High Dimensional Indexing in Image Databases with Adaptive Cluster Distance Bounding

Rammohan Reddy Dondeti¹, Ch. Ramesh Kumar²

¹Pursuing M.Tech (CS), Nalanda Institute of Engineering & Technology, Siddharth Nagar, Sattenapalli, Guntur., Affiliated to JNTUK, Kakinada, A.P., India.

²Asst. Professor, Department of Computer Science Engineering, Nalanda Institute of Engineering & Technology, Siddharth Nagar, Sattenapalli, Guntur., Affiliated to JNTUK, Kakinada, A.P., India.
rammohanreddy.51@gmail.com

Abstract - We consider approaches for exact similarity search in a high dimensional space of correlated features representing image datasets, based on principles of clustering and vector quantization. We develop an adaptive cluster distance bound based on separating hyperplanes, that complements our index in selectively retrieving clusters that contain data entries closest to the query. This bound enables efficient spatial filtering, with a relatively small pre-processing storage overhead and is applicable to Euclidean and Mahalanobis similarity measures.

Keywords — Cluster, Indexing, Image databases.

1. Introduction

With the proliferation of digital multi-media devices, such as digital cameras and video recorders, there has been an explosive growth in multi-media data and new applications that handle these data, such as image search engines, bio-medical imaging etc., hence necessitating efficient storage and data mining solutions. Searching and indexing image databases is a challenging task given the large number of elements to be handled and the high dimensionality of the search space. While searches based on keywords is the current paradigm in many search engines, keywords are not necessarily the most efficient representatives of multimedia information. For example, it would be ineffective to mine databases of medical images based on keywords or “metadata” if the goal is to discover hidden correlations that are unknown and hence have not been quantified through metadata. Clearly, content-based image search and retrieval (CBIR) would be the appropriate paradigm.

Images are represented by feature vectors and the measure of similarity between two images is assumed to be proportional to the distance between their feature vectors. Recently, a combination of texture features (extracted through Gabor filters) and color features (histograms) have been found to be efficient descriptors of the underlying images and form a part of the MPEG-7 multimedia standard (see [1]). Such feature vectors themselves are typically high-dimensional, such as the 60 dimensional texture descriptors [1] or the 256 dimensional color histograms of QBIC [2]. Similarity search is the search for elements in the database most similar to the query image. A popular query model is the k-nearest neighbor (kNN) query, where given a query image, the k most similar images are extracted from the database. Since, the feature vectors themselves are large in number and of high-dimensionality, it is more cost effective to store them on a hard-storage device, typically a hard disk. In the general database search literature, several index structures exist that facilitate search and retrieval of multi-dimensional data, such as the R-tree [3] and in low-dimensional spaces, these outperform sequential scan. But it has been observed that the performance of many multi-dimensional index structures degrades as the dimensions of the features increase and after a certain dimension threshold, they underperform sequential scan [4].
The time incurred in nearest neighbor search is largely dominated by IO time, which is determined by the number of sequential and random hard disk accesses. Irrespective of the access strategy, data are always stored and retrieved from the disk in units of disk blocks or pages. Random IOs would be faster in retrieving pages that are spaced far apart while less costly sequential access of pages would optimal if the required pages are spaced close together (even if not contiguously). However, due to the exponential growth of hyper volume with dimensionality ("the curse of dimensionality" [5]), a very large portion of the space is actually empty and hence, searching on naive index structures, leads to a large number of needless and costly random disk accesses, making it slower than the simple sequential scan.

A very popular and effective technique employed to overcome the curse of dimensionality is the Vector Approximation File (VA-File) [4]. In the VA-File, the space is partitioned into a number of hyper-rectangular cells, which approximate the data that reside inside the cells. The non-empty cell locations are encoded into bit strings and stored in a separate approximation file, on the hard-disk. In the search for the nearest neighbors, first, the vector approximation file is sequentially scanned and upper and lower bounds on the distance from the query vector to each cell are estimated. The bounds are used to prune the data-set of irrelevant vectors. The final set of candidate vectors are then read from the hard-disk and the exact nearest neighbors are determined. At this point, we note that the name “Vector Approximation” is somewhat misleading, since what is actually being performed is scalar quantization, where each component of the feature vector is separately and uniformly quantized (in contradistinction with vector quantization in the signal compression literature).

In this paper, we consider a clustering approach towards similarity search as an alternative to the Vector Approximation (VA) Files. The data set is clustered using a standard clustering or vector Quantization (VQ) technique, e.g., K-means or Lloyd’s algorithm and during query processing, load the "nearest" clusters into the main memory. We motivate such a solution since vector quantization, unlike the scalar quantization of the VA-File, can exploit dependencies across dimensions and hence, would be a more compact representation of the database. We propose to retrieve clusters till the kth nearest neighbor discovered so far is closer to the query than the remaining clusters, which guarantees that the k nearest neighbors have been discovered.

While such a vector quantization and clustering approach to search has been studied in the image database community (see [6, 7, 8]), the earlier approaches have focussed more on approximate nearest neighbor search. The distance bounds (based on bounding hyper spheres) derived in [7] are loose and hence the search strategy performs poorly when adapted to exact nearest neighbor search. We next present an effective cluster distance bound that complements our branch-and bound search algorithm.

2. Cluster Distance Bound

The Vector Approximation (VA)-file technique [4] implicitly assumes independence across dimensions, and that each component is uniformly distributed. This is an unrealistic assumption for real data-sets that typically exhibit significant correlations across dimensions and non-uniform distributions. To approach optimality, an indexing technique must take these properties into account. We resort to a Voronoi clustering framework as it can naturally exploit correlations across dimensions.
Moreover, we show how our clustering procedure can be combined with any other generic clustering method of choice requiring only one additional scan of the data-set. Lastly, we note that the sequential scan is in fact a special case of clustering based index i.e. with only one cluster. Crucial to the effectiveness of the clustering-based search strategy is efficient bounding of query-cluster distances. This is the mechanism that allows the elimination of irrelevant clusters.

Traditionally, this has been performed with bounding spheres and rectangles. However, hyperspheres and hyperrectangles are generally not optimal bounding surfaces for clusters in high dimensional spaces. In fact, this is a phenomenon observed in the SR-tree, where the authors have used a combination spheres and rectangles, to outperform indexes using only bounding spheres or bounding rectangles. The premise herein is that, at high dimensions, considerable improvement in efficiency can be achieved by relaxing restrictions on the regularity of bounding surfaces (i.e., spheres or rectangles). Specifically, by creating Voronoi clusters, with piecewise-linear boundaries, we allow for more general convex polygon structures that are able to efficiently bound the cluster surface. With the construction of Voronoi clusters under the Euclidean distance measure, this is possible. By projection onto these hyperplane boundaries and complementing with the cluster-hyperplane distance, we develop an appropriate lower bound on the distance of a query to a cluster.

While the Euclidean distance metric is popular within the multimedia indexing community [4], it is by no means the “correct” distance measure, in that it may be a poor approximation of user perceived similarities. The Mahalanobis distance measure has more degrees of freedom than the Euclidean distance [9] and by proper updation (or relevance feedback), has been found to be a much better estimator of user perceptions. We extend our distance bounding technique to the Mahalanobis distance metric, and note large gains over existing indexes.

3. The Hyperplane Bound

Let \( d(x, y) \) be a distance function that estimates the distance between vectors \( x \) and \( y \) in the feature space.

\[
d : \mathbb{R}^n \times \mathbb{R}^n \rightarrow [0, \infty)
\]

In subsequent discussion, we shall specialize to the Euclidean distance over (real vector spaces) as the feature similarity measure i.e. \( d(x, y) = ||x - y|| \). We define the distance from query \( q \) and a cluster \( X_m \) as

\[
d(q, X_m) = \min_{x \in X_m} d(q, x) \tag{2}
\]

The distance of vector \( q \) to a hyper plane \( H(n, p) = \{ y : y^T n + p = 0 \} \) is defined in the normal fashion as

\[
d(q, H) = \frac{|q^T n + p|}{||n||_2} \tag{3}
\]

Given a cluster \( X_m \), the query \( q \) and a hyperplane \( H \) that lies between the cluster and the query (a "separating hyperplane", see Figure 2), by simple geometry it is easy to see that for any \( x \in X_m \)

\[
d(q, x) \geq d(q, H) + d(x, H) \geq d(q, H) + \min_{x \in X_m} d(x, H) = d(q, H) + d(X_m, H)
\]

\[
\Rightarrow d(q, X_m) \geq d(q, H) + d(X_m, H) \tag{4}
\]

If \( H_{sep} \) represents a countably finite set of separating hyperplanes (that lie-between the query \( q \) and the cluster \( X_m \)),

\[
\Rightarrow d(q, X_m) \geq \max_{H \in H_{sep}} \{d(q, H) + d(X_m, H)\} \tag{5}
\]

The second lower bound presented in (5) can be used to tighten the lower bound on \( d(q, X_m) \). Next, we note that the boundaries between clusters generated by the K-means algorithm are linear hyperplanes. If \( c_1 \) and \( c_2 \) are centroids of two clusters \( X_1 \) and \( X_2 \), and \( Y_{12} \) the boundary between them, then \( \forall y \in Y_{12} \)

\[
d(c_1, y) = d(c_2, y)
\]

\[
\Rightarrow ||c_1||^2 - ||c_2||^2 - 2(c_1 - c_2)^T y = 0
\]
distances that need to be pre-calculated and stored, in addition to the cluster centroids themselves. The total storage for all clusters would be $O(M^2 + Md)$, where $d$ is the dimensionality. This heavy storage overhead makes the hyperplane bound, in this form, impractical for a large number of clusters. We can loosen the bound in (5) as follows:

$$d(q, \mathcal{X}_m) \geq \max_{H \in \mathcal{H}_{sep}} \{d(q, H) + d(H, \mathcal{X}_m)\}$$

$$\geq \max_{H \in \mathcal{H}_{sep}} d(q, H) + \min_{H \in \mathcal{H}_{sep}} d(H, \mathcal{X}_m)$$

This means that for every cluster $\mathcal{X}_m$ we would only need to store one distance term

$$d_m = \min_{1 \leq n \leq M, n \neq m} d(H_{mn}, \mathcal{X}_m)$$

4. Conclusion

We proposed an image database indexing technique for exact nearest neighbor search, based on the principles of clustering and vector quantization. The image feature vectors are clustered and during query processing, the nearest clusters are visited in order. We developed an adaptive cluster distance bound, based on separating hyperplanes, that complements our branch-and-bound search. Our index has low storage and computation costs and is able to provide significant reduction in random disk accesses over known methods.

References


AUTHORS PROFILE

Rammohan Reddy Dondeti, Pursuing M.Tech (CS) from Nalanda Institute of Engineering & Technology, Siddharth Nagar, Sattenapalli, Guntur, Affiliated to JNTUK, Kakinada, A.P., India. My research Interests are computer networks, Data Mining.

Ch. Ramesh Kumar, working as Asst. Professor, Department of Computer Science Engineering at Nalanda Institute of Engineering & Technology, Siddharth Nagar, Sattenapalli, Guntur, Affiliated to JNTUK, Kakinada, A.P., India. My research Interests are Mobile Computing, Data Mining.