

An Efficient Content Based Image Retrieval Algorithm Using Clustering Techniques For Large Dataset

1stMonika Jain
Department of computer science
Mewar university
Rajasthan, India
monika_smec@yahoo.co.in

2nd Dr S.K.Singh
Department of computer science
GCET
Greater Noida (U.P), India
singhsks123@gmail.com

Abstract— Content based image retrieval (CBIR) is a well chosen fast and accurate retrieval technique. In recent years, a range of techniques have been developed to improve the performance of CBIR. An unsupervised method called data clustering is used for extracting hidden patterns from large data sets. Also, there is possibility of high dimensionality in large datasets. Most of clustering and segmentation algorithms suffer from the curse of high dimensionality and the optimal number of clusters provided by a human user. In this paper, we present a method HDK that optimizes these limitations. It uses more than one clustering technique with efficient indexing based on color feature. A new cluster based similarity measure conforming like human perception is applied and shown to be effective. From the experimental results, it is evident that our system is powerful, accurate and efficient.

Keywords—content based image retrieval, clustering, indexing, high dimensional data, color, number of clusters.

I. INTRODUCTION

Content-Based Image Retrieval (CBIR) is the retrieval of digital pictures from huge databases automatically. This technique makes utilization of the characteristic of inherent visual features of an image to execute a query. Instead of prior image retrieval strategies which included the manual annotation of images based on tags, CBIR framework recognizes the images via consequently extracted comparable features automatically. With the development in innovation and regularly expanding utilization of digital cameras, CBIR turns out to be considerably more productive and practical to oversee and store substantial databases of information. Amid the previous decade, an astounding advancement in CBIR has been made in both theoretical research and framework improvement. Similarity comparison is the main tasks for CBIR systems. It extracts features of each image of the database dependent on its pixel esteems and characterizes rules for comparing images.

Content Based Image Retrieval is extremely helpful in different applications, for example, historical research, crime prevention, medical diagnosis conclusion, architectural and engineering design, geological data and remote sensing frameworks, and so forth [1]. In document retrieval, millions of objects with dimension more than 100 have to be clustered to achieve data

abstraction. Because of the size of image databases on web and in diverse fields, it is important to enhance the classification algorithm to handle a large number of features (i.e high dimensional dataset).

We propose a framework which joins relational meaning of clustering space and separate and overcome strategy to enhance effectiveness and exactness in CBIR frameworks for extensive and high dimensional datasets. For this purpose, we need efficient indexing for retrieval of images in large and high dimensional database. Proposed indexing technique would apply in a substantial dataset with high dimensional space, separating huge space into subspaces horizontally which lead us to high efficiency and exactness.

So as to compare images by content, a feature vector should be determined for each image. This feature vector gives the description of the picture for similarity matching. Highlight extraction strategies attempt to find statistical regularities (or once in a while inconsistencies) specifically from the information sources. Among low level features, Color is most appropriate feature of an image used in CBIR [2,3,4].

The retrieval becomes more efficient when one or more low-level features such as texture or shape, is added to the system. But again compiling more than one feature takes a lot of time.

By applying Clustering on determined feature vectors, marks of comparable images are grouped together as one cluster. While querying, a CBIR framework require just take a representative for each group to limit the search. In this paper we are concentrating on few clustering techniques. Section 2 is a short overview of hierarchical and divides and conquer k-Means clustering algorithm. The proposed framework is explained in section 3. The experimental results and performance are shown in section 4. Section 5 is conclusion.

II. HIERARCHICAL AND DIVIDE AND CONQUER K MEANS CLUSTERING TECHNOLOGY

Image clustering is unsupervised learning strategy utilized for data retrieval reason and dependent on some low dimension highlights. It groups an arrangement of

image information in clusters to expand the comparability and limit the similarity between various clusters.

Clearly from the earlier information of the number of clusters isn't required in the hierarchical technique. Be that as it may, various leveled strategy requires broad computational assets as vast divergence lattices must be figured and put away for substantial informational indexes. In any case, this isn't an issue for its utilization as one can utilize it as preprocessing for whole database. Under unsupervised learning process, it is extremely hard to predict number of clusters in images without information of data.

The hierarchical clustering and its varieties are regularly utilized for image clustering for its simplicity and ease of use. Agglomerative or divisive are two types of hierarchical algorithms. The agglomerative (bottom-up) approach more than once combines two clusters, while the divisive (top-down) approach over and over parts a cluster into two. In troublesome, for a cluster with n objects, there are $2^{n-1}-1$ conceivable two-subset divisions, which is over the top expensive in calculation [10]. Along these lines, divisive clustering is not normally utilized in practice. Numerous new HAC procedures have showed up and extraordinarily enhanced the clustering performance. Typical examples include "CURE" [6], "ROCK" [7], "Chameleon" [8] and "BIRCH" [9].

For agglomerative hierarchical cluster(HC) investigation on an informational index, we pursue this methodology [5]:

(a) Find the similarity or difference between each match of items in the informational collection dependent on measurements, for example, "Euclidean", "minkowski", "chebychev", "mahalanobis", "cosine", "connection", "spearman", "cityblock", "hamming", "jaccard" [10].

(b) Group the objects into a pair, HC tree utilize some linkage criterias, for example, "Single linkage", "Complete linkage", "Average linkage", "Centroid technique", "Ward's strategy".

(c) Find the cutoff point in the various HC tree.

Make clusters by recognizing comparative group in the HC tree or by removing the tree at a self-assertive point or some threshold value. The time complexity nature of the HAC is $O(n^2)$ where n is the quantity of objects.

Though its complexity is high, we can use it for preprocessing work. As it does not require any input from user, we can obtain initial results from it. How to choose ideal number of groups at this point? A better solution in all, a superior arrangement is one in which each cluster is altogether different from different groups (between-group heterogeneity) and protests inside each cluster are as comparable as could be allowed (inside cluster homogeneity). To survey the homogeneity as well as heterogeneity of the clustering arrangement, different measures have been proposed. For a detailed discussion the interested reader is referred to "Sharma (1996)" and "Aldenderfer and Blashfield (1984)" [11, 12]. Larger

estimation of "cophenet relationship coefficient", "Root-Mean-Square Standard Deviation (RMSSTD)", "Semi-Partial R-Squared (SPR)" and "Centroid distance (CD)" represent homogeneity of the cluster solution whereas larger value of R-Square (RS) represents heterogeneity of the group arrangement. It gives high certainty when these measures recommend a similar number of clusters. Then again, if there is no agreement among the measures with respect to the ideal number of groups then it is judicious to inspect all the proposed arrangements and decide what number of clusters are fitting utilizing other criteria, for example, interpretability and value of the group arrangements.

A. Divide and conquer K means

In this novel approach, we present K-means divide and conquer clustering by selecting subspaces based on cluster index and perform clustering again on these subspaces. For bigger dataset, it is conceivable to partition the information into various subsets and apply chosen cluster calculation independently to these subsets. This methodology is known as divide and conquer [13,14]. The divide and conquer algorithm divides the whole informational collection into a subset dependent on a few criteria. The chosen subset is again clustered with a clustering algorithm K-Means. This nonhierarchical technique at first assigns the quantity of objects of the population equivalent to the final required number of clusters. The final computed number of clusters is picked with the end goal that the comparative gathering groups are most remote separated among others. It looks at every segment in the population and assigns it to one of the clusters relying upon the minimum separation.

Each time centroid's position is recomputed at whatever point a segment is added to the cluster and this procedure proceeds until the point when every one of the segments are assembled into the last required number of groups. The K-Means algorithm is exceptionally straightforward and can be effortlessly solve numerous practical issues. It can work exceptionally well for smaller and hyperspherical groups. The time complexity of K-Means is $O(TNKd)$ where T number of iteration, N number of samples, K number of clusters and d number of dimensions. Since K and d are almost constants and much less than N , K-means can be utilized to group vast informational collections. Parallel systems for K-Means are produced that can to a great extent accelerate the calculation [15], [16], [17]. The preferred standpoint of utilizing K-Means is to accelerate search and reduce complexity which directly relies upon number of samples.

For finding optimal clusters, take variation in number of clusters(NC) and apply K-Means. This may take too much time in guessing NC. So, there should be previous knowledge of data for deciding NC or either use other

clustering algorithm say HAC for previous knowledge of NC. Different validity indexes can be applied such as Dunn [18], Davies-Bouldin[19], Silhouette, Calinski-Harabasz and Sum-of-Squares [20] for finding optimal NC.

In any case, the K-Means algorithm is delicate to beginning choice of K and even in the best case, it can deliver just hyperspherical clusters. HC are more flexible. A conceivable answer for the issue of clustering substantial informational collections is the successive combination of techniques.

B. Sequential combination of methods for data clustering analysis for large data sets

The Sequential combination of strategies is an adaptable cluster analysis calculation intended to deal with extensive datasets. Conventional single clustering techniques are effective and exact on small datasets, yet for the most part don't scale up to the specific expansive datasets. In the underlying stage, apply a fast group strategy to the substantial dataset to compress the thick areas and shape sub-clusters. In the second stage, apply an exact group technique on the sub-clusters to locate the ideal number of groups. The records in a single sub-group ought to be in one of the final clusters so the pre-cluster step won't influence the exactness of the last clustering. As the quantity of sub-clusters from the pre-group step builds incorrectness will be diminished. Too many sub-groups will slow down the second stage clustering. Pick the number of sub-clusters with the goal that the number is sufficiently vast and little enough to not restrain execution in the later clustering technique.

As referenced above, many clustering algorithms necessitate that the number of clusters will be at first pre-set by the user. It is notable that this parameter influences the performance of the calculation altogether.

The proposed algorithm HDK which uses Hierarchical (for simplicity) and divide and conquer k means, is tried to optimize this problem by applying sequential method.

III. PROPOSED HDK ALGORITHM

CBIR using HDK algorithm for large image database involves the following steps.

- a) Pre-processing is based on RGB color Components of an image using HAC. Based on feature vector, we find optimal number of clusters which is an index for that image. Instead of vector only index is compared with every stored image in the database. After finding the matched index, the query image is contrasted with other images inside that group and the best matches are received by the user.
- b) Apply divide and conquer k means for further accurate retrieval.

Some hypothesis has been considered for proposed technique:

H1 - It would have the capacity to cluster samples having same no of groups and discover similarity among them.

H2 - It is quicker because of utilization of divide and conquer method.

H3 - It is more precise than a single step clustering.

H4 - It would enable Euclidean separation to be utilized in high dimensional information.

In this examination, we expect that the space has symmetrical dimensions for all objects. At last we utilize ordinal information type for application.

The principle point is to propose a framework combination of clustering and divide and conquer technique to defeat previously mentioned difficulties and enhance proficiency and accuracy. In past examination, entire space is divided into subspaces vertically dependent on object's features; we apply a strategy to partition whole space into subspaces horizontally dependent on comparable objects. Because of performing of vertical data reduction for extremely extensive and high dimensional data sets before applying the grouping systems results in dimension reduction and giving up the nature of results. Besides, dimension reduction is beyond the realm of imagination in a few applications e.g. individual closeness, client inclinations profiles grouping in client recommender framework or information produced by sensor systems. Existing HDK procedure would typically apply in an extensive space with high dimension; dividing enormous space into subspaces horizontally can lead us to high efficiency and precision. Analysis results show accuracy and speed up.

IV. EXPERIMENTAL RESULT AND PERFORMANCE

First we extract color feature of query image and image dataset considering dimensions of all images same. We apply single linkage to perform HAC. Utilize RGB space and choose similarity cut off point at 0.8 or 0.7, we cluster the image. Taking greatest estimation of cluster vector at cutoff point, we discover NC. We use here validity indices, for example, "cophenet connection coefficient", "RMSSTD" and "CD" for finding ideal number of clusters. Here, user doesn't require any knowledge of data and system becomes totally machine dependent. Using optimal NC, we divide space into subspace and apply K-Means clustering to query image and retrieved subspace.

The materials utilized in this work comprised of the following:

- a) Computer: Intel Atom CPU N450 speed of 1.66 GHz, 1.00 GB of RAM.
- b) Microsoft windows 7 professional
- c) MATLAB version 7.10.0

Test is performed on 10 image classes of corel image database, each containing 100 images of various categories. In preprocessing phase, suppose for query image id 777 optimal NC is found to be 111 and retrieved subspace which has same NC.

The poor decision of NC builds the time unpredictability in partitional calculations. By taking information of NC from HAC, subspace clustering is used. Taking variation of 5% in NC, we have applied K means algorithm on range of NC. By using validity indexes discussed in section 2, NC has found. Here we have used four validity indexes in which calinski Harabasz is not performed well. In the following tables, we have shown NC based on three validity indexes for query image 777 and its retrieved subspace. In Table I, higher silhouette value, constant minimum sum of squares and higher dunn is chosen for optimal NC. Some of the retrieved results for different query images are shown in Table II after performed HDK on 1000 image database.

The accuracy increases with similar matched images that are retrieved. When we performed HAC on 1000 and 10000 corel image dataset the results were not favourable. After applying Divide and conquer K Means accuracy is tremendously improved and achieved more than 90% accuracy for large dataset. Since time complexity of partitional algorithms increases with poor choice of NC, it is reduced by providing optimal NC using HAC.

TABLE I. Optimal NC of different images found by validity indexes after HDK for Query image 777 where NC is chosen in range (105-118)





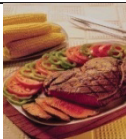











Image id	Silhouette	Sum of squares distance	Dunn
777	105	109	113
170	117	113	118
233	113	107	117
268	115	105	116
302	108	106	113
305	115	107	106
501	113	110	112
557	115	106	105
619	111	110	113
634	111	117	105
638	111	109	109
689	112	110	110
750	117	109	112
894	114	108	110

V. CONCLUSION

Using two steps clustering, HDK algorithm in high dimensional data sets with considering number of clusters(NC) helps us to improve accuracy and efficiency of Content-Based Image retrieval . Instead of

dividing whole space into subspaces vertically based on object's features we apply a horizontal method to divide entire space into subspaces based on similar objects. When subspace clustering is used based on NC, we achieve more accurate and efficient results. For this purpose we should consider orthogonal space which means that there should be no correlation among attributes of objects and dimension should be equal in all objects. The experiments on real world dataset illustrate that our approach is powerful and effective.

TABLE II. Retrieved results for different query images after performed HDK on 1000 imagedatabase

Query Image	Results			
 777	 750	 638	 777	
 941	 941	 115	 219	
 57	 57			
 556	 556			
 102	 102	 558	 592	

REFERENCES

- [1] V. Gudivada, and V. Raghavan, "Content-based image retrieval systems," IEEE Computer, vol. 28, no 9, pp18-22 (1995).
- [2] V.Castelli, and L. D.Bergman, (Eds.), Image Databases: Search and Retrieval of digital Imagery, Wiley, New York (2002).
- [3] M. J.Swain, and D. H Ballard, "Color indexing," International journal of Computer Vision ,vol 7,no. 1,pp. 11-32 (1991).
- [4] S. Mohan, T. Kankanalli., M. Mehtre, and J. K. Wut, "Cluster-based color matching," Pattern Recognition, Vol. 29, No. 4, pp.

- 701-708 (1996).
- [5] S. M. Holland , Cluster Analysis, Department of Geology, University of Georgia, Athens, GA 30602-2501.
 - [6] S.Guha, R. Rastogi, and K..Shim, "CURE: An efficient clustering algorithm for large databases," Proc. ACM SIGMOD Int. Conf. Management of Data, pp. 73–84 (1998).
 - [7] S.Guha, R. Rastogi, and K..Shim, "ROCK: A robust clustering algorithm for categorical attributes," Inf. Syst., vol. 25, no. 5, pp. 345– 366 (2000).
 - [8] G. Karypis, E.Han, and V.Kumar, "Chameleon: Hierarchical clustering using dynamic modeling," IEEE Computer, vol. 32, no. 8, pp. 68–75 (1999).
 - [9] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An efficient data clustering method for very large databases," Proc. ACM SIGMOD Conf. Management of Data, pp. 103–114(1996).
 - [10] A. Jain, and R. Dubes, Algorithms for Clustering Data, Englewood Cliffs, NJ: Prentice-Hall, (1988).
 - [11] S.Sharma, and A.Kumar, Cluster analysis and factor analysis, South Carolina: Sage publications, (1996).
 - [12] M.S. Aldenderfer, and R.K. Blashfield, Cluster analysis, California: Sage publications, (1984).
 - [13] X.Wang, and D.M.Wilkes, "A Divide-and-Conquer Approach for Minimum Spanning tree-Based Clustering," Member, IEEE Transactions on knowledge and data engineering, vol. 21 No7 (2009).
 - [14] M. Jain, and S.K. Singh, "A Survey On: Content Based Image Retrieval Systems Using Clustering Techniques for Large Datasets," International Journal of Managing Information Technology (IJMIT), vol. 3, no. 4, (2011) .
 - [15] E. Dahlhaus, "Parallel algorithms for hierarchical clustering and applications to split decomposition and parity graph recognition," Journal of Algorithms, vol. 36, no. 2, pp. 205–240, (2000).
 - [16] C.Olson, "Parallel algorithms for hierarchical clustering," Parallel Computer, vol. 21, pp. 1313–1325 (1995).
 - [17] K.Stoffel, and A.Belkoniene, "Parallel K-means clustering for large data sets," Proc. EuroPar'99 Parallel Processing, pp.1451–1454 (1999).
 - [18] J.C Dunn, "Well-Separated Clusters and Optimal Fuzzy Partitions," Journal of Cybernetics, vol. 4, no.1, pp. 95–104 (1974).
 - [19] D. Davies, and D.Bouldin, "A Cluster Separation Measure," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 1, no.2, pp. 224–227 (1979).
 - [20] Q. Zhao, M. Xu, and P. Franti, "Sum-of-Squares Based Cluster Validity Index and Significance Analysis," Proceedings of the International Conference on Adaptive and Natural Computing Algorithms (ICANNGA 2009), pp. 313–322 (2009).