2. Feature Representations

Feature extraction and representation is the fundamental process behind CBIR systems. As mentioned, features are properties of the image extracted with image processing algorithms, such as color, texture, shape, and edge information. Our discussion will focus on three general features representations that have been extensively studied in the literature: color, texture, and shape. However, there is no single "best" feature that gives accurate results in any general setting.

2.1 Color feature

The first and most straightforward feature for indexing and retrieving images is color, the basic constituent of images (we consider gray-scale a color). All other information computed by image processing algorithms start with the color information contained in an image. Color moments have been successfully used in many retrieval systems especially when the image contains just the object. The first order (mean), the second (variance) and the third order (skewness) color moments have been proved to be efficient and effective in representing color distributions of images.

2.2 Texture

In computer vision, texture is defined as all what is left after color and local shape have been considered or it is defined by such terms as structure and randomness. The ability to retrieve images on the basis of texture similarity may not seem very useful. But the ability to match on texture similarity can often be useful in distinguishing between areas of images with similar color (such as sky and sea, or leaves and grass).

2.3 Shape

The ability to retrieve by shape is perhaps the most obvious requirement at the primitive level. Unlike texture, shape is a fairly well-defined concept – and there is considerable evidence that natural objects are primarily recognized by their shape.

3. User Interaction

For content-based image retrieval, user interaction with the retrieval system is crucial since flexible formation and modification of queries can only be obtained by involving the user in the retrieval procedure. User interfaces in image retrieval systems typically consist of a query formulation part and a result presentation part.

3.1 Query Specification

Specifying what kind of images a user wishes to retrieve from the database can be done in many ways. Commonly used query formations are: category browsing, query by concept, query by sketch, and query by example. Category browsing is to browse through the database according to the category of the image. For this purpose, images in the database are classified into different categories according to their semantic or visual content .Query by concept is to retrieve images according to the conceptual description associated with each image in the database. Query by sketch and query by example is to draw a sketch or provide an example image from which images with similar visual features will be extracted from the database.

3.2 Relevance Feedback

Human perception of image similarity is subjective, semantic, and task-dependent. Although content-based method provides promising directions for image retrieval; generally, the retrieval results based on the similarities of pure visual features are not necessarily perceptually and semantically meaningful. In addition, each type of visual feature tends to capture only one aspect of image property and it is usually hard for a user to specify clearly how different aspects are combined. To address these problems, interactive *relevance feedback*, a technique in traditional text-based information retrieval systems, was introduced. With relevance feedback, it is possible to establish the link between high-level concepts and low-level features. For a given query, the system first retrieves a list of ranked images according to a predefined similarity metrics. Then, the user marks the retrieved images as relevant (positive examples) to the query or not relevant (negative examples). The system will refine the retrieval results based on the feedback and present a new list of images to the user. Hence, the key issue in relevance feedback is how to incorporate positive and negative examples to refine the query and/or to adjust the similarity measure.

4. Neural Network

Neural network is a network of "neuron like" units called nodes. The architecture of neural network consists of a large number of nodes and interconnection of nodes. A multiple-input neuron with multiple inputs '*R*' is shown in Figure 1. The individual inputs are each weighted by corresponding elements of the weight matrix. The neuron also has a bias '*b*', which is summed with the weighted inputs to form the net input '*n*'s.

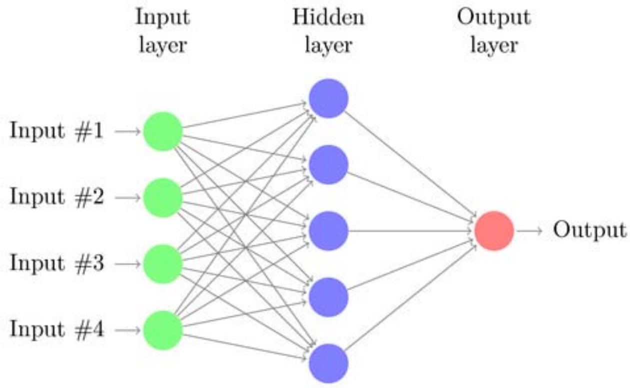

Figure 4-3: Neural Network

5. K-Means Algorithm

K-means (Macqueen) is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. K Means algorithm in CBIR systems is used to initialise the clusters. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycentre of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more. Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 ,$$

Where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre c_j, is an indicator of the distance of the *n* data points from their respective cluster centres. The k-means algorithm can be run multiple times to reduce this effect. The algorithm is composed of the following steps:

- Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
- Assign each object to the group that has the closest centroid.
- When all objects have been assigned, recalculate the positions of the K centroids.
- Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

6. Proposed Work

Step-1 The training process includes the conguguring the neural network then train them about image features like color texture etc. Training helps in building up of knowledge base. Later it is used in matching and decision level.

Algorithm for Training Process:

- *setup ANN and initialize the following parameters as: number_of_layers= 3; epochs=2000;*
- *learning_rate=80%; permissible_error=0.03;*
- ***input:** network, training set*
- ***do***
- ***for** each image in training set*
- *extract its color features using color histogram algorithm;*
- *extract its edge features using edge histogram algorithm;*
- *extract its texture features using normalized r2;*
- *fuse the extracted features into a single features matrix;*
- ***until** a single feature vector matrix is built;*
- ***do***
- *train the network about class labels and feature vectors;*
- ***until** stopping criterion epochs=2000 is satisfied*
- ***Output:** a trained neural network.*

Step-2 The validation phase in neural network, checks whether the image that we are giving in according to input query image is correct or not. If it is correct i.e. features are matched then it is retrieved otherwise again searching takes place.

Algorithm for validation phase:

- ***for** each image i , validation*
- *V = Validate each image*
- *T = desired output for i*
- ***until** some features are matched*
- ***return** the network*
- ***output**: validate image*

Step-3 The testing phase in neural network includes the querying and retrieving task. The query image is first preprocessed and also its features are extracted. The trained network is presented with query image features. The network, acting as a classifier, selectively retrieves top matched, relevant, similar images as that of query image from the database and are presented to user.

Algorithm for Testing Phase:

- ***Input:** a query image.*
- *load the input query image;*
- *extract its color, edge and texture features;*
- *load the fused features database;*
- *compute similarity between query image features and training set features;*
- ***output:** set of similar images if present; if not, display "No similar images found"*

7. Results

The results of the research are given below:

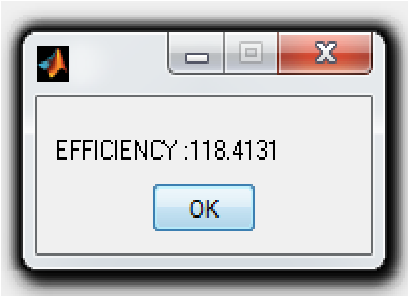

Figure 1.1: Efficiency

The fig. 1.1 shows the efficiency.

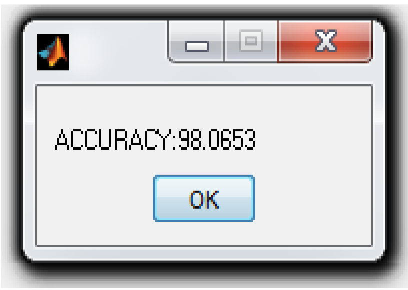

Figure 1.2: Accuracy

The fig. 1.2 shows the accuracy.

Figure 1.3: Performance Parameters

In the similar way as shown in fig. 1.1 & 1.2, the fig. 1.3 shows all the performance parameters i.e. Efficiency, Specificity, Sensitivity, Precision, Recall, Accuracy, G-Mean and False-Measure.

8. Graph

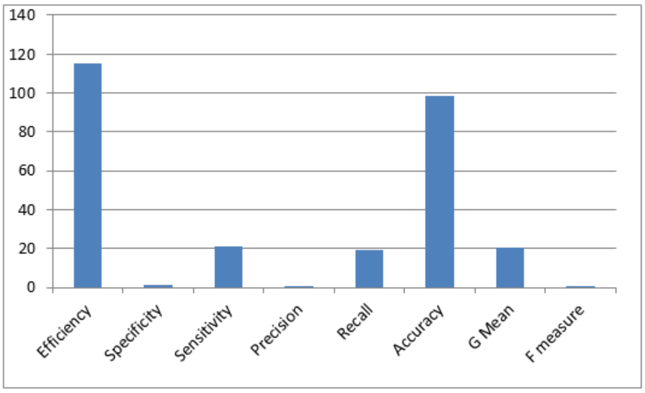

Figure 1.4: Performance Graph

The fig. 1.4 shows the average performance parameters for proposed method i.e. Efficiency, Specificity, Sensitivity, Precision, Recall, Accuracy, G-Mean and False-Measure.

Table 5.1: Performance Table

Efficiency	*Specificity*	*Sensitivity*	*Precision*	*Recall*	*Accuracy*	*G Mean*	*F measure*
117	1.207	21.29	0.80	19.3	98.2	20.6	0.68

Table 5.1 shows the average performance parameters for the proposed method of CBIR.

Figure 1.5: Performance Graph for Diacom Images

The fig. 1.5 shows the average performance parameters for diacom image i.e. Efficiency, Specificity, Sensitivity, Precision, Recall, Accuracy, G-Mean and False-Measure.

Table 5.2: Performance of Diacom Images

Efficiency	*Specificity*	*Sensitivity*	*Precision*	*Recall*	*Accuracy*	*G Mean*	*F measure*
117	1.207	8	60	8.00	98	6.0	2.08

Table 5.2 shows the average performance parameters for diacom image.

Table 5: Classification Rate and Time to Compute

S. No	*Performance Values*
Classification Rate	96-98%
Time to compute	5-6 sec

Table 5.3 shows the classification rate and computing time of the system.

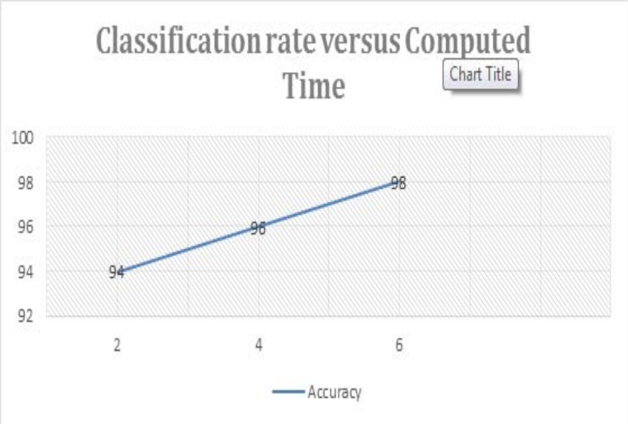

Figure 1.6: Classification Rate and Computing Time

Above graph shows the how much time is needed to classify the images when query image is inputted to the data-base. Less time means, high accuracy of the CBIR systems.

9. Conclusion

This research has presented a CBIR system using back-propagation neural network and K-Mean algorithm. The color distribution histograms are used as color information of an image. Also, normalized r2, rgb pattern to help characterize the images. The use of back-propagation neural network and K-Mean algorithm has considerably improved the accuracy, recall rate and also retrieval time, due to its highly efficient and accurate classification capability. Also, the back-propagation neural network and K-Mean algorithm has increased the retrieval precision due to its capability of minimizing the error during training process itself. Future scope includes implementing the CBIR system considering more low-level image descriptors and highly efficient deep learning neural network, which might prove to be very fast and precised one.

References

[1] T. Dharani I. Laurence Aroquiaraj, "*A Survey on Content Based Image Retrieval*", IEEE, pp.485-490, 2013.

[2] I. Felci Rajam and S. Valli, "*A Survey on Content Based Image Retrieval*", Life Science Journal, pp. 2475-2487, 2013.

[3] Chih-Chin Lai, and Ying-Chuan Chen, "*A User-Oriented Image Retrieval System Based on Interactive Genetic Algorithm*," IEEE TRANS. Instrumentation and measurement, VOL. 60, NO. 10, pp. 3318 – 3325, OCTOBER 2011.

[4] Tasweer Ahmad, Ahlam Jameel, Dr. Balal Ahmad, "*Pattern recognition using statistical and neural techniques*," IEEE CONFERENCE. Computer Networks and Information Technology, pp. 87 - 91, 11-13 July 2011.

[5] Y. D. Chun, N. C. Kim, and I. H. Jang, "*Content-based image retrieval using multi-resolution color and texture features*," IEEE Trans. Multimedia, vol. 10, no. 6, Oct. 2008.

[6] M. Pi and H. Li, "*Fractal indexing with the joint statistical properties and its application in tesxture image retrieval*," IET Image Process., vol. 2, no. 4, pp. 218–230, Aug. 2008.

[7] T.-C. Lu and C.-C. Chang, "*Color image retrieval technique based on color features and image bitmap*," Inf. Process. Manage. vol. 43, no. 2, Mar. 2007.

[8] M. H. Pi, C. S. Tong, S. K. Choy, and H. Zhang, "*A fast and effective model for wavelet subband histograms and its application in texture image retrieval*," IEEE Trans. Image Process., vol. 15, no. 10, Oct. 2006.

[9] M. Kokare, P. K. Biswas, and B. N. Chatterji, "*Texture image retrieval using new rotated complex wavelet filters*," IEEE Trans. Syst., Man, Cybern. B, Cybern, vol. 35, no. 6, Dec. 2005.

[10] H.-W. Yoo, H.-S. Park, and D.-S. Jang, "*Expert system for color image retrieval*," Expert Syst. Appl., vol. 28, no. 2, pp. 347–357, Feb.2005.

[11] S. Liapis and G. Tziritas, "*Color and texture image retrieval using chromaticity histograms and wavelet frames*," IEEE Trans. Multimedia, vol. 6, no. 5, Oct. 2004.

[12] M. Egmont-Petersen, D. De Ridder, and H. Handels, "*Image processing with neural networks - a review*," Pattern Recognition, 35:2279-2301, 2002.

[13] Andre Folkers and Hanan Samet, "*Content based image retrieval using Fourier descriptors on a Logo database*", IEEE Proceedings of 16th International Conference of Pattern Recognition, 2002, vol. 3 pp. 521-524.

[14] T. Sikora, "*The MPEG-7 visual standard for content description—an overview*," IEEE Trans. Circuits Syst. Video Technol., vol. 11, no. 6, Jun. 2001.

[15] Xu Chenyang and Jerry L. Prince, "*Gradient Vector Flow: A New External Force for Snakes* ", IEEE Transactions on Image and multidimensional Signal Processing Workshop, 1996.

Author Profile

Gurpreet Kaur, Student of M.Tech. in the department of Computer Science and Engineering at Swami Vivekanand Institute of Engineering and Technology, Banur (Punjab). She has done B.Tech. (2012) from Yadvindra College of Engineering, Talwandi Sabo under Punjabi University, Patiala (Punjab).

Er. Mnupreet Kaur, Assistant Professor in the department of Computer Science & Engineering at Swami Vivekanand Institute of Engineering & Technology, Banur under Punjab Technical University, Jalandhar. She has done M.Tech (2013) from Punjabi University, Patiala & B. Tech. (2007) from Punjab Technical University, Jalandhar.