Content Based Image Retrieval system Using K-Means Clustering Technique

Deepika Nagthane
(Assistant Professor)
Department of CSE,
MVJ College of Engineering, Bangalore.

Abstract

In a typical CBIR system, low- level visual image features that is colour, texture, and shape are automatically extracted for image descriptions and indexing purposes. To search for desirable images, a user presents an image as an example of similarity, and the system returns a set of similar images based on the extracted features. In CBIR systems with relevance feedback (RF) [3], a user can mark returned images as positive or negative, which are then fed back into the systems as a new, refined query for the next round of retrieval. The process is repeated until the user is satisfied with the query result. Such systems are effective for many practical CBIR applications.

Target search in content-based image retrieval (CBIR) systems refers to finding a specific (target) image such as a particular registered logo or a specific historical photograph. Existing techniques, designed around query refinement based on relevance feedback (RF), suffer from slow convergence, and do not guarantee to find intended targets. To address the aforementioned limitations, four target search methods are proposed Naive Random Scan Method (NRS), local neighbouring movement (LNM), and neighbouring divide-and conquer (NDC), and global divide-and-conquer (GDC) methods. The target search methods uses the concept of voronoi diagram[1] approach to aggressively prune the search space and move toward the target image with minimum iteration.

To improve the results K-means clustering technique is used. K-means clustering technique is helpful to reduce the elapsed time of the system for all of the target search methods. So, the elapsed time of the presented CBIR system is reduced effectively.

I.INTRODUCTION

Content-based image retrieval (CBIR) [2] also known as query by image content and content-based visual information retrieval is the application of computer vision techniques to the image retrieval problem, that is, the problem of searching for digital images in large databases. "Content-based" means that the search will analyze the actual contents of the image rather than the metadata such as keywords, tags, and/or descriptions associated with the image. The term 'content' refer to colours, shapes, textures, or any other information that can be derived from the image itself. CBIR is desirable because most web based image search engines rely purely on metadata and this produces a lot of garbage in the results. Also having humans manually enter keywords for images in a large database can be inefficient, expensive and may not capture every keyword that describes the image. Thus system that can filter images based on their content would provide better indexing and return more accurate results.

Advances in data storage and image acquisition technologies have enabled the creation of large image datasets. To deal with these data, it is necessary to develop appropriate information systems to efficiently manage these collections. Image searching is one of the most important services that need to be supported by such systems. Two different approaches have been applied to allow searching on image collections: one based on image textual metadata and another based on image content information. The first retrieval approach is based on attaching textual metadata to each image and use traditional database query techniques to retrieve them by keywords [1, 2]. However, these systems require a previous annotation of the database images, which is a very laborious task. Furthermore, the annotation process is usually inefficient because users, generally, do not make the annotation in a systematic way.In fact, different users tend to use different words to describe a same image characteristic. The lack of systematization in the annotation process decreases the performance of the keyword- based image search. Conventional information retrieval is based solely on text, and these approaches to textual information retrieval have been transplanted into image retrieval in a variety of ways, including the representation of an image as a vector of feature values. However, *"a picture is worth a thousand words."* Image contents are much more versatile compared with text, and the amount of visual data is already enormous and still expanding very rapidly. To cope with these special characteristics of visual data, content- based image retrieval methods have been introduced. It has been widely recognized that the family of image retrieval techniques should become an integration of both low-level visual features, addressing the more detailed perceptual aspects, and high-level semantic features underlying the more general conceptual aspects of visual data. Neither of these two types of features is sufficient to retrieve or manage visual data in an effective or efficient way. More efforts have been devoted to combining these two aspects of visual data, the gap between them is still a huge barrier. Intuitive and heuristic approaches do not provide with satisfactory performance. Again in the CBIR system with Relevance Feedback (RF) [5] a user can mark the returned images as positive and negative which are then fed back into a system as a new, refined query for next round of retrieval. The process is repeated until the user is satisfied with query result. So, more number of iterations is required to lead to the target image and again CPU time will be increased .Therefore, there is an urgent need of finding and managing the talent correlation between low-level features and high-level concepts.

A. ARCHITECTURE OF CBIR SYSTEM

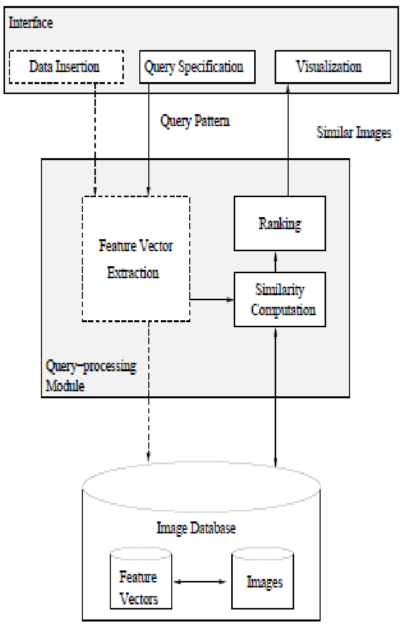

Fig1.1 Typical Architecture Of Content Based Image Retrieval System

The figure1 shows a typical architecture of a content-based image retrieval system. Two main functionalities are supported: data insertion and query processing. The data insertion subsystem is responsible for extracting appropriate features for mining and storing them into the image database. This process is usually performed off-line. The query processing, in turn, 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 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.

B.PROBLEMS WITH EXISTING CBIR SYSTEMS

In CBIR systems with relevance feedback (RF), a user can mark returned images as positive or negative, which are then fed back into the systems as a new, refined query for the next round of retrieval. The process is repeated until the user is satisfied with the query result. Such systems are effective for many practical CBIR applications. There are two general types of image search: target search and category search [10], [11]. The goal of target search is to find a specific (target) image, such as a registered logo, a historical photograph, or a particular painting. The goal of category search is to retrieve a given semantic class or genre of images, such as scenery images or sky scrapers. In other words, a user uses target search to find a known image. In contrast, category search is used to find relevant images the user might not be aware ahead of time.

Two orthogonal issues in CBIR research are efficiency and accuracy. An effective CBIR system, therefore, needs to have both an efficient search mechanism and accurate set of visual features. The Euclidean distances between the images reflect their semantic similarity, and focus on investigating new search techniques to improve the efficiency of target search. Existing target search techniques re- retrieve previously examined images (i.e., those retrieved in the previous iterations) when they again fall within the search range of the current iteration. This strategy leads to the following disadvantages:

A. Local Maximum Trap Problem:-

The search operation generally takes several iterations of RF to examine a number of regions in the feature space, before it reaches the target image. During this iterative process, the search advancement might get trapped in a region as illustrated in Fig. 2

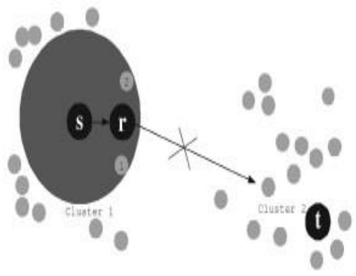

Fig 1.2 Local maximum trap

Fig1.2 shows that **s** and **t** as the starting point ps and the target point pt, respectively. Initially, the 3-NN search with ps as the query point yields three points ps, p1, and p2 as the query result. Let us say, the user marks points p1 and p2 as relevant. This results in point pr, their centroid, as the new query point. With pr as the refined query, the next 3-NN computation again retrieves point p1, p2, and ps as the result. In this scenario, the search process is trapped in this local region, and can never reach the target point pt. Although, the system can escape the local maximum trap with a larger k, it is difficult to guess a proper threshold. Consequently, the user might not even know a local maximum trap is occurring.

B. Slow Convergence Problem

Including previously examined images in the computation of the current centroid results in repeat retrieval of some of the images. This prevents a more aggressive movement of the search in the feature space. This drawback is illustrated in Fig. 3 where k =3. It shows that it takes six iterations for the search operation to reach the target point pt. This slow convergence incurs longer search time, and significant computation and disk access overhead.

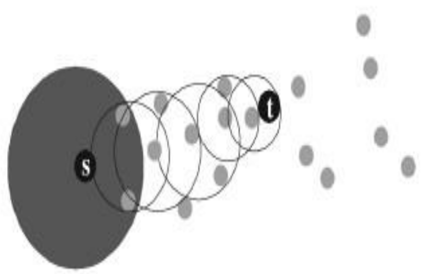

Fig1. 3. Slow convergence.

To address the aforementioned limitations, four target search methods are proposed naive random scan (NRS), local neighbouring movement (LNM), and neighbouring divide-and conquer (NDC), and global divide-and-conquer (GDC) methods. All these schemes are built around a common strategy: they do not re examine previously checked images.

II .TARGET SEARCH SYSYEM

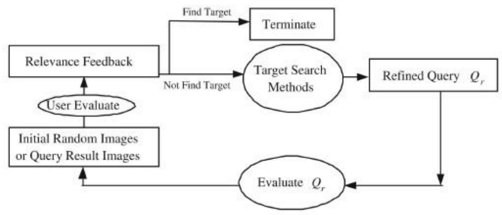

Fig 2.1 Overview of target search system

Fig 2.1 shows the target search system. In this system user chooses initial random images or the query result images and then evaluate that images. User marks the returned images as positive or negative by applying the concept of relevance feedback if user finds the target image search will be terminated. Otherwise one of the target search method will be applied to that current image database and refined query will be constructed. The query is again evaluated by user.

In target search [2], the ultimate goal is to locate the target images, and if none is found, the final precision and recall of the search is zero. In CBIR with RF, the traditional recall and precision can be computed for individual iterations. For target search, "aggregate" recall and precision is used if after several, say i, iterations the target image is found, the average precision and recall are 1/(i.k) and 1/i, where k is the fixed number of images retrieved at each iteration. In short, the number of iterations to find a target image is not only the most significant measure of efficiency but also the most significant indicator of precision and recall. query (Q) for target search is defined as Q = $\langle n_q, P_q, W_q, D_q, S', k\rangle$ where n_q denotes the number of query points in Q, P_q denotes the set of n_q query points in the current search space S', W_q denotes the set of weights associated with P_q, D_q denotes the distance function, and k denotes the number of points to be retrieved in each iteration. Various techniques have been proposed to automatically determine n_q and P_q as well as adjusting W_q and D_q for improved retrieval effectiveness. For single-point movement techniques, n_q=1 for multiple point movement techniques, n_q>1.

The generalized definition of query is given as Q = <n_q,P_q,W_q,D_q> defined in [8], where the search space is assumed to be the whole database for every search. In generalized definition, S' is included to account for the dynamic size of the search space, which shrinks gradually after each iteration. Let Q_s denote the starting query, Q_r denote a refined query at a feedback iteration, Q_t denote a target query which results in the retrieval of the intended target, and S_k denote the query result set.

2.1 NAÏVE RANDOM SCAN METHOD

The NRS method [2] randomly retrieves k different images at a time until the user finds the target image or the remaining set is exhausted. Specifically, at each iteration, a set of k random images are retrieved from the candidate (unchecked) set S' for RF (lines 2 and 6), and S' is then reduced by k (lines 3 and 7). The algorithm 2.1.1 gives the detailed description about how NRS method works.

2.1.1 Algorithm for NRS method

NAIVERANDOMSCAN(S, k)

Input:

Set of images S
Number of retrieved images at each iteration

Output:

Target image P_t

1 $Q_s \leftarrow$ < 0,P_q,W_q,D_q,S,k >
2 $S_k \leftarrow$ EVAULATEQUERY(Q_s)
/* randomly retrieve k points in S */
3 S' ← S − S_k
4 **while** user does not find P_t in S_k do
5 $Q_r \leftarrow$ < 0,P_q,W_q,D_q,S',k >
6 $S_k \leftarrow$ EVAULATEQUERY(Q_r)
/* randomly retrieve k points in S'*/
7 S' ← S'− S_k
8 **End do**
9 return P_t

Clearly, the naive scan algorithm does not suffer local maximum traps and is able to locate the target image after some finite number of iterations. But NRS is only suitable for a small database set.

2.2 LOCAL NEIGHBORING MOVEMENT METHOD

Existing techniques allow already checked images to be reconsidered, which leads to several major drawbacks. And by applying this non re-retrieval strategy to one such method, such as MindReader [2], to produce the LNM method. LNM is similar to NRS except lines 5 and 6 as follows. The algorithm 2.2.1 gives the detailed steps how LNM method works and how it approaches to the target image.

2.2.1 Algorithm for LNM method.

LOCALNEGHIBORINGMOVEMENT(S, k)

Input:

set of images S
number of retrieved images at each iteration k

Output:

Target image P_t

1 $Q_s \leftarrow$ < 0,P_q,W_q,D_q,S,k >
2 $S_k \leftarrow$ EVAULATEQUERY(Q_s)
3 S' ← S − S_k
4 **while** user does not find P_t in S_k do
5 $Q_r \leftarrow$ <n_q,P_q,W_q,D_q,S',k>
6 $S_k \leftarrow$ EVAULATEQUERY(Q_r)
/* perform a constrained k-NN query */
7 S' ← S'− S_k
8 **End do**
9 return P_t

When LNM encounters a local maximum trap, it enumerates neighbouring points of the query, and selects the one closest to the target. Therefore, LNM can overcome local maximum traps, although it could take much iteration to do so. Again one iteration is required in the best case. To simplify the worst-case and average-case complexity analysis, S is uniformly distributed and the distance between two nearest points is a unit.

2.3 NEIGHBORING DIVIDE AND CONQUER METHOD

Although LNM can overcome local maximum traps, it does so inefficiently, taking many iterations and in the process returning numerous false hits. To speed up convergence, we propose to use Voronoi diagrams [1], [28] in NDC to reduce search space.

2.3.2 Algorithm for NDC method

NEGHIBORINGDIVIDECONQUER (S, k)

Input:
Set of images S
number of retrieved images at each iteration k

Output:
target image P_t

1 $Q_s \leftarrow < 0, P_q, W_q, D_q, S, k >$
2 $S_k \leftarrow$ EVAULATEQUERY(Q_s) /* randomly retrieve k points in S */
3 $VR_i \leftarrow$ the minimum bounding box of S
4 iter $\leftarrow$ 1
5 **while** user does not find p_t in S_k **do**
6 **if** iter $\neq 1$ **then**
7 $\quad S_{k+1} \leftarrow S_k + \{p_i\}$
8 **else**
9 $\quad S_{k+1} \leftarrow S_k$
10 **endif**
11 $P_i \leftarrow$ the most relevant point € S_{k+1}
12 construct a Voronoi diagram VD inside VR_i using points in S_{k+1}as Voronoi seeds
13 $VR_i \leftarrow$ the Voronoi cell region associated with the Voronoi seed p_i in VD
14 S' $\leftarrow$ such points € S that are inside VR_i except pi
15 $Q_r \leftarrow < 1, \{P_i\}, W_q, D_q, S', k>$
16 $S_k \leftarrow$ EVALUATEQUERY(Q_r) /* perform a constrained k-NN query */
17 iter $\leftarrow$ iter +1
18 **enddo**
19 **return** P_t

From the starting query Q_s, k points are randomly retrieved (line 2).Then, the Voronoi region (VR) is initially set to the minimum bounding box (MBB) of S (line 3). In the while loop, NDC first determines the Voronoi seed set S_{k+1}(lines 6 to 10) and P_i, the most relevant point in S_{k+1} according to the user's RF (line 11). Next,

it constructs a Voronoi diagram VD inside VR_i using S_{k+1}(line 12). The Voronoi cell region containing P_i in VD is now the new VR_i (line 13). Because only VR_i can contain the target other voronoi cell region can be safely pruned. To continue the search in VR_i, NDC constructs a k-NN query using P_i as the anchor point (line 15), and evaluates it (line 16). The procedure is repeated until the target P_t is found. When NDC encounters a local maximum trap, it employs Voronoi diagrams to aggressively prune the search space and move toward the target image, thus significantly speeding up the convergence. Therefore, NDC can overcome local maximum traps and achieve fast convergence

2.4 GLOBAL DIVIDE AND CONQUER METHOD

To reduce the number of iterations in the worst case in NDC, we propose the GDC method. Instead of using a query point and its neighboring points to construct a Voronoi diagram, GDC uses the query point and k points randomly sampled from VR_i. The algorithm 2.4.1 gives the detailed description about how the global divide and conquer method works. It uses the concept of voronoi diagram to purne the search space and move to the target image with minimum iterations

2.4.1 Algorithm for GDC method.

GLOBALGDIVIDECONQUER (S , k)

Input:
Set of images S
number of retrieved images at each iteration k

Output:
target image P_t

$Q_s \leftarrow < 0, P_q, W_q, D_q, S, k >$
$S_k \leftarrow$ EVAULATEQUERY(Q_s)/* randomly retrieve k points in S */
$VR_i \leftarrow$ the minimum bounding box of S
iter $\leftarrow$ 1
while user does not find p_t in S_k **do**
if iter $\neq 1$ **then**
$\quad S_{k+1} \leftarrow S_k + \{p_i\}$
else
$\quad S_{k+1} \leftarrow S_k$
endif
$P_i \leftarrow$ the most relevant point € S_{k+1}
construct a Voronoi diagram VD inside VR_i using points in S_{k+1}as Voronoi seeds
$VR_i \leftarrow$ the Voronoi cell region associated with the Voronoi seed p_i in VD
S' $\leftarrow$ such points € S that are inside VR_i except p_i

```
Qr ← < 0,{Pi},Wq,Dq,S',k>
Sk ← EVALUATEQUERY(Qr) /* randomly retrieve k in S' */
iter ← iter +1
enddo
return Pt
```

There is a not major difference in the working of GDC and NDC method. Except that instead of using a query point and its neighboring points to construct a voronoi diagram GDC uses query point and k points randomly sampled from VR_i .

2.5 K-MEANS CLUSTERING

Clustering is a way of grouping together data samples that are similar in some way according to some criteria that we pick its form of unsupervised learning So, it's a method of data exploration – a way of looking for patterns or structure in the data that are of interest .Clustering algorithms are generally used in an unsupervised fashion. They are presented with a set of data instances that must be grouped according to some notion of similarity. The algorithm has access only to the set of features describing each object; it is not given any information as to where each of the instances should be placed within the partition.

K-means clustering is a method commonly used to automatically partition a data set into k groups. It proceeds by selecting k initial cluster centers and then iteratively refining the results. The algorithm converges when there is no further change in assignment of instances to clusters.

2.5.1 Algorithm for K-means

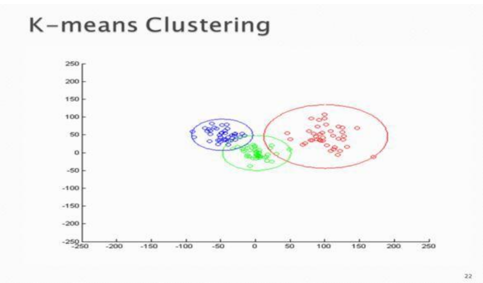

Fig 2.2 Clusters formed for K=3

1 Decide on a value for *k.*
2. Initialize the *k cluster centers (randomly, if* necessary).
3. Decide the class memberships of the *N objects by* assigning them to the nearest cluster center.
4. Re-estimate the *k cluster centers, by assuming the* memberships found above are correct.
5. If none of the *N objects changed membership in* the last iteration, exit. Otherwise go to 3.

In this way the clusters of similar kind are formed and it helps to reduce the elapsed time of the system.

III. EXPERIMENTAL RESULTS

The COREL photo database of ten thousand images is used to evaluate the performance of CBIR system. We have selected a database of almost six thousand images and then divide the database into number of concepts of different complexities to verify the results. Here we have chosen the butterfly concept and worked on that by applying the target search methods on that database we have got the most relevant images.

3.1 Query Image selection

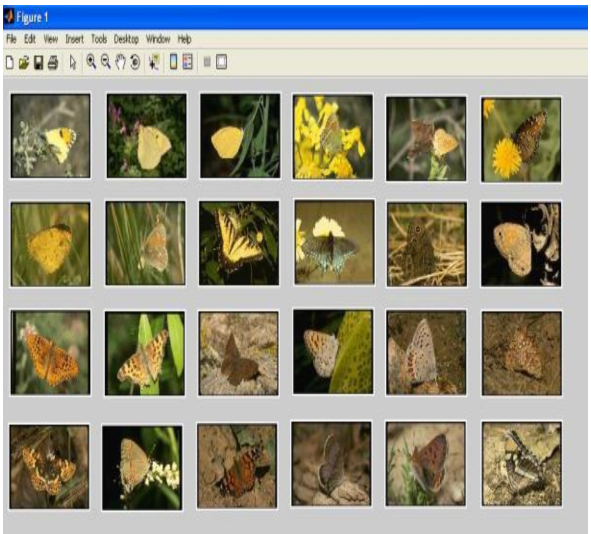

Fig 3.2 Images retrieved

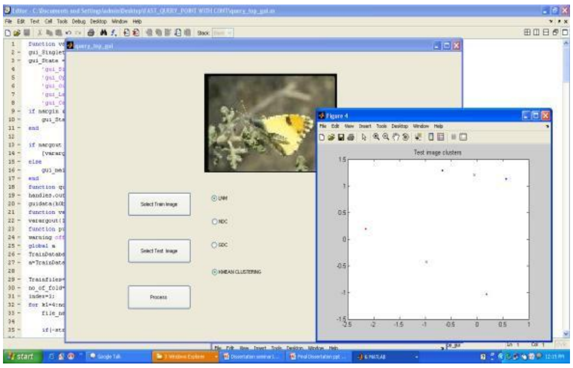

Fig 3.3 K means clustering-test image cluster

A . COMPARING ELAPSED TIME

The table 4.1 shows the comparison among the all fast query point movement techniques and by applying K-means clustering the elapsed time for all these methods is reduced effectively The comparison of time is enlisted in the table for twenty images in the database . The time is measured in seconds for each method without K-means.

Database Images	LNM withcut K-means	LNM with K-means	NDC without K-means	NDC with K-means	GDC without K-means	GDC with K-menas
Image 01	0.094489 sec	0.032653 sec	0.057697 sec	0.000483 sec	0.055015 sec	0.000194 sec
Image 02	0.121260 sec	0.001603 sec	0.069975 sec	0.000276 sec	0.069069 sec	0.000173 sec
Image 03	0.135458 sec	0.000173 sec	0.080516 sec	0.000202 sec	0.081359 sec	0.000172 sec
Image 04	0.147316 sec	0.000207 sec	0.091456 sec	0.000197 sec	0.094677 sec	0.000164 sec
Image 05	0.159560 sec	0.000164 sec	0.102276 sec	0.000267 sec	0.107528 sec	0.000159 sec
Image 06	0.172543 sec	0.000253 sec	0.113274 sec	0.000188 sec	0.120774 sec	0.000208 sec
Image 07	0.185013 sec	0.000202 sec	0.125470 sec	0.000264 sec	0.133738 sec	0.000160 sec
Image 08	0.197754 sec	0.000206 sec	0.137522 sec	0.000250 sec	0.147852 sec	0.000169 sec
Image 09	0.210666 sec	0.000168 sec	0.149059 sec	0.000264 sec	0.162508 sec	0.000158 sec
Image 10	0.224892 sec	0.000155 sec	0.160951 sec	0.000268 sec	0.177625 sec	0.000159 sec
Image 11	0.238985 sec	0.000212 sec	0.172986 sec	0.000259 sec	0.191749 sec	0.000157 sec
Image 12	0.252510 sec	0.000173 sec	0.185315 sec	0.000260 sec	0.205921 sec	0.000158 sec
Image 13	0.266826 sec	0.000158 sec	0.198495 sec	0.000190 sec	0.220182 sec	0.000157 sec
Image 14	0.280755 sec	0.000213 sec	0.212281 sec	0.000354 sec	0.234691 sec	0.000158 sec
Image 15	0.294767 sec	0.000156 sec	0.224875 sec	0.000266 sec	0.250531 sec	0.000168 sec
Image 16	0.309523 sec	0.000203 sec	0.238169 sec	0.000302 sec	0.265215 sec	0.000174 sec
Image 17	0.324479 sec	0.000174 sec	0.251400 sec	0.000309 sec	0.280171 sec	0.000170 sec
Image 18	0.340048 sec	0.000204 sec	0.264870 sec	0.000230 sec	0.311936 sec	0.000168 sec
Image 19	0.356348 sec	0.000165 sec	0.279393 sec	0.000190 sec	0.327583 sec	0.000157 sec
Image 20	0.372134 sec	0.000166 sec	0.294028 sec	0.000189 sec	0.343449 sec	0.000168 sec

Table 1 Comparing the elapsed time.

Table 1 shows the elapsed time forLNM, NDC and GDC method. It's clear that by applying K-means technique the elapsed time for target search methods reduced effectively. There is a big difference in elapsed time for all these methods. From table 4.1 it's clear that the elapsed time for the CBIR system is reduced effectively.

IV.CONCLUSION

This paper summarizes the specific contribution from the work and outlines some direction of future work. Review of previous study reveals that as manual observations are required in the text-based retrieval using content-based approach is more desirable than text-based approach. Color is low level feature. In addition to color, texture feature is used. For CBIR system 'Fast Query Point Movement Methods' are used. The images which are retrieved by using LNM, NDC and GDC methods are quite similar but the difference is there in the elapsed time. The voronoi diagram approach is used in NDC and GDC methods which lead to target image with minimum number of iterations.

Again the performance of the system is improved by using K-Means clustering technique**.** The proposed CBIR system is interactive for image storage and retrieval from image database based on target search method. Under this frame work weight associated with the query point, distance of query point is taken into account to improve retrieval effectiveness and efficiency. The target search methods are used to lead to target image with minimum number of iterations by avoiding local maximum trap and slow convergence problem.LNM method, NDC method and GDC method gives the most relevant images as a query result.The elapsed time for all these target search methods is minimum, the time required for GDC method is less compare to LNM and NDC method.Again by using the K-means clustering method the elapsed time for LNM, NDC and GDC method is reduced effectively. The K-means Clustering technique is working in a real sense as it is again reducing the time, the color feature is used to form the clusters here K=2 is specified for this current CBIR system.The results are improved much with this technique by forming the two means of the color and centroid of the cluster is playing a major role to reduce the elapsed time and lead to query image with minimum time. The results are improved by using the K- Means clustering concept in terms of the elapsed time.

REFERENCES

[1] D. Liu, K.A. Hua, K. Vu, and N. Yu, ''Fast Query Point Movement Techniques for Large CBIR Systems'' IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 21, NO. 5, MAY 2009.

[2] I. Bartolini, P. Ciacci, and F. Waas, "Feedbackbypass: A NewApproach to Interactive Similarity Query Processing," Proc. 27th Int'l Conf. Very Large Data Base (VLDB '01), pp. 201-210, 2001.

[3] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger, "The R-Tree: An Efficient and Robust Access Method for Points and Rectangles," Proc. ACM SIGMOD '90, pp. 322-331, 1990.

[4] C. Bo¨hm, S. Berchtold, and D.A. Keim, "Searching in High- Dimensional Spaces: Index Structures for Improving the Performance of Multimedia Databases," ACM Computing Surveys, vol. 33, no. 3, pp. 322-373, 2001.

[5] P. Browne and A.F. Smeaton, "Video

Information Retrieval Using Objects and Ostensive Relevance Feedback,” Proc. ACM Symp.Applied Computing (SAC ’04), pp. 1084-1090, 2004.

[6] R. Brunelli and O. Mich, “Image Retrieval by Examples,” IEEE Trans. Multimedia, vol. 2, no. 3, pp. 164-171, 2000.

[7] C. Carson, S. Belongie, H. Greenspan, and J. Malik, “Blobworld:Image Segmentation Using Expectation-Maximization and Its Application to Image Querying,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 24, no. 8, pp. 1026-1038, Aug. 2002.

[8] K. Chakrabarti, M. Ortega-Binderberger, S. Mehrotra, and K. Porkaew, “Evaluating Refined Queries in Top-K RetrievalSystems,” IEEE Trans. Knowledge and Data Eng., vol. 16, no. 2, pp. 256-270, Feb. 2004.

[9] L. Chen, M.T. O ¨ zsu, and V. Oria, “MINDEX: An Efficient Index Structure for Salient-Object-Based Queries in Video Databases,” Multimedia Systems, vol. 10, no. 1, pp. 56-71, 2004.

[10] I.J. Cox, M.L. Miller, T.P. Minka, T.V. Papathomas, and P.N. Yianilos, “The Bayesian Image Retrieval System, PicHunter: Theory, Implementation, and Psychophysical Experiments,” IEEE Trans. Image Processing, vol. 9, no. 1, pp. 20-37, 2000.

[11] H. Ferhatosmanoglu, I. Stanoi, D.Agrawal, and A.E. Abbadi, “Constrained Nearest Neighbor Queries,” Proc. Seventh Int’l Symp. Spatial and Temporal Databases (SSTD ’01), pp. 257-278, 2001.