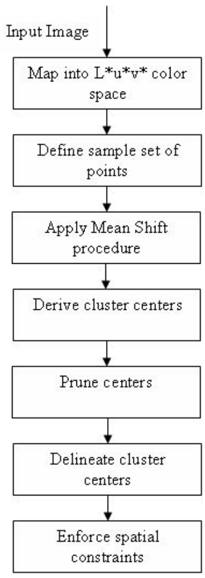



**Fig. 2 Flow of cluster-based image segmentation**

A simple mode detection procedure can be derived by recursive application of the mean shift property. The modes are the local maxima of the density, i.e. $\nabla f(x) = 0$ . They can be found by moving at each iteration the window by the mean shift vector, until the magnitude of the shifts becomes less than a threshold. The procedure is guaranteed to converge.

**(a)** **(b)**

(c) (d)

**Fig. 3 Natural images segmented using Mean shift algorithm. (a) (c) Original. (b)(d) Segmented images.**

## 4. Database Design and Implementation

Image retrieval problem is concerned with retrieving images that are relevant to user's requests from a large collection of images referred to as image database. Image database design is the process of producing a detailed Image Data Model (IDM) of a database. This IDM contains the required logical & physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. A fully attributed IDM contains detailed attributes for each entity. We have designed three databases with population is as shown in table 1.

**Table 1. Population in the Databases**

| Database | # Images | Categories | Regions | # Queries |
|---|---|---|---|---|
| Flag image | 200 | *** | 440 | 13 |
| Natural image | 3185 | 17 | 5059 | 143 |
| Ground Truth | 1085 | 20 | 1200 | 67 |

Each representative color descriptor F is defined to be F= $\{c_i\}$, i=1,., 3, where $c_i$ is 3-D color vector of a region. Each object or region in the database is represented using color descriptor. Given a query image, similarity retrieval involves searching the database for similar color as the input query. Searching for the individual colors can be done very efficiently in a 3-D color space. We consider only fixed range queries in which the range value limits the search range.

The User Interface is designed to provide the options of Insertion, Deletion and Retrieval of the Images from the image database as in figure 11 (d). Fig. 4 shows images in database. Figure 5 and 6 shows sample images inserted in to database.

## 5. Indexing

A unique image ID is assigned for each image in the database. A Unique image region ID identifies each region in the database. The entries in each index node are sorted by region ID. The proposed indexing scheme allows the database to be dynamic, allowing straightforward insertion and deletions of database entries thus avoiding the reconstruction of the entire index structure of the database.

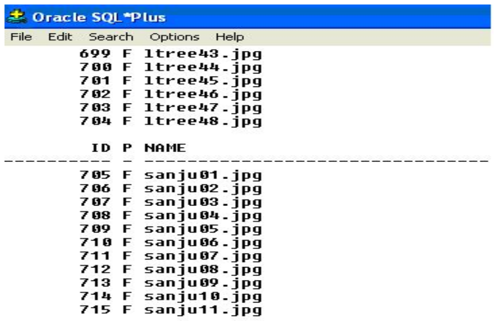



**Fig. 4 Sample view of natural image database**

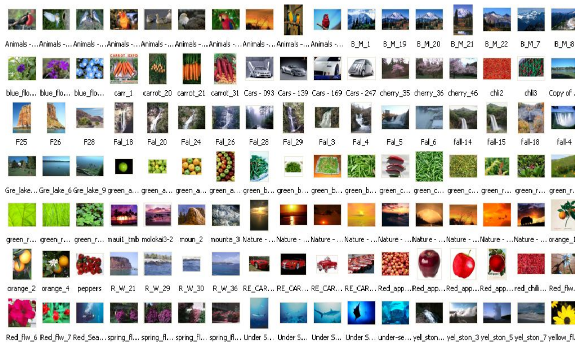


**Fig. 5 Sample view of images in the Natural image database**

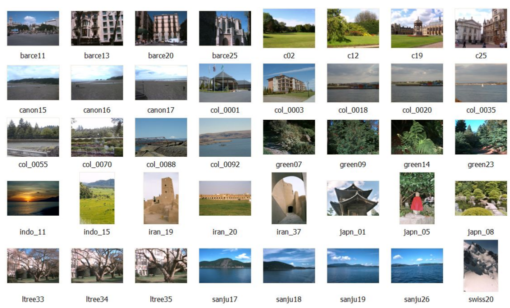


**Fig. 6 Sample view of images in the Ground Truth image database**

To speed-up the evaluation of range queries we describe cluster-based $R^*$-tree indexing method. This is carried out by reducing the number of candidate images, the images to be indexed on which the optimal region-matching problem has to be solved. From each category of images which are stored in the database, n -clusters are constructed for grouping the features, where $n$ is the number of region features chosen for query. The features in the database are mapped to a corresponding cluster based on Euclidean similarity measure. Each representative color of a cluster is indexed using $R^*$-tree [20] rather than indexing all the regions thus reducing the time. We access only the images selected for retrieval as candidates.

The procedure is as follows:

1. Given $n$ the number of query regions, for each query region $q_j$, find the regions belonging to cluster $c_j$, where $j = 1,..,n$.
2. For each region $r_i$ in the image database
   a. Find the feature vector $f_i$, for region $r_i$.
   b. For each query region $q_j$, in the query set
      i. Find query feature vector $f_j$, for $q_j$.
      ii. Find the Euclidean distance between $f_i$ and $f_j$ using: $d_{ij} = \sqrt{\sum_{k=1}^{m} \left(f_{i_k} - f_{j_k}\right)^2}$, where $m$ is the dimension of the feature vector. This score is zero if the regions features are identical, it increases as the match becomes less perfect.
      iii. Measure the similarity between $f_i$ and $f_j$ using $\mu_{ij} = d_{ij} - \tau$, where $\tau$ is the search range limit set by user.
      iv. If $\mu_{ij} \le 0$, then $f_i$ belongs to cluster $c_j$ and go to step 2.

After the completion of the above procedure, we index only representatives of $c_j$, where $j = 1,...,n$ using $R^*$-tree. Once the user selects the query, we apply range search on the tree and the selected regions belonging to image are retrieved as resultant set. Members of resultant set is ranked according to overall score and return the best matches in decreasing order of similarity along with their relative information.

In case of $R^*$-tree, all the regions in the database are being indexed. When we pose a query-by-example, based on the range, selected images are displayed as a resultant set according to ranking of similarity in descending order.

In sequential search, all the regions stored in the database are compared for similarity and intern retrieved for display making it inefficient.

All three methods yield good performance when the accuracy of resultant set is considered as depicted in table 3, table 4, and table 5.

## 6. Experimental Results

The representative color descriptor is tested on a database of 200 color flag images and a database of 3185 natural images belonging to 17 various categories. After segmentation 440 regions from flag images and 5059 regions from natural images are obtained. Among them, 13 and 143 image regions containing a variety of colors and color combinations are chosen from flag and natural image database respectively as queries. Table 1 summarizes the population. Additional testing has been carried out by experimenting with a sizeable database with an established ground truth available http://www.cs.washington.edu/research/imagedatabase/groundtruth/ to check the suitability of our approach. Here 1085 images belonging to 20 various categories are stored in the database. Among them 67 images are chosen as queries.

To determine relevant matches in the database to the query image region a subjective test is carried before evaluation. The time complexity associated with proposed method (cluster-Index) and other methods ($R^*$-tree and Sequential search) are shown in Fig 7(a), 7(b) and 7(c) and listed the average values in the table 2. As we can observe from the graph that efficiency of the proposed method is high.

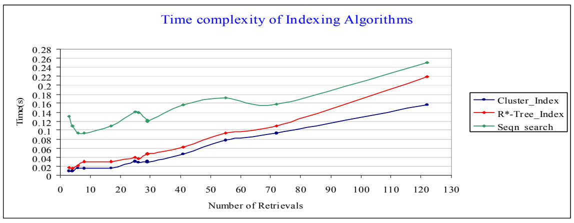
The retrieval accuracy is measured by precision and recall, $\mathrm{Precision}(k) = c_k / k$ and $\mathrm{Recall}(k) = c_k / M$, where $k$ the number of retrievals is, $c_k$ is the number of relevant matches among all the $k$ retrievals, and $M$ is the total number of relevant matches in the database obtained through the subjective testing. The precision and recall values for different queries are listed in table 3, 4, and 5 for flag image, natural image, and Ground Truth image databases respectively.

The average precision and recall graphs for image queries are plotted in Fig. 8, Fig. 9 and Fig.10 for all three methods. In general, a more effective system shows a higher precision for all values of recall. That can be observed from plotted graphs. It can be seen from the table 3, 4, and 5 and Figures 8 to 10 that the proposed method achieves better results in terms of the retrieval accuracy. Fig. 11 shows the snapshots of region-based image retrieval and data insertion and display operations from the database. The retrievals in the examples show good match of the query images.

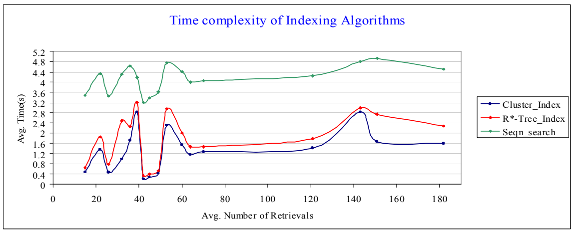
The results obtained are compared with [18]. Here Figure 18 shows an example output for fruits, flowers and vegetables database (140 images) and for flag database

(200 images) (totally about 900 images) using only color feature. The average values of precision and recall are 0.708851 and 0.282746 and 0.759438 and 0.31985 for flowers and flag databases respectively as given in table 2 and table 3 in [18]. These values are compared with average (average values of all queries in the database) values of precision and recall of our method as given in table 6. It is observed from the table values that the proposed method achieves better retrieval accuracy.

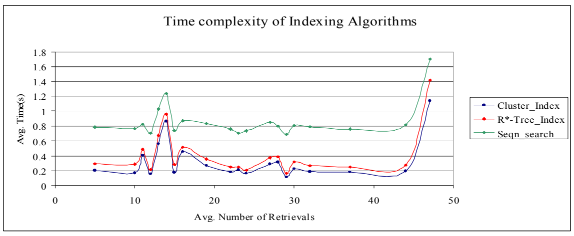
Similarly, the results are also compared with [19] listed in table III, where color feature is used to retrieve natural images collected from COREL database. The values of precision and recall can be compared both with Natural image database and Ground Truth image database values because both contain natural images. The average values of precision and recall are 0.647 and 0.474. These values can be compared with the values of the proposed method given in the table 6. It is observed that the proposed method over scores results of [19].




(a)




(b)




(c)

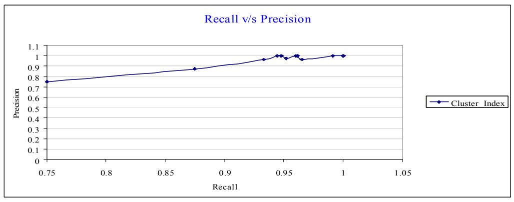
**Fig. 7 Time in seconds versus number of retrievals (a) Flag image database (b) Natural image database (c) Ground Truth image database.**

**Table 2 Average time (in seconds) for all queries in the database**

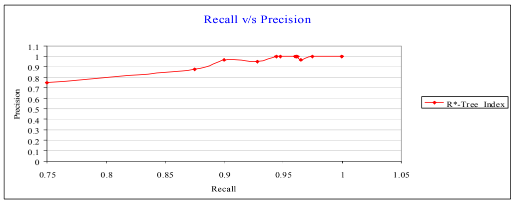
| Database | Cluster-based R*-Tree Index | R*-Tree Index | Sequential search | Average result set size |
|---|---|---|---|---|
| Flag Image | 0.0432 | 0.0593 | 0.138 | 33.6 |
| Natural Image | 1.327 | 1.779 | 4.138 | 66.35 |
| Ground Truth | 0.321 | 0.41 | 0.859 | 22.9 |

**Table 6 Comparison of Experimental Results**

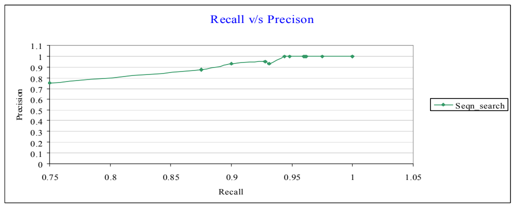
| | Our proposed method | | | | | | Reference [197] | | Reference [193] | |
|---|---|---|---|---|---|---|---|---|---|---|
| Database | Cluster R*-Tree Index | | R*-Tree Index | | Sequential search | | K.K.Biswas et.al., 2004 | | Deng et. al., 2001 | |
| | Precision | Recall | Precision | Recall | Precision | Recall | Precision | Recall | Precision | Recall |
| Flag images | 0.964 | 0.942 | 0.962 | 0.936 | 0.957 | 0.933 | 0.759438 | 0.31985 | *** | *** |
| Natural images | 0.969 | 0.877 | 0.956 | 0.869 | 0.953 | 0.867 | 0.708851 | 0.282746 | 0.647 | 0.474 |
| Ground Truth images | 0.969 | 0.981 | 0.966 | 0.965 | 0.963 | 0.953 | *** | *** | 0.647 | 0.474 |




**(a) Recall versus Precision for each query (Cluster Index)**

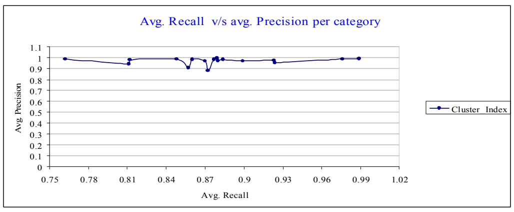



**(b) Recall versus Precision for each query ($R^*$-Tree)**

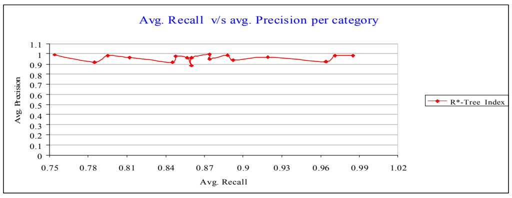



**(c) Recall versus Precision for each query (Sequential search)**

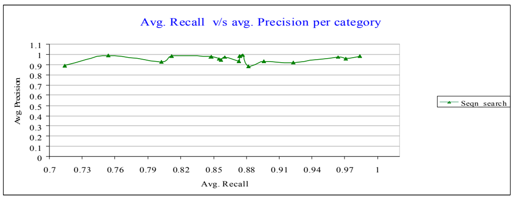
**Fig. 8 Flag image Database**




**(a) Average Recall versus Average Precision for each category (cluster Index)**

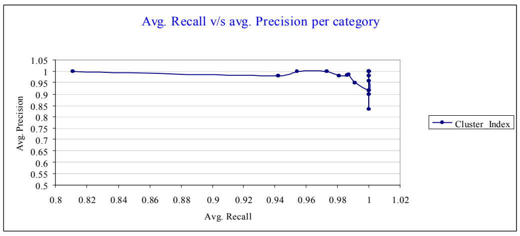



**(b) Average Recall versus Average Precision for each category (R*-Tree)**

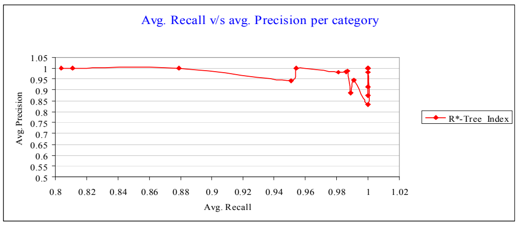



**(c) Average Recall versus Average Precision for each category (Sequential search)**

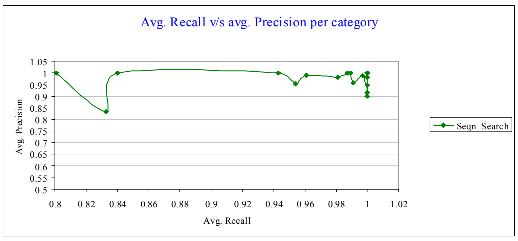
**Fig. 9 Natural image Database**




**(a) Average Recall versus Average Precision for each category (cluster Index)**

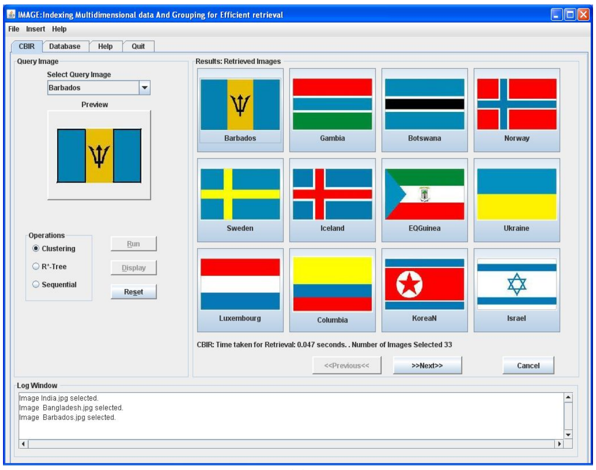



**(b) Average Recall versus Average Precision for each category (R*-Tree)**

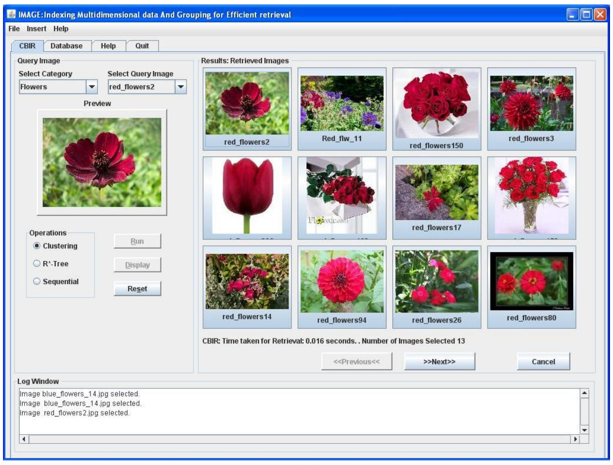



**(c) Avg. Recall versus Avg. Precision for each category (Sequential search)**

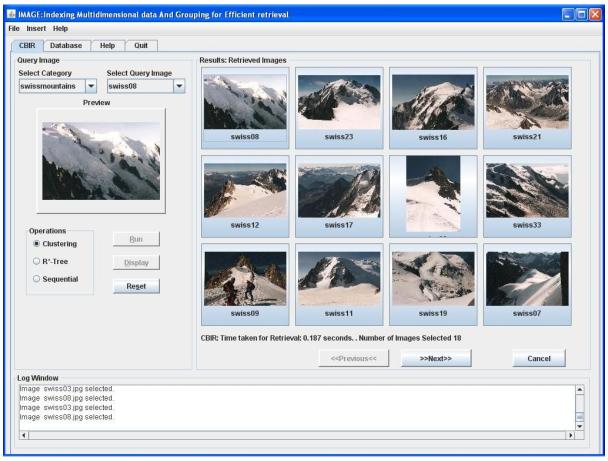
**Fig. 10 Ground Truth image Database**




**(a)**




**(b)**




**(c)**


(d)

**Fig.11 Example region-based image search using the representative color descriptor. (a) The query is the Sky Blue of Barbados Flag. (b) The query is the red flower of flowers category. (c) Mountain with snow cap of Swiss mountains category. (d) Database insertion and retrieve operations.**

## 7. Conclusions

A novel Cluster-based $R^*$-Tree indexing method for efficient retrieval of the image is discussed. The technique is tested on a natural image database, flag image database and ground truth database as applications. The mean shift algorithm is used for segmentation to obtain regions of interest to improve the effectiveness of the retrieval system. The region color features as a result of segmentation are ingested into the database. Experimental results depict the proposed method gives better performance over the $R^*$-Tree and sequential search methods in terms of efficiency. Accuracy of retrieval is compared and shown to be high. A query-by-example based toolbox called IMAGE is implemented for database manipulation and retrieval in JAVA. As further extension of the proposed work, the system should be tested on a more populated database.

## References

[1] Kian-lee tan, Beng chin ooi and cchia yeow yee, *Multimedia Tools and Applications, 14,* 2001,pp .55–78.

[2] J. R. Smith, C. S. Li, Image classification and querying using composite region templates, *Journal of Computer Vision and Image Understanding, vol. 23,* 2001.

[3] A. Gupta, R. Jain, *Visual information retrieval, Communications of the ACM, vol. 40,* no. 5, May 1997, pp. 70-79.

[4] S. Mukherjea, K. Hirata, Y. Hara, *AMORE: a World Wide Web image retrieval engine, World Wide Web, vol. 2,* no. 3, Baltzer, 1999, pp. 115-32.

[5] Natsev, R. Rastogi, K. Shim, WALRUS: A similarity retrieval algorithm for image databases, *Proc. ACM SIGMOD,* Philadelphia, PA, 1999.

[6] Pentland, R. W. Picard, S. Sclaro, Photobook: tools for content-based manipulation of image databases, *SPIE, vol. 2185*, San Jose, February 7-8, 1994, pp. 34-47.

[7] R. W. Picard, T. Kabir, Finding similar patterns in large image databases, *Proc. IEEE ICASSP, Minneapolis, vol. V,* 1993, pp. 161-64.

[8] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, J. Malik, Blobworld: a system for region-based image indexing and retrieval, *Proc. Int. Conf. on Visual Information Systems,* D. P. Huijsmans, A. W.M. Smeulders (eds.), Springer, Amsterdam, The Netherlands, June 2-4, 1999.

[9] S. Stevens, M. Christel, H. Wactlar, Informedia: Improving access to digital video, *Interactions, vol. 1,* no. 4, 1994, pp. 67-71.

[10] S. Mehrotra, Y. Rui, M. Ortega-Binderberger, T.S. Huang, Supporting content-based queries over images in MARS, *Proc. IEEE International Conference on Multimedia Computing and Systems,* Ottawa, Ont., Canada 3-6 June 1997, pp. 632-3.

[11] W. Y. Ma, B. Manjunath, NeTra: A toolbox for navigating large image databases, *Proc. IEEE Int. Conf. Image Processing,* 1997, pp. 56-71.

[12] R. Jain, S. N. J. Murthy, P. L.-J. Chen, S. Chatterjee, Similarity measures for image databases, *SPIE, vol. 2420,* San Jose, CA, Feb. 9-10, 1995, pp. 58-65.

[13] J. Z. Wang, G. Wiederhold, O. Firschein, X. W. Sha, Content-based image indexing and searching using Daubechies' wavelets, *International Journal of Digital Libraries, vol. 1,* no. 4, 1998, pp. 311-328.

[14] J. Z. Wang, *Integrated Region-Based Image Retrieval* (Kluwer Academic Publishers, 2001).

[15] Zaher aghbari, Akifumi makinouchi, Semantic Approach to Image Database Classification and Retrieval, *NII Journal,* No. 7, 2003.

[16] Shu-Ching, Chen Stuart H. Rubin, Mei-Ling, A Dynamic User Concept Pattern Learning, Framework for Content-Based Image Retrieval, *IEEE transactions on systems, man, and cybernetics—part c: applications and reviews, vol. 36,* no. 6, Nov. 2006.

[17] M. V. Sudhamani, C.R. Venugopal, Segmentation of Images through clustering to Extract Color Features: An application for image Retrieval, *International Journal of Computer Science, vol. 2,* No.1, 2007, pp.54-61.

[18] B. G. Prasad, K.K. Biswas, and S.K. Gupta, Region-based image retrieval using integrated color , shape, and location Index, *International journal of computer vision and Image understanding,* vol. 94, 2004, pp. 193-233.

[19] Yining Deng, B. S. Manjunath, Charles Kenney, Michael S. Moore, and Hyundoo Shin, *An Efficient Color Representation for Image Retrieval,* IEEE Transactions on Image Processing, vol. 10, no. 1, January 2001, pp. 140-147.

[20] N. Beckmann, Hans-Peter Kriegel, Ralf Schneider, Bernhard Seeger, The $R^*$-Tree: An Efficient and Robust Access Method for Points and Rectangles, ACM SIGMOD International Conference on Management of Data, Atlantic City, May 1990.

**Table 3. Precision and Recall values for Flag Database Image queries**

| Query # | | | Cluster_Index (new) | | R* -Tree Index | | Seuential search | | # Retrievals |
|---|---|---|---|---|---|---|---|---|---|
| | Flag | Region color | Precision | Recall | Precision | Recall | Precision | Recall | |
| 1 | Bahrain | Red | 1 | 0.991 | 1 | 0.975 | 1 | 0.975 | 122 |
| 2 | India | Saffron | 1 | 1 | 1 | 1 | 1 | 1 | 6 |
| 3 | Australia | N Blue | 1 | 0.961 | 1 | 0.961 | 1 | 0.961 | 25 |
| 4 | Bangladesh | Green | 1 | 0.948 | 1 | 0.948 | 1 | 0.948 | 55 |
| 5 | Barbados | L Blue | 0.965 | 0.933 | 0.964 | 0.9 | 0.931 | 0.9 | 29 |
| 6 | Cyprus | White | 1 | 0.96 | 1 | 0.96 | 1 | 0.96 | 72 |
| 7 | Columbia | Yellow | 0.975 | 0.952 | 0.951 | 0.928 | 0.951 | 0.928 | 41 |
| 8 | Kenya | Black | 1 | 0.962 | 1 | 0.962 | 1 | 0.962 | 26 |
| 9 | Lavita | Brown | 0.75 | 0.75 | 0.75 | 0.75 | 0.75 | 0.75 | 4 |
| 10 | Gabon | L Green | 1 | 0.944 | 1 | 0.944 | 1 | 0.944 | 17 |
| 11 | Capvarde | Dark Blue | 0.965 | 0.965 | 0.965 | 0.965 | 0.931 | 0.931 | 29 |
| 12 | Georgia | Pink Brown | 1 | 1 | 1 | 1 | 1 | 1 | 3 |
| 13 | Argentina | Sky Blue | 0.875 | 0.875 | 0.875 | 0.875 | 0.875 | 0.875 | 8 |
| Average | | | **0.964** | **0.942** | **0.962** | **0.936** | **0.957** | **0.933** | **33.62** |

**Table 4. Precision and Recall values for Natural Database Image queries**

| Category | Cluster_Index (new) | | R*-Tree Index | | Sequential search | | # Retrievals |
|---|---|---|---|---|---|---|---|
| | Precision | Recall | Precision | Recall | Precision | Recall | |
| 1 | 0.987 | 0.884 | 0.956 | 0.874 | 0.983 | 0.874 | 42 |
| 2 | 0.99 | 0.762 | 0.99 | 0.754 | 0.99 | 0.754 | 45 |
| 3 | 0.995 | 0.879 | 0.995 | 0.845 | 0.993 | 0.877 | 143 |
| 4 | 0.945 | 0.811 | 0.918 | 0.795 | 0.925 | 0.802 | 26 |
| 5 | 0.99 | 0.848 | 0.962 | 0.848 | 0.979 | 0.848 | 49 |
| 6 | 0.981 | 0.812 | 0.982 | 0.812 | 0.984 | 0.812 | 15 |
| 7 | 0.988 | 0.976 | 0.981 | 0.964 | 0.975 | 0.964 | 53 |
| 8 | 0.973 | 0.88 | 0.947 | 0.857 | 0.947 | 0.857 | 32 |
| 9 | 0.986 | 0.86 | 0.973 | 0.86 | 0.973 | 0.86 | 22 |
| 10 | 0.882 | 0.872 | 0.882 | 0.892 | 0.882 | 0.882 | 36 |
| 11 | 0.987 | 0.877 | 0.959 | 0.971 | 0.959 | 0.971 | 39 |
| 12 | 0.91 | 0.857 | 0.916 | 0.785 | 0.89 | 0.714 | 121 |
| 13 | 0.972 | 0.87 | 0.96 | 0.86 | 0.955 | 0.855 | 151 |
| 14 | 0.979 | 0.923 | 0.967 | 0.888 | 0.933 | 0.8968 | 64 |
| 15 | 0.971 | 0.899 | 0.938 | 0.8738 | 0.9373 | 0.873 | 70 |
| 16 | 0.993 | 0.989 | 0.983 | 0.985 | 0.982 | 0.984 | 182 |
| 17 | 0.956 | 0.9245 | 0.922 | 0.9188 | 0.9203 | 0.9236 | 35 |
| Average | **0.969** | **0.877** | **0.956** | **0.869** | **0.953** | **0.867** | **66.35** |

**Table 5. Precision and Recall values for Ground Truth Database Image queries**

| Category | Cluster_Index (new) | | R* -Tree Index | | Sequential search | | # Retrievals |
|---|---|---|---|---|---|---|---|
| | Precision | Recall | Precision | Recall | Precision | Recall | |
| 1 | 1 | 1 | 1 | 1 | 1 | 0.989 | 47 |
| 2 | 0.9 | 1 | 0.875 | 1 | 0.9 | 1 | 13 |
| 3 | 0.958 | 1 | 0.887 | 0.989 | 0.916 | 1 | 14 |
| 4 | 0.983 | 0.986 | 0.983 | 0.986 | 0.99 | 0.961 | 30 |
| 5 | 0.98 | 0.942 | 1 | 0.879 | 1 | 0.84 | 27 |
| 6 | 1 | 0.811 | 1 | 0.804 | 1 | 0.801 | 11 |
| 7 | 0.916 | 1 | 0.914 | 1 | 0.947 | 1 | 19 |
| 8 | 1 | 1 | 1 | 1 | 1 | 1 | 28 |
| 9 | 1 | 1 | 1 | 1 | 1 | 1 | 16 |
| 10 | 1 | 0.973 | 1 | 0.811 | 1 | 0.801 | 37 |
| 11 | 1 | 1 | 1 | 1 | 1 | 1 | 12 |
| 12 | 1 | 1 | 1 | 1 | 1 | 0.987 | 32 |
| 13 | 0.981 | 0.981 | 0.981 | 0.981 | 0.981 | 0.981 | 22 |
| 14 | 0.916 | 1 | 0.943 | 0.951 | 1 | 0.943 | 5 |
| 15 | 1 | 0.954 | 1 | 0.954 | 0.954 | 0.954 | 23 |
| 16 | 0.987 | 0.987 | 0.987 | 0.987 | 0.987 | 0.997 | 24 |
| 17 | 0.833 | 1 | 0.833 | 1 | 0.833 | 0.833 | 10 |
| 18 | 0.981 | 1 | 0.982 | 1 | 0.981 | 1 | 15 |
| 19 | 1 | 1 | 1 | 1 | 1 | 1 | 29 |
| 20 | 0.951 | 0.991 | 0.944 | 0.991 | 0.958 | 0.991 | 44 |
| **Average** | **0.969** | **0.981** | **0.966** | **0.965** | **0.963** | **0.953** | **22.9** |

**M V Sudhamani** is a doctoral student at Visvesvaraya Technological University, Karnataka, INDIA. She has received her M. Tech degree in Computer Science & Engg. from Mangalore University and B.E degree in Computer Science & Engg. from Bangalore University. She is currently working as Assistant Professor in Information Science & Engineering Department at Siddaganga Institute of Technology, Tumkur. Her main research interests include Content- based image retrieval, Image processing, Multimedia computing and computer vision.

**Dr. C R Venugopal** currently working as Professor in Electronics & Communication Department, S.J. College of Engineering, Mysore. He has obtained his M.E Degree from Indian Institute of Science, Bangalore. He has received his PhD from Indian Institute of Technology, Bombay, INDIA. His research interests include Multimedia Computing and Communication, Information Retrieval, Computer vision, Pattern Recognition and Security in Digital Storage Area Network. His other activities are IEEE Staff Advisor, IETE Journal Reviewer, and IEEE Software Journal Reviewer