RC 20704 (1/28/97)
Computer Science/Mathematics

IBM Research Report

RCSVD: Recursive Clustering with Singular Value Decomposition for Dimension Reduction in Content-based Retrieval of Large Image/Video Databases

Alex Thomasian, Vittorio Castelli, and Chung-Sheng Li

IBM Research Division
T.J. Watson Research Center
Yorktown Heights, New York

IBM Research Division
Almaden · T.J. Watson · Tokyo · Zurich

RCSVD: Recursive Clustering with Singular Value Decomposition for Dimension Reduction in Content-based Retrieval of Large Image/Video Databases

Alex Thomasian
athomas@watson.ibm.com
(914) 784-7299

V. Castelli
vittorio@watson.ibm.co
(914) 784-7946

Chung-Sheng Li
csli@watson.ibm.com
(914) 784-6661

Work funded in part by NASA/CAN grant no. NCC5-101

* Vittorio Castelli is the **primary contact.**

Abstract

Efficient indexing in feature space is crucial for many digital library applications. However, the efficiency of spatial indexing techniques usually deteriorates with the increase of dimensionality. A new algorithm, Recursive Clustering with Singular Value Decomposition (RCSVD) is proposed in this paper for reducing the dimensionality of the feature space. In the proposed algorithm, singular value decomposition and clustering techniques are applied recursively to the feature vectors until the dimensions cannot be further reduced. Performance of the proposed algorithm is evaluated based on a selected set of texture features extracted from satellite images. We report experimental results on the tradeoff between increased storage efficiency (due to reduced dimensionality) and reduced search efficiency to attain the same accuracy in the context of the ubiquitous nearest neighbor search operation. The results show that significant dimension reduction can be achieved by using the proposed algorithm without much impact on efficiency.

1 Introduction

Images and videos are captured by an increasing number of sources such as art galleries and museums, civilian and military satellites, biomedical imaging (such as CT, MRI, and PET), and home entertainment systems. As an example, the instruments on the first two Earth Observing System (EOS) platforms, to be launched in 1998 and 2000, will generate data at a rate of 281 GB/day. These raw data generated by various EOS platforms will be processed and stored in distributed active archive centers (DAACs) located throughout the United States [5]. The large volume of images and videos poses a significant challenge for data storage, data retrieval and data dissemination.

Conventional retrieval mechanism for images and videos from databases is through indexing appropriate metadata such as the title, the location, and the description of the image [32]. This mechanism is no longer adequate to manage massive databases of images and video. Recently, several image or video database systems allowing content-based queries have been developed. These systems index images on shape, color histogram, or texture. Application domains include photographic images[14, 11, 9, 8], medical images[15, 16], art work[10], and video clips [17, 18, 19, 20, 21]. Techniques for content-based event selection on satellite images have also been investigated [4, 12]. These techniques invariably require precomputing the image/video features (e.g. textures, color histogram or shape) to allow efficient indexing at the query time. The resulting dimensions of the feature vectors computed from the images can be potentially large. As an example, the dimensions of the feature vector computed from the local color histogram of an RGB image can easily exceed 64.

One of the key requirements in content-based retrieval from large image/video database consisting of millions of feature vectors is to efficiently index the feature vectors with high dimensionality. Several spatial indexing techniques such as R-trees [39] can be used for performing range and nearest neighbor queries. However, the efficiency of these techniques deteriorates rapidly as the number of dimensions of the feature space grows, since the search space becomes increasingly sparse (for a discussion of the topic, see for example [31][Chapter 1]). Thus, it is necessary to reduce the dimensionality of the search space to improve the efficiency of the existing spatial indexing algorithms.

We can categorize the approaches to dimension reduction in two main families: (1) *Techniques based on linear transformations*, such as the Karhunen-Loeve transform, the singular value decomposition (SVD) method, or the Principal Component Analysis (PCA). These techniques are shown to be optimal among all linear transformations for *concentrating* the information in fewer dimensions, for a given data distribution. Therefore, KL transformation and SVD have been widely used for dimension reduction and data compression [36]. (2) *Techniques based on nonlinear transformations*, such as vector quantization [37], the transform operated by a multilayer feedforward neural network (for an application to dimension reduction see [35],) and Kohonen self-organization map [38]. The focus of these studies is to extract feature vectors from the signals and images with maximal discriminating capabilities while maintaining minimal dimensions. A recent effort in addressing this issue uses two sets of features [40]. The first set, which consists of 120 features generated by the Gabor filter, is used for defining a similarity measure. Meanwhile, a much smaller second set, which consists of one or two features, is used for indexing. The feature for indexing is selected based on the target texture vector.

The focus of this paper, similar to [40], is to allow spatial indexing techniques to be performed efficiently. To achieve this goal, we propose a new technique, *Recursive Clustering Singular Value Decomposition (RCSVD)*, to reduce the dimension of the feature space in a recursive fashion. In RCSVD, SVD is applied to the feature space first to reduce its dimensionality. One of the clustering techniques, for instance LBG or K-means [43], is then applied to the reduced feature space to cluster the feature space into nonoverlapping regions based on the distribution of the feature vectors. SVD is then applied to each individual cluster to further reduce its dimensionality. This process can be recursively repeated until the dimensions cannot be further reduced. The effectiveness of the proposed scheme is measured in terms of the retrieval efficiency and accuracy of the nearest neighbor queries. The experiments are conducted on a selected set of features extracted from satellite images. The results show that, for a given set of images, significant dimension reduction can be achieved without significant impact on the efficiency and accuracy.

The rest of the paper is organized as follows: Section 2 contains preliminaries and notation. The proposed algorithm is outlined in Section 3. Section 4 describes the experimental results. The summary is given in Section 5.

2 Preliminaries

Content-based retrieval of image and video databases usually involves comparing a query object (also called *target* object), with the objects stored in the data repository. The search is usually based on a *similarity* comparison rather than on exact match, and the retrieved results are ranked according to a *similarity index*, e.g., a *metric*.

The objects of an image or video database can be defined and referred to at different abstraction levels, as described below:

1. *Raw Pixels*: At the lowest abstraction level, object are simply aggregations of raw pixels from the image. Comparison at the pixel level, which is also referred to as *template matching*, is very specific, and therefore is only used when a relatively precise match is required.

2. *Feature*: The next higher abstraction level for representing images is at the feature level. An image feature is a distinguishing primitive characteristics or attribute of an image [28]. Some features such as luminance, shape descriptor, and gray scale texture are natural as they correspond to visual appearance of an image. Other features such as amplitude histogram, color histogram, and spatial frequency spectra are artificial as they are usually obtained from specific manipulations of an image. Each image in an image archive can be segmented by using a set of n features, which are grouped into a feature vector, into regions consisting of homogeneous feature vectors. Similarity search in the n-dimensional feature space thus consists of comparing the target feature vector with the feature vectors stored in the database.

3. *Semantic*: This is the highest abstraction level at which a content-based search can be performed. Semantic information from an image is usually

extracted from a pre-trained classifier or supplied through human interpretation. For satellite images, this information could include the type of land cover of a specific area such as water, forest, or urban.

In this paper, we focus on the retrieval at the feature level. The most commonly used similarity metric when objects are represented as n-dimensional feature vectors $\mathbf{u}$ and $\mathbf{v}$, where $\mathbf{u} = [u_1, ..., u_n]^T$ and $\mathbf{v} = [v_1, ..., v_n]^T$, is the Euclidean distance between these vectors:

$$D(\mathbf{u}, \mathbf{v}) = \Big[(\mathbf{u} - \mathbf{v})^{\mathbf{T}}(\mathbf{u} - \mathbf{v})\Big]^{1/2} = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2} \tag{1}$$

or, in general, the L^p distance metric which is defined as

$$\begin{aligned} D_p(\mathbf{u}, \mathbf{v}) &= \left(\sum_{i=1}^{n} |u_i - v_i|^p\right)^{1/p}, \quad \forall p \in [1, \infty), \\ &= \max_i |u_i - v_i|, \qquad \text{for } p = \infty. \end{aligned}$$

The Euclidian distance is solely used in this study.

3 The RCSVD Algorithm

Clustering of feature vectors provides the opportunity for higher dimesionality reduction than is possible when all vectors are considered. This is because different clusters may appear "flat", while this is not so for all data points. As an example, Fig. 1 shows three clusters in the 3-dimensional feature space. In this figure, cluster 1 is on the y-z plane, cluster 2 is on the x-z plane, while cluster 3 is on the x-y plane. Apparently, the minimum number of dimensions in order to characterize the entire data set is at least 3. However, when the feature space is clustered and segmented into three nonoverlapped subspaces, additional dimension reduction opportunity is possible within each cluster. For the case of Fig. 1, each cluster can be represented by two dimensions.

Translation into uncorrelated axes, without even reducing the dimensionmality, is beneficial from the viewpoint of attaining "better" clusters. Applying clustering first is of course an option, which may not be desirable because of the high cost of clustering methods and that the cost increases with dimensionality. If we assume that there is a total of m feature vectors and the dimension of each feature vector is n, the cost of computing the covariance matrix is of the

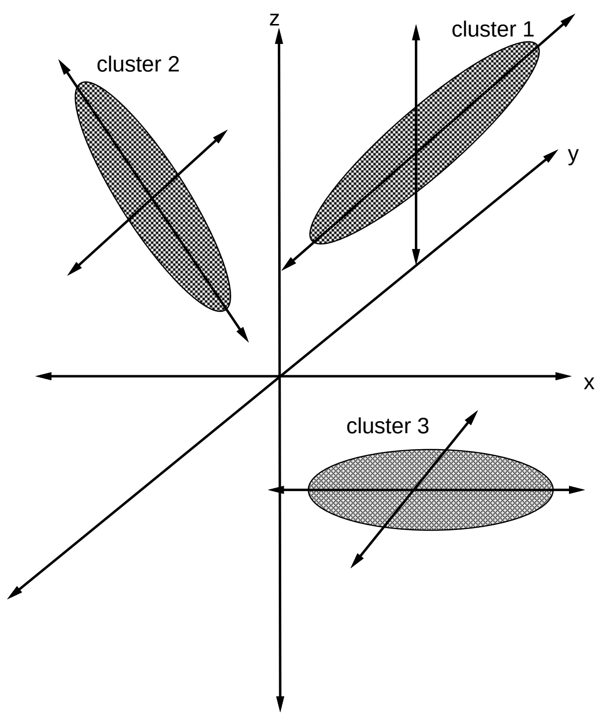

Figure 1: Intuition for dimension reduction by clustering.

order of $O(mn^2)$ multiplications. Additional computation cost is also required to find the eigenvalues of the covaraince matrix.

The SVD followed by clustering paradigm can be applied recursively, resulting in further reduction in dimensionality. A point of dimishing returns is reached eventually, as the dimensionality cannot be reduced any further.

The proposed RCSVD algorithm consists of the following steps:

1. Consider matrix $\mathbf{A}$ with m rows consisting of feature vectors with n dimesions. Normalize (studentize) the feature vectors $\mathbf{u}$ by computing $(u_i - \overline{u_i})/\sigma_{u_i}, 1 \leq i \leq n$, where $\overline{u_i}$ is the sample mean, and $\sigma_{u_i}^2$ is the sample variance of the i^{th} dimension.

2. Obtain the *covariance* matrix $\mathbf{C}$ by multiplying A by its transpose and divide the entries of the resulting matrix by m. The trace of this matrix is the sum of the variances or the total "energy", which remains invariant under rotation.

3. Apply principal component analysis or to C yielding a new covariance matrix $\mathbf{C}'$ which is diagonal, i.e., the resulting features are uncorrelated in the new axes. Since $\mathbf{C}$ is symmetrix; its n eigenvalues are nonnegative $\lambda_0 \geq \lambda_1 \geq ... \geq \lambda_n \geq 0$. Equivalently, SVD or the Karhunen Loeve or Hetelling transforms could be applied to $\mathbf{A}$ yielding he *singular values,* which are the square root of the eigenvalues $\mathbf{C}$ (without division by m). Both approaches provide eignenvectors to realign the feature vectors such that the resulting dimensions are uncorrelated.

4. Use the eignenvectors to transform coordinates. Let Λ denote the sum of the eigenvalues or the trace of the covarinace matrices $\mathbf{C}$ and $\mathbf{C}'$. Let T_0 denote the fraction of energy to be reserved. Sort the nonnegtaive eignevalues according to their magnitude and select $J_0 \leq n$ eigenavlues such that
$$\sum_{i=0}^{J_0-1} \lambda_i < T_0, \qquad \sum_{i=0}^{J_0} \lambda_i \geq T_0. \tag{2}$$
Retain J_0 dimensions cooresponding to the eigenvalues in matix $\mathbf{A}'$.

5. Cluster the vectors $\mathbf{u}'$ in $\mathbf{A}'$ into p clusters. Apply PCA to each individual cluster using a different T'_0 as in Step 4. Reduce the number of dimensions.

Additional computational complexity is incurred for multi-level indexing structure, as exemplified in the case of a nearest neighbor query. Given a target vector (with n dimensions) obtain its N nearest neighbors in the Euclidian space. We need to transform the axes of the original target vector using the eigenvectors at level one and then select an appropriate subset of its dimesnions. We next determine the cluster to which the target vector belongs and proceed to apply the appropriate rotation and dimensionality reduction again according to the metadata for that cluster.

Furthermore, the nearest neigbors to a point may reside in other clusters as well as the target cluster, which is usually the cluster with the closest centroid to the target vector. To determine which of the clusters, in addition to the target cluster, are candidates for nearest neighbor search, we consider the hypersphere centered on the taget point which encompasses N nearest neighbors located in that cluster. Other clsuters need to be coinsidered if this hyperspere intersects with them. A search for nearest neighbors in the second cluster may replace some of the points from the previous search. This step is repeated after computing the radius of the new hypershere, until there are no intersections. The distances among various clusters is recomputed to determine the ordering in which other clusters need to be considered, if at all.

4 Performance Study

As noted in the previous section, the nearest neighbor search is used in this paper to assess the effect of RCSVD on retrieval performance. While we have evaluated the efficiency of nearest neighbor search using an indexing structure due to Park and Kim [44], we are also experimenting with R-tree and other spatial indexing techniques. Nevertheless, this paper is concerned with the inherent efficiency of SVD and clustering methods.

As far as the reduction in indexing space is concerned, we consider a measure which is independent of the implementation of a specific indexing technique. This measure, *volume* (denoted by V), is defined as the total amount of space in order to store the indexing information. The volume of the original feature space, V_0, thus equals mn. The volume of the clustered feature space, is

$$V_1 = \sum_{i=1}^{K} m_i n_i \tag{3}$$

where K is the total number of clusters, m_i is the number of feature vectors in cluster i, and n_i is the dimension in cluster i. The reduction in *volume* by applying the algorithm described in the previous section can then be defined as $1 - V_1/V_0$. The space required for eigenvectors and selecting appropriate features at each level tends to be small and is ignored in this study.

The reduction in the number of dimensions results in a perforamnce degradation, which is defined below. Let D be the set of the feature vectors in the database. Let A be the set of the k nearest neighbors to a vector v. Let the query ask for a set containing the k nearest neighbors to v, and call B the result of the query. In general $|B| \geq |A|$, where $|\cdot|$ denotes the number of elements of the set. Let $C = A \cap B$ be the subset of the k nearest neighbors actually retrieved by the query. It is clear that, for a fixed template vector v, the size of C is a non-decreasing function of the size of A.

Two metrics are used to measure the performance of the proposed algorithm:

- *Retrieval Efficiency,* R_E, which is defined as the ratio between the total number of feature vectors retrieved that are correct, $|C|$ and the total number of feature vectors retrieved,$|B|$:

$$R_E = \frac{|C|}{|B|} \tag{4}$$

- *Retrieval Accuracy,* R_A, which is defined as the ratio between the total number of feature vectors retrieved that are correct, $|C|$, and the total number of feature vectors that are supposed to be retrieved, $|A|$:

$$R_A = \frac{|C|}{|A|} \tag{5}$$

Clearly, as A approaches D, so do B and C. Thus, R_A approaches unity while R_E approaches its lower bound $|C| \,/\, |D|$.

In the special case of nearest neighbor search, we were concerned with finding the N_t nearest neighbors (according to Euclidian distance) to a feature point. Since a search for N_t vectors based on the transformed feature space with reduced dimension will yield less than desired accuracy, a larger number of points, N_r, needs to be considered. The experiments are carried out for S randomly selected vectors. During the i^{th} experiment, a total of $N_{r,i}$ vectors are retrieved in order to locate the $N_{t,i}$ nearest neighbors. Assuming that there are a total of K agreements between the retrieved vector list (of size $\sum_{i=1}^{S} N_{r,i}$) and the target vector list (of size $\sum_{i=1}^{S} N_{t,i}$), the efficiency of the retrieval is $K/\sum_{i=1}^{S} N_{r,i}$ while the accuracy of the retrieval is $K/\sum_{i=1}^{S} N_{t,i}$.

The goal of any retrieval algorithms is to maximize the efficiency and accuracy simultaneously. However, this is often impossible as they do usually have conflicting requirements.

Nine texture features from two different sets of images are extracted for this experiment. The first set consists of 204,655 feature vectors from 55 synthetic

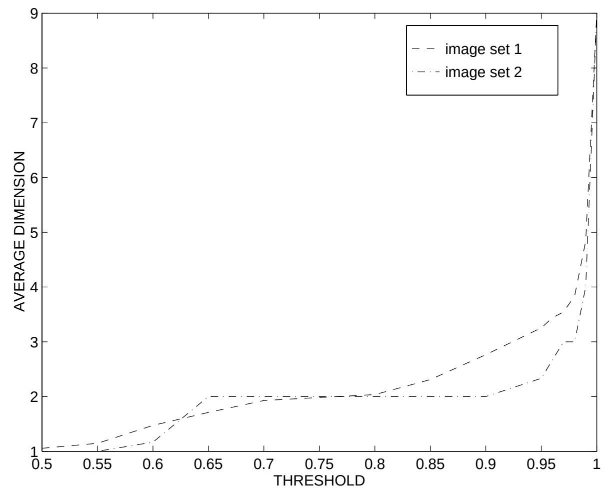

Figure 2: Effective dimension as a function of the cutoff threshold.

aperture radar (SAR) images on Alaska. The second set consists of 1.5 million feature vectors from 6 LANDSAT Multi-Spectral Scanner (MSS) images. The texture features used in this study include fractal dimension, coarseness, entropy, circular moran autocorrelation function, and several spatial grey-level difference (SGLD) statistics (described in detail in [45]).

Figure 2 shows the effective number of dimensions retained as a function of the energy cutoff threshold used in the SVD. In this figure, the number of dimensions for both image sets grows slowly as the energy threshold increases. In fact, three dimensions are adequate for retaining more than 95% of the energy.

Although the retrieval accuracy monotonically increases with the cutoff threshold, it is not possible to infer the accuracy nor efficiency from the cutoff threshold directly. Figure 3 and 4 shows the retrieval efficiency for the first and the second feature vector set, respectively, as a function of the number of dimensions retained for a given minimal target accuracy. For the first image set, the retrieval efficiency is independent of the target accuracy as the retrieval accuracy already exceeds the target accuracy when the number of dimensions is greater than 5. When the number of dimensions is less than 5, the retrieval efficiency drop faster with the decrease of the dimension for higher target accuracy. Apparently, less information is retained for fewer dimensions, and thus more candidates are required to be explored in order to achieve the same accuracy. The retrieval efficiency for the second image set demonstrates similar

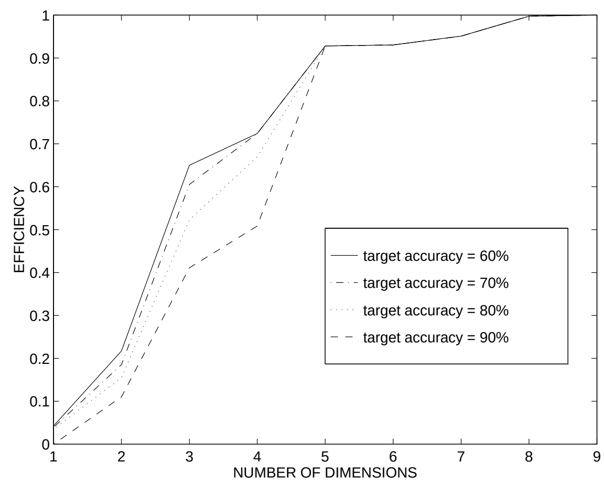

Figure 3: Efficiency as a function of the number of dimensions for image set 1.

trend.

5 Discussion And Summary

In this paper, a new technique, *Recursive Clustering Singular Value Decomposition (RCSVD)*, is proposed to reduce the dimension of the feature space. In this method, SVD is applied to the feature space first to reduce its dimensionality. One of the clustering techniques such as CART or neural networks are then applied to the reduced feature space to cluster the feature space into overlapped or nonoverlapped regions based on the distribution of the feature vector. SVD is then applied to each individual cluster to further reduce the dimensionality of each cluster. This process can be applied recursively until the dimensions cannot be further reduced. Spatial indexing techniques can then be applied to the clustering results. The performance of spatial indexing working in conjunction with RCSVD, however, remains to be investigated.

The effectiveness of the proposed scheme is measured in terms of the retrieval efficiency and accuracy of nearest neighbor queries. The experiment is conducted on a selected set of features extracted from the satellite images. The results show that significant dimension reduction (more than 200%) can be achieved without much impact on the efficiency and accuracy.

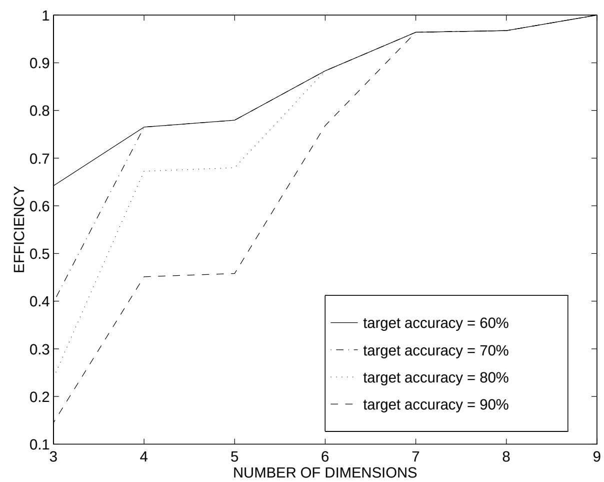

Figure 4: Efficiency as a function of the number of dimensions for image set 2.

6 Acknowledgements

The authors wish to acknowledge The authors are also grateful to the NOAA NALC project for providing us with relevant satellite imagery.

References

[1] M. Stonebraker, "Sequoia 2000: A Reflection on the First Three Years," IEEE Computer Science Engineering, vol. 1, no. 4, Winter 1994, pp. 63-72.

[2] D. DeWitt, N. Kabra, J. Luo, J. Naughton, J. Patel, and J.-B. Yu, "Paradise - A Parallel Geographic Information System," NRA/CAN Workshop, Maryland, 1995.

[3] E. Mesrobian, R. R. Muntz, J. R. Santos, E. C. Shek, C. R. Mechoso, J. D. Farrara, and P. Stolorz, "Extracting Spatio Temporal Patterns from Geoscience Dataset," Proc. IEEE Workshop on Visualization and Machine Vision, 1994, pp. 92-103.

[4] J. J. Turek et al, "Searching in Large Image Archives," submitted to special issue on Digital Library, IEEE Computer Magazine.

[5] J. C. Tilton and M. Manohar, "Earth Science Data Compression Issues and Activities," Remote Sensing Reviews, vol. 9, 1994, pp. 271-298.

[6] N. M. Short, Jr., R. F. Cromp, W. J. Campbell, J. C. Tilton, J. L. LeMoigne, G. Fekete, N. S. Netanyahu, G. Sylvain, "AI Challenges within NASA's Mission to Planet Earth," 1994 National Conference on Artificial Intelligence, Seattle, 1994.

[7] J. LeMoigne, "Parallel Registration of Multi-sensor Remotely Sensed Imagery Using Wavelet Coefficients,"

[8] J. R. Smith and S.-F. Chang, "VisualSeek: A fully automated content-based image query system," Proc. International Conference on Image Processing, Lausanne, Switzerland, 1996.

[9] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, and R. Jain, "The Virage Image Search Engine: An Open framework for Image Image Management," Proc. SPIE, Storage and Retrieval for Still Image and Video Databases, vol. 2670, pp. 76-87, 1996.

[10] B. Holt and L. Hartwick, "Visual Image Retrieval for Applications in Art and Art History" Proc. SPIE, vol. 2185, Storage and Retrieval for Image and Video Databases, Feb. 1994, pp.70-81.

[11] A. Pentland, R. W. Picard, S. Sclaroff, "Photobook: Tools for Content-Based Manipulation of Image Databases," Proc. SPIE, vol. 2185, Storage and Retrieval for Image and Video Databases, Feb. 1994, pp.34-47.

[12] R. Samadani, C. Han, and L. K. Katragadda, "Content-based Event Selection from Satellite Images of the Aurora," Proc. SPIE, vol. 1908, Storage and Retrieval for Image and Video Databases, Feb. 1993, pp. 50-59.

[13] R. Shann, D. Davis, J. Oakley, F. White, "Detection and Characterization of Carboniferous Foraminifera for Content Based Retrieval From an Image Database," Proc. SPIE, vol. 1908, Storage and Retrieval for Image and Video Database, 1993, pp. 188-197.

[14] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, G. Taubin, "The QBIC Project: Querying Images by Content Using Color Texture, and Shape," Proc. SPIE, vol. 1908, Storage Retrieval for Image and Video Databases, 1993, pp. 173-187.

[15] T. Y. Hou, P. Liu, A. Hsu, M. Y. Chiu, "Medical Image Retrieval by Spatial Feature," Proc. IEEE International Conference on System, Man, and Cybernetics, 1992, pp. 1364-1369.

[16] T. Y. Hou, A. Hsu, P. Liu, M. Y. Chiu, "A Content Based Indexing Technique using Relative Geometry Features," Proc. SPIE, vol. 1662, Image Storage and Retrieval Systems, 1992, pp. 29-68.

[17] E. Deardorff, T. D. C. Little, J. D. Marshall, and D. Venkatesh, “Video Scene Decomposition with the Motion Picture Parser,” Proc. SPIE, vol. 2187, Digital Video Compression on Personal Computers: Algorithms and Technologies, Feb. 1994, pp.44-55.

[18] H. Zhang and S. W. Smoliar, “Developing Power Tools for Video Indexing and Retrieval,” Proc. SPIE, vol. 2185, Storage and Retrieval for Image and Video Databases, Feb. 1994, pp. 140-149.

[19] S. W. Smoliar and H. Zhang, “Content based Video Indexing and Retrieval,” IEEE Multimedia, vol. 1, no. 2, Summar, 1994, pp. 62-72.

[20] F. Arman, A. Hsu, M. Y. Chiu, “Image Processing on Compressed Data for Large Video Database,” Proc. ACM Multimedia 93, pp. 267-272.

[21] F. Arman, A. Hsu, M. Y. Chiu, “Feature Management for Large Video Database,” Proc. SPIE, vol. 1908, 1993, pp. 2-12.

[22] N. Dal Degan, R. Lncini, P. Migliorati, and S. Pozzi, “Still Images Retrieval from a Remote Database: The System Imagine,” Signal Processing: Image Communications, vol. 5, 1993, pp. 219-234.

[23] S. K. Chang, C. W. Yan, D. C. Dimitroff and T. Arndt, “An Intelligent Image Database System,” IEEE Transactions on Software Engineering, vol. 14, no. 5, May 1988.

[24] B. Chiptrasert and K. R. Rao, “Discrete Cosine Transform Filtering,” Signal Processing, vol. 19, no. 3, pp. 233-245, March, 1990.

[25] J. B. Lee and B. G. Lee, “Transform Domain Filtering Based on Pipelining Structure,” IEEE Transaction on Signal Processing, vol. 40, no. 8, Aug. 1992, 2061-2064.

[26] S. F. Chang, “Compositing and manipulation of video signals for multimedia network video services,” Ph.D. Dissertation, U. C. Berkeley, Aug. 1993.

[27] B. C. Smith “Implementation Techniques for Continuous Media Systems and Applications,” Ph.D. Dissertation, U. C. Berkeley, Aug. 1994.

[28] W. K. Pratt, “Digital Image Processing,” Wiley, New York, 1991.

[29] P. P. Vaidyanathan, “Orthonormal and Biorthonormal Filter Banks as Convolvers, and Convolutional Coding Gain,” IEEE Transactions on Signal Processing, vol. 41, no. 6, June 1993.

[30] H. S. Stone and C.-S. Li, “Image Matching by Means of Intensity and Texture Matching in the Fourier Domain,” Storage and Retrieval for Still Image and Video Databases, Proc. SPIE vol. 2670, 1996.

[31] V Cherkassky, J.H. Friedman and H. Wechsler, "From Statistics to Neural Networks," NATO ASI Series, vol 136, Springer-Verlag, 1994.

[32] Special issue on "Finding the Right Image - Content-based Image Retrieval Systems," IEEE Computer Magazine, Sept. 1995.

[33] C.-S. Li, J. J. Turek, and E. Feig, "Progressive Template Matching for Content-Based Retrieval in Earth Observing Satellite Image Database," Proc. SPIE, vol. 2606, Oct., 1995.

[34] R. Y. Wong, E. Hall, and J. Rouge, "Hierarchical Search for Image Matching," Proc. IEEE Conf. Decision Control, Clearwater, Fl., December 1976.

[35] Y. H. Pao, "Dimension Reduction, Feature Extraction and Interpretation of Data with Network Computing," International Journal of Pattern Recognition and Artificial Intelligence, vol. 10, no. 5, Aug. 1996, pp. 521-535.

[36] F. Hermansen and A. A. Lammertsma, "Linear Dimension Reduction of Sequences of Medical Images," Phys. Medical Biology, vol. 41, no. 8, Aug. 1996, pp.1469-1481.

[37] Y. C. Lin and S. C. Tai, "Fast Feature Based Vector Quantization Algorithm of Image Coding," Optical Engineering, vol. 34, no. 10, Oct. 1995.

[38] J. Joutsensalo and A. Miettinen, "Self Organization Operator Map for Nonlinear Dimension Reduction," Proc. IEEE ICNN, pp. 111-114, 1995.

[39] A. Guttman, "R-trees: A dynamic index structure for spatial searching," *Proc. ACM SIGMOD Conf. on Management of Data,* Boston, MA, June 1994, pp. 47-57.

[40] A. D. Alexandrov, W. Y. Ma, A. El Abbadi, and B. S. Manjunath. "Adaptive filtering and indexing for image databses," *Proc. SPIE Vol 2420: Stirage and Retrieval for Image and Video Databases III,* Feb. 1995, pp. 12-23.

[41] M. Turk and A. Pentland, "Egenfaces for recognition," *J. Cognitive Neuroscience 3,*1 (1991), 71-86.

[42] R. Ng and A. Sedighian. "Evaluating multi-dimensional indexing structures for images transformed by principal component analysis" *Proc. SPIE Vol. 2670: Storage andRetrieval for Still Image and Video Dtabases IV,* San Jose, CA, Feb. 1996, pp. 50-61.

[43] B. Everitt *Cluster Analysis, 2nd Edition*, Halsted Press, 1980

[44] B. S. Kim and S. B. Park "A fast k nearest neighbor finding algorithm based on the ordered partition," *IEEE Trans. on Pattern Analysis and Machine Intelligence PAMI-8,*6 (Nov. 1986), 761-766.

[45] C.-S. Li and M.-S. Chen, "Progressive Texture Matching for Earth Observing Satellite Image Database," *SPIE Multimedia Storage and Archiving Systems*, Boston, pp. 150-161, Nov. 1996.