

ADVANCED WEB IMAGE RETRIEVAL USING CLUSTERING ALGORITHMS

Umesh K K¹ and Suresha²

Research Scholar, Department of Studies in Computer Science, University of Mysore
Mysore, India

umeshkatte@gmail.com

²Research Supervisor, Department of Studies in Computer Science, University of Mysore
Mysore, India

suresha@yahoo.co.in

ABSTRACT

In this paper we propose a novel methodology for Web Image retrieval system that takes an image as the input query and retrieves images based on image content. Content Based Image Retrieval is an approach for retrieving semantically-relevant images from an image store based on algorithmically-derived image features. We propose an algorithm to represent images using divisive and partitioned based clustering approaches. The HSV color component and Haar wavelet transform has been used to extract image features. These features are taken to segment an image to obtain objects. For segmenting an image, modified k-means clustering algorithm was used to group similar pixel together into K groups with cluster centers. To modify K-means, a divisive based clustering algorithm has been proposed to determine the number of cluster and get back with number of cluster to k-means to obtain significant object groups. In addition, the similarity distance measure using threshold value and object uniqueness to quantify the results was also measured.

KEYWORDS

Color Feature Extraction, Pixel Segmentation, Modified k-means Clustering, Object Clustering and Object Uniqueness.

1. INTRODUCTION

Off late due to rapid exponential growth of image data and its related technologies, many users get motivated to browse, search and download images from an image store. Thus it is very essential to develop innovative tools for managing and searching the images. The existing search engines provide similar kind of information to the end user. This is good for those who are looking the text related documents which could be retrieved from the web by sending query in the form of set of keywords but is not good for the intuition of web image browsers in order to retrieve similar images. Also it is not possible to retrieve based on the contents of the image.

We have many search engines on the Web to retrieve images and they provide an interface to receive the query in the form of set of keywords [1], [2], [3]. Most of retrieved images are similar but are not relevant to the user query in most of the cases. Thus it is mandatory to process the content of an image to represent with some meaning. We are motivated to utilize the technique of Content Based Image Retrieval to implement our proposed model to retrieve Web images.

In recent years, many researchers have been involved in the area of Content-based image retrieval (CBIR) from unannotated image databases [4]. Now a days, enormously many people use a Digital image and video libraries as a visual information. The technologies needed for retrieving, managing and browsing this growing amount of information are still open issues; however, quite still there is an open challenge for the research community. Many projects have been started in recent years to research and develop efficient systems for content- based image retrieval. The best-known CBIR system is probably Query by Image Content (QBIC) [5] developed at the IBM Almaden Research Center. Other noteworthy systems include MIT's Photobook [6] and its more recent version, FourEyes [7], the search engine family of VisualSEEk, WebSEEk, and MetaSEEk [8],[9],[10], which all are developed at Columbia University. The Virage [11] is a commercial content-based search engine developed at Virage Technologies.

Our proposed model uses a cluster based approach to compute the centroids for every image. This is used for image representation. The main advantage of this algorithm is very easy and takes less time to preserve the cluster of pixels. Basically K-Means is unsupervised and iterative clustering algorithm, which finds the optimal set of cluster. The retrieval method is based on the similarity measure is discussed in detail in the section 2.4.

The organization of the paper is as follows: In section 2, the Methodology of Extracting Image Features, Pixel Segmentation, Object Clustering, Similarity Distance Computation and Implementation are discussed. In Section 3, we discussed the conclusion of the paper.

2. METHODOLOGY

The block diagram of cluster based approach image retrieval system is shown in figure 1. It consists of two phases:

Off-line and On-line.

In Off-line, feature extraction model extracts color and texture features. These features are later input to the clustering module to find the centroids of every image, and then stored in the centroids' database.

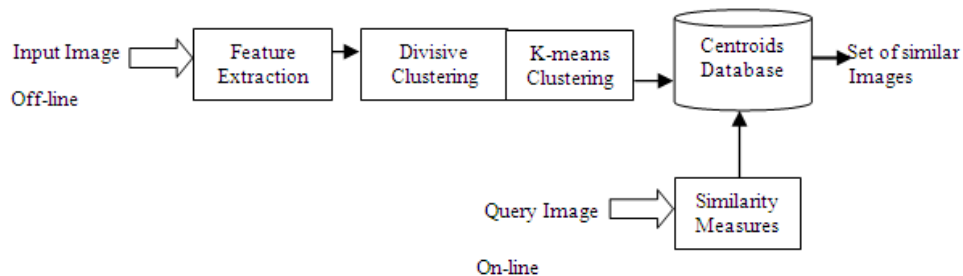


Figure1: The Block Diagram of Image Retrieval system.

In on-line, Image searching is actually processed in the retrieval phase. It accepts a query image and compares with image features using threshold and object uniqueness to measure the similarity to retrieve the images from the image database.

In brief, in this paper we have discussed about three important task of the proposed system.

1. Image pre-processing module, it helps to obtain image features. These features are used to segment to get objects/regions by grouping pixels with similar descriptions.

2. Clustered the objects to tune up the speed of the retrieval system.
3. Employed similarity distance measure to determine the distance between the image query and image features database.

2.1. Feature Extraction

In this section, we discuss how to extract the image features. The features that we can extract from the image are color and textures. According to Li and Wang [12] and Zhang [13], image consists of large number of pixels. The values of pixels are redundant; it can be reduced in the pre-processing stage of an image. In the pre-processing stage, to achieve good result of the image, the image is portioned into 4 by 4 blocks; each block is replaced by the average of 4 by 4 pixel values to compromise between texture granularity, segmentation coarseness, and computation time. The HSV color space is widely used for extracting color features due to its ability to transform from RGB to HSV and vice versa. Since HSV color space has an advantage to produce a collection of colors that is also compact and complete. These features are denoted as $\{f_1, f_2, \text{ and } f_3\}$.

To obtain other three texture features, we applied Haar wavelet transform to the L component of an image. The Haar wavelet is discontinuous and resembles a step function. It represents the energy in high frequency bands of the Haar wavelet transform [14]. After one-level wavelet transform, a 4 by 4 block is decomposed into four frequency bands, each band containing a 2 by 2 matrix of coefficients, suppose the coefficients in the HL band are $\{c_{k+i,ck,l+1,ck+1,l,ck+1,l+1}\}$. Then the feature of the block in HL band is computed using equation 2.1:

$$r = \left[\frac{1}{4} \sum_{i=0}^1 \sum_{j=0}^1 c_{k+i,l+j}^2 \right]^{\frac{1}{2}} \quad 2.1$$

The other two features are computed similarly in the LH and HH bands. These three features of the block are denoted as $\{f_4, f_5, f_6\}$.

Let, Image I represented as $I = \{R_1, R_2, \dots, R_n, \quad x_i \in I\}$ be a region based feature vector set. These regions feature have extracted by using K-means algorithm. To find the signature of weight vectors, that is $c = \{c_1, c_2, \dots, c_n\}$ as representatives of the feature vector of $I = \{f_1, f_2, f_3, f_4, f_5, \text{ and } f_6\}$.

2.2. Pixel Segmentation

In section 2.1, we discuss the procedure to get the required features. The six features which are obtained are taking into consideration for segment an image into objects. For segmenting an image, we modified K-Means clustering algorithm to group similar pixel together into k groups with cluster centers. This is quite efficient and minimizes the sum square error is defined in the equation 2.2.

$$D(k) = \sum_{i=1}^l m_i \min_{1 \leq j \leq k} (x_i - \bar{x}_j)^2 \quad 2.2$$

The advantage of K-Means algorithm is simple and quite efficient. It works well when clusters are not well separated from each other [15]. This could be happen in web images. But the disadvantage of K-means is expected to receive the value of k from the user; each value of k is represented by their centroids. To certain extent for selecting the value of k requires domain knowledge. If the value of k is selected randomly, it gives different distinct total Sum Squared Errors. Thus it is not a good idea to choose the value of k randomly to make a good cluster. This

is the issue mainly we focused in this paper to choose the value of k of K-Means dynamically to determined the number of k values instead of randomly chosen.

This is the reason we used divisive based algorithm to obtain initial cluster centers. In this approach, we first merge pair of the closest pixels to form cluster. A number of clusters are generated. These clusters are then merged to generate a new bigger cluster and finally form a single hierarchical cluster that covers the whole image.

Whenever a new cluster is created we need to calculate the distance between the new cluster and remaining clusters, we use the average linkage method. The average linkage method says let c_n be a new cluster produced by merging of clusters c_i and c_j . Let c_k be a remaining cluster. The distances between the new cluster and the remaining clusters are computed using equation 2.3.

$$\text{Dist}(c_n, c_k) = \frac{|c_i|}{|c_i| + |c_j|} \text{Dist}(c_n, c_k) + \frac{|c_j|}{|c_i| + |c_j|} \text{Dist}(c_j, c_k) \quad 2.3$$

We use person correlation coefficient function to find distance between each pixel to obtain better cluster than Euclidean distance during image segmentation. Let x_i be the features of pixel1 and y_i be pixel2, where $i=1...6$. To find the distance between the pixel x and y computation is done using the person correlation coefficient. From this we get number of features per pixel. Since the number of features per pixel is 6. We set different weight between color (W_c) and texture (W_t) to get better clustering result.

2.3. Object Clustering and Similarity Distance Computation

In this section, we discuss about object clustering and similarity distance computation. The scope of this discussion is to increase the speed of image retrieval system. We cluster similar objects to compare the query image to all images in the database. This takes more time and it consumes more memory, when the database is large. To reduce the computational cost of image retrieval system, we perform hierarchical clustering to the entire objects. Each object is identified by six features. These 6 features are the average features of all the member pixels.

The query image which we input also goes through the same image segmentation algorithm to obtain objects. These objects are compared to the cluster centers in the database and similarity is calculated using L2 distance. The objects in the database that has a minimum distance will be returned to perform global image distance computation between query image and database image. The objects in the database that has a minimum distance will be returned to perform global image distance computation between query image and database image.

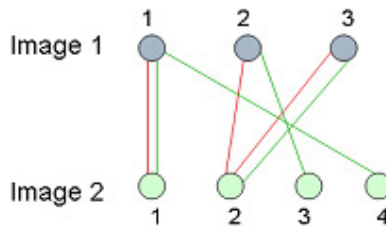


Figure2.2: Overall similarity distance between image query and image database.

Fig 2.2 shows that, image1 is the query image and image2 is the returned image from the image database. To determine the overall distance between the query image and the returned image, we should match all the objects from the two images. From these two images, we measure the

distance between the first object from image1 (O_{11}) and the first object from image2 (O_{22}), is defined as:

$$O_{11,21} = \sum_{i=1}^5 (f_{11i} - f_{21i})^2 \quad 2.4$$

where f_i is the color and texture from each object.

The similarity distance computation between image1 and image2 can be defined as:

$$W_p = \alpha_p \beta_p \gamma_p \quad 2.5$$

Where $\alpha_p = 1$ when the two region being match is the closest distance

$\alpha_p = 0$ when the two region being matched is not the closest distance

$\beta_p = (\text{\#of pixels for object } p) / (\text{\#of pixel in the whole image})$

$\gamma_p = \text{Object uniqueness of the object based on the appearance of the object}$

We determine the uniqueness of the object that is based on the visual appearance of the object in the object cluster. When an object appears frequently in the database, it can be consider less unique and vice versa. To get the value of γ_p , first we carry out hierarchical clustering to group similar objects in the same group. Next, we determine the uniqueness of object p (γ_p) based on the total number of objects that belong to the cluster compared to the total number of objects in the database. The similarity distance between $D(Im1, Im2)$ and $D(Im2, Im1)$ is not symmetric, therefore in order to make it symmetric, we take the average between them. The overall distance is defined as:

$$\text{Overall } D(Im1, Im2) = \frac{D(Im1, Im2) + D(Im2, Im1)}{2} \quad 2.6$$

By using this equation, it is straightforward to determine $D(ImX, ImY)$ for every images in the database. This is most well balanced technique in similarity measure between regional and global matching. This cluster of object groups is necessary for two reasons: for faster image retrieval and for determining the value of object uniqueness is discussed in the subsection 2.4.2

2.4. Implementation

To retrieve relevant images, user sends a query in the form of image to the image retrieval system to find out the set of similar images. Whenever it receives the query, it forwards to feature extraction module. In feature extraction module the computation of the signature of the image is done. The signature of the image is compared with features vectors stored in the database using the similarity measure. The similar images are retrieved and then displayed to the corresponding thumbnail images at the user place. These thumbnail images are available in the database for the user perspective point of view in the form of JPEG format.

2.4.1. Performance

Currently, we have conducted an experiment with two image databases of different sizes. The first database contains 1,000 images with size of 384 X 256 resolution. The second database contains 10,000 low resolution web-crawled miscellaneous image database with the size of 128 X 85 resolution. Most of the images are color photographs in JPEG format and used in WBIIS downloaded from the image collection residing at <http://wang.ist.psu.edu/docs/related>.

shtml. There are 10 categories, around 100 images per category in the first database and 100 categories, around 100 images per categories in the second database.

We used a Pentium(R) 4 CPU 3.00 GHz PC using the Windows-XP operating system. It takes approximately 3 hours to compute the feature vectors. On average, 0.003 seconds is needed to compute the features of all blocks of an image. The feature clustering process is performed by only once for our database. The clustering procedure has been discussed in detail in section 2.3.

We have conducted an experiment to measure the recall and precision of the system based on the images which are presented in the database as a query image. Similarly, images for which are not present in the image database as a query image. We obtain better precision for the given query image shown in figure 2.3, 2.4, 2.5 and 2.6. Due to the limitations of space, we will be able to show only two rows of images with the top 10 matches of sample query.

2.4.2. Experiments

In pre-processing stage, we used MATLAB image processing tools to extract the pixel features on each image.

2.4.2.1. Experiments #1-Similarity Distance Measure.

We obtained objects during pixel clustering. To find a suitable number of objects per image, we need to set the similarity distance measure during divisive clustering. To do this, we randomly selected 10 images from each image class which corresponds to 100 images from the database. We ran this experiment for one time. From the experiment we conducted for each image, we set the similarity measure at 0.6, 0.7 and 0.8. The number of clusters based on these measures was recorded and the average number of clusters for each similarity measure was computed is shown in Table 2.1.

Table 2.1 Correlation between similarity measures and average number of objects

similarity measure	Average number of objects
0.6	3.3
0.7	6.4
0.8	9.2

We concluded to set a similarity measure equal to 0.7 because at similarity measure 0.7 the average number of clusters for each image is 6.3 which is a good number of objects. We argue that 6.3 is a good number of objects for this particular database because when we randomly select 100 images and visually counted the number of objects per image, the average is 6.3. To see the performance of experiment #1 we randomly select 2 images from different class, namely bus and dinosaurs. Each query returns the top 10 images from database. The two query retrievals are shown in Figure 2.2 and 2.3

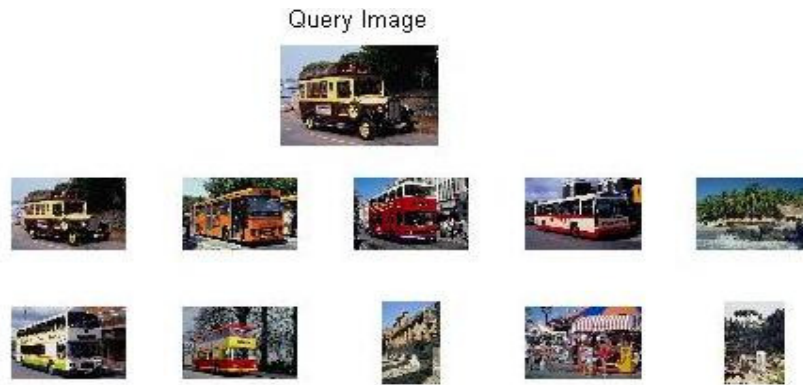


Figure 2.2: Best top 10 matches of a sample bus image query using similarity distance.

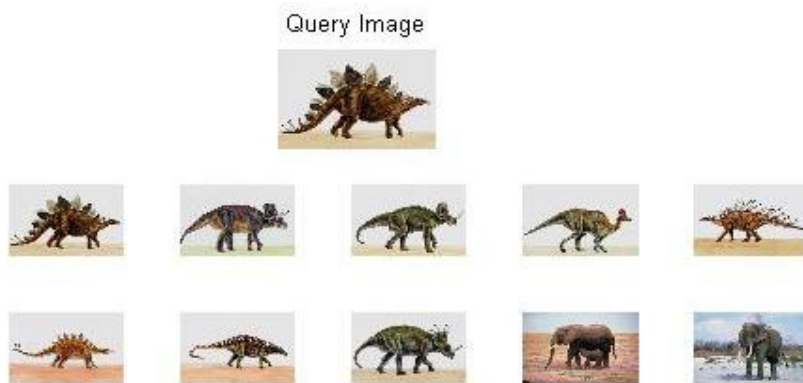


Figure 2.3: Best top 9 matches of a sample dinosaur's image query using similarity distance.

2.4.2.2. Experiments #2 – Object Uniqueness

The idea of object uniqueness is to give more weight to objects that are more distinctive compared to other objects that are more common. To do this, we clustered all objects in database. Objects that belong to larger clusters will get less uniqueness weight compare to objects that belong to smaller clusters.

The selection of the number of clusters in object clustering affects the result of accuracy during image retrieval. The parameter of object uniqueness is obtained from the object clustering. Since object uniqueness is one of the variables to compute similarity distance computation, therefore, to select the number of cluster that results in the best performance. During the conduction of experiment we choose four different number of cluster during object clustering: 1, 10, 25, and 35. All 100 images from each image class are selected for query, and each query returns the result of top 10 from the database. For each query we examine the precision of the returned images on their relevance of semantic meaning. Since similar semantic meaning belong to the same class, and each class has the same number of images, therefore it is straightforward to calculate the precision and recall. The precision and recall using different numbers of object clusters are then compared is shown in table 2.1.

Table 2.1: Average precision/recall comparison of top 1, 10, 25, and 35 closest distances.

Image Id	object cluster(1)	object cluster(10)	object cluster(25)	object cluster(35)
1	0.22	0.27	0.31	0.3
2	0.28	0.31	0.35	0.34
3	0.29	0.35	0.42	0.423
4	0.75	0.61	0.76	0.75
5	0.89	0.96	1.02	0.97
6	0.6	0.59	0.78	0.76
7	0.31	0.28	0.32	0.32
8	0.41	0.4	0.42	0.41
9	0.34	0.31	0.36	0.35
10	0.41	0.39	0.44	0.42

From Figure 2.4 we examined that the precision and recall is higher in almost all image class when the number of object cluster is set to 25. The outcome of 35 object clusters is better compared with 1 object cluster, while the result for 10 object cluster is worse compare to the rest of the cluster number. From this result we conclude that the parameter of object uniqueness in similarity distance computation increase the performance of image retrieval. However, to get the best result, we need to find the accurate number of cluster during object clustering.

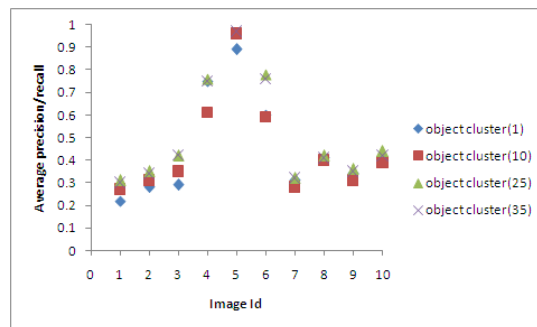


Figure 2.4 Correlation between similarity measures and average number of objects

To see the performance of Experiment #2, we randomly select 2 images from different class, namely bus and dinosaurs. Each query returns the top 10 images from database. The two query retrievals are shown in Figure 2.5 and 2.6

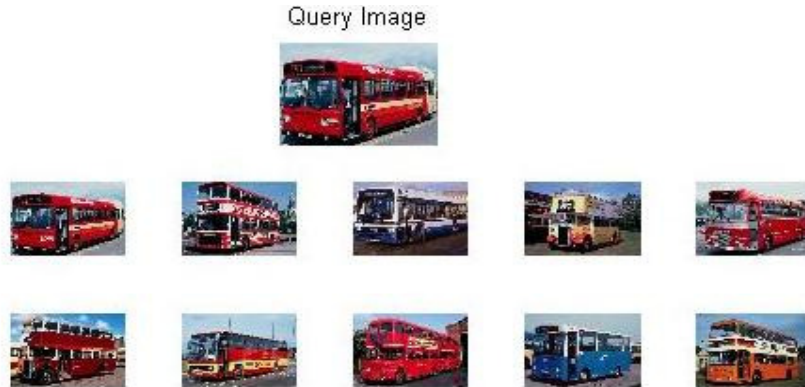


Figure 2.5: Best top 10 matches of a sample bus image query.

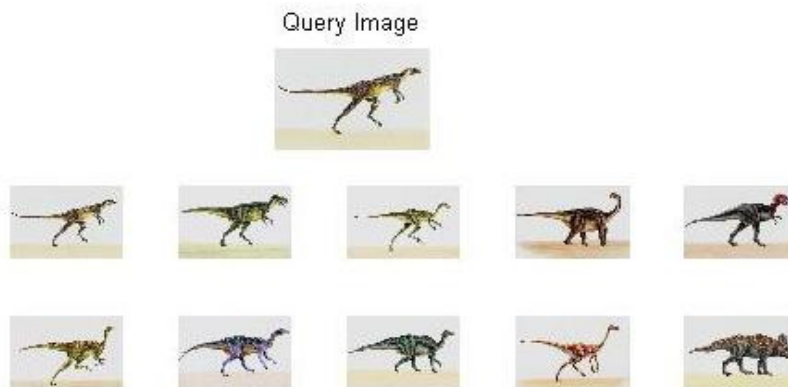


Figure 2.6: Best top 10 matches of a sample dinosaur's image query.

3. CONCLUSIONS

We devised low resolution web image retrieval system using clustering approaches. Our proposed model uses a modified clustering algorithm to improve image segmentation and discusses new similarity distance measure where object is speculated and considered during computation. This fastens the retrieval system to achieve better precision and recall. We have conducted experiments to evaluate the performance of the system by similarity measure using a threshold value and object uniqueness. From the experiment conducted it was observed in experiment #2, the object uniqueness measures gives better results than experiment #1, as it gives more weight to the object that are more distinctive compared to other objects. The performance of the system is evaluated based on similarity measures using a threshold value and object uniqueness. Our proposed model gives better result in some of the classes and performs worse when the image is complex and the objects have smooth edges. During the implementation, we also proved that by considering object uniqueness during similarity distance computation improve the accuracy during retrieval. This work can be extended to classify the images using an advanced classifier to achieve better performance for a particular class.

ACKNOWLEDGEMENTS

We thank the Penn State, University Park, James Z. Wang Research Group for making the low resolution web-crawled image data available for this study.

REFERENCES

- [1] <http://images.google.com/>
- [2] <http://www.ditto.com/>
- [3] <http://www.altavista.com/>
- [4] Rui, Y., Huang, T.S., Chang, S.-F., 1999. Image retrieval: current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation* 10 (1), 39-62.
- [5] Flickner, M., Sawhney, H., Niblack, W., et al., 1995. Query by image and video content: the QBIC system. *IEEE Computer* September, 23-31.
- [6] Pentland, A., Picard, R.W., Sclarof, S., 1994. Photobook: tools for content-based manipulation of image databases. In: *Storage and Retrieval for Image and Video Databases II*. In: *SPIE Proceedings Series*, Vol. 2185. San Jose, CA, USA.
- [7] Minka, T.P., 1996. An image database browser that learns from user interaction. Master's thesis, M.I.T., Cambridge, MA.
- [8] Michael J. Swain., Charles Frankel., and Vassilis Athitsos, "WebSeer: An Image Search Engine for the World Wide Web", Technical Report 96-14, 1997.
- [9] J. R. Smith, "Integrated Spatial and Feature Image Systems: Retrieval, Compression and Analysis". PhD thesis, Graduate School of Arts and Sciences, Columbia University, February 1997.
- [10] S. Sclaroff., L. Taycher., and M. La Cascia, "Imagerover: A content-based image browser for the world wide Web". In *Proceedings IEEE Workshop on Content-based Access of Image and Video Libraries*, June '97, 1997.
- [11] Bach, J.R., Fuller, C., Gupta, A., et al., 1996. The Virage image search engine: an open framework for image management. In: Sethi, I.K., Jain, R.J. (Eds.), *Storage and Retrieval for Image and Video Databases IV*. In: *SPIE Proceedings Series*, Vol. 2670. San Jose, CA, USA.
- [12] Li, J., Wang, J. Z. and Wiederhold, G., (2000), "Integrated Region Matching for Image Retrieval," *ACM Multimedia*, p. 147-156.
- [13] Zhang, R. and Zhang, Z., (2002), "A Clustering Based Approach to Efficient Image Retrieval", *Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence*, pp. 339.
- [14] Daubechies, I., (1992), "Ten Lectures on Wavelets," Capital City Press, Montpelier, Vermont.
- [15] Guha, S., Rastogi, R., and Shim, K., (1998), "CURE: An Efficient Clustering Algorithm for Large Databases" *Proc. of ACM SIGMOD International Conference on Management of Data*, pp.73-84.

Authors

Umesh K. K. is a faculty member with the Department of Information Science and Engineering, Sri Jayachamarajendra College of Engineering, Mysore. He is also a Research Scholar, Department of Studies of Mysore and M.Tech degree in 2003 from Visvesvaraya Technological University, Karnataka. He is working towards doctoral work under the supervision of Dr. Suresha.



Dr. Suresha is presently working as Associate Professor, Department of Studies in Computer Science, University of Mysore. He received his B.Sc. degree in 1987 from University of Mysore, M.Sc degree in 1989 from University of Mysore and M. Phil degree in 1991 from DAVV, Indore. He received M.Tech degree from Indian Institute of Technology, Kharagpur in 1996 and Ph. D. from prestigious Indian Institute of Science, Bangalore in 2007. He was awarded second prize in IRISS-2002 competition, which is an all India level research student competition called “Inter Research Institute Student Seminar”. He has a teaching experience of 21 years at post graduate level, has 12 publications to his credit and currently supervising five research scholars towards their doctoral work.

