An Improved Incremental Singular Value Decomposition

[1] Rong Hu, [2]Weihong Xu,[3]Fangjun Kuang
[1,] School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing, China , E-mail:hremail000@qq.com
**1, Changsha Aeronautical Vocational and Technical College. Changsha, China,E-mail:410907140@qq.com*
2 ,College of Computer and Communications Engineering. Changsha University of Science&Technology
3,Department of Computer Science and Technology, Hunan Vocational Institute of Safety & T echnology, Changsha , China
3,School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing, China

Abstract

This paper proposes an incremental singular value decomposition (SVD) algorithm. This method was used to handle dynamic database, where new date arrive once the model is built. First a suitably sized SVD using batch method is computed. Then as new data arrive the singular values of a sequence of samples incrementally are computed. The algorithm has minimal memory requirements, and therefore interesting in Face Recognition domain, where very large datasets are often used, and datasets quickly become in tractable. The technique is demonstrated on the task of classifier for Face Recognition.

Keywords*: Inremental Singular Value Decomposition , Real-Time Application , Face Recognition*

1. Introduction

Feature extraction and classification of images in high-dimensional feature space has been an active research area n computer vision and pattern recognition communities [1-4,18,19]. Since the original input image space has a very high dimensionality, a dimensionality reduction technique is usually employed before classification takes place. Principal component analysis (PCA) [2–4] is one of the most popular representation methods for face recognition. But it is computationally inefficient to directly compute the covariance matrix of samples in this high dimensional space. This is due to the high dimension of this space while the number of samples is typically smaller than the dimension. Furthermore, computing the eigenvalues and eigenvectors of a small sample set is even degenerate. The Singular Value Decomposition (SVD) is a well-established method to extract features from images. SVD is a basic and useful mathematical tool in face recognition community. Muller et al. [5] describes how SVD is applied to problems involving image processing, in particular, how SVD aids the calculation of so-called eigenfaces, which provide an efficient representation of images for face recognition. In [6], a detailed survey of SVD under the uniform framework of matrix decomposition is presented, which includes theoretical analysis and various applications in face recognition and gene expression data communities. However, computing SVD from a large data set in high dimensional space using a batch method requires that the complete set of data is available. The algorithm is of a high order complexity and it requires a large storage space. To address these problems, incremental methods are desirable.

There have been published methods for updating SVD[15-17], either dropping some data items or adding some data items, but the computation of SVD for the updated matrices is still in a batch fashion.[7-9].

This paper we introduce an algorithm. It was used to handle dynamic database, where new date arrive once the model is built. We first compute a suitably sized SVD used batch method. Then we developed to compute the singular values of a sequence of samples incrementally as new data arrives(so our new way called Batch incremental Singular Value Decomposition BISVD). Our incremental way has the advantages of low time complexity, high speed in computation, smaller storage space, and a potential for real time applications.

The rest of this paper is organized as follows. The next Section, we gives a brief overview of SVD , section 3 outlines our incremental BISVD algorithm. In section 4, we report our experimental procedure, results and discussion. Finally in Section 5, we draw the conclusion and point out some directions for future research.

2. An Overview of SVD

As a powerful approach in linear subspace analysis, SVD is a basic and useful mathematical tool in face recognition community. Muller et al. [14] describes how SVD is applied to problems involving image processing, in particular, how SVD aids to calculate so-called eigenfaces, which provide an efficient representation of images for face recognition. In [15], a detailed survey of SVD under the uniform framework of matrix decomposition is presented, which includes theoretical analysis and various applications in face recognition.

The singular value decomposition of a rectangular data matrix A , can be presented as

$$A = U \Sigma V^T \qquad (1)$$

Where U and V are matrices of orthogonal left and right singular vectors respectively, and Σ is a diagonal matrix of the corresponding singular values.

Calculating the SVD consists of finding the eigenvalues and eigenvectors of AA^T and $A^T A$. The eigenvectors of $A^T A$ make up the columns of V, the eigenvectors of AA^T make up the columns of U. Also, the singular values in Σ are square roots of eigenvalues from AA^T or $A^T A$. The singular values are the diagonal entries of the Σ matrix and are arranged in descending order. The singular values are always real numbers. If the matrix A is a real matrix, then U and V are also real.

One important result of the SVD of A is that

$$A^{(l)} = \sum_{k=1}^{l} u_k s_k v_k^T \qquad (2)$$

is the closest rank-l matrix to A. The term "closest" means that $A^{(l)}$ minimizes the sum of the squares of the difference of the elements of A and $A^{(l)}$, $\sum_{ij}|a_{ij} - a^{(l)}_{ij}|^2$.

To give a vivid visualization, Suppose A, B is the image of same size , is decomposed by SVD respectively as follows

$$A = \sum_{1}^{k} u_k s_k v_k \qquad B = \sum_{1}^{k} u_k^{'} s_k^{'} v_k^{'} \qquad (3)$$

We exchanged $u_1, u_2, \cdots, u_k$ and $u_1^{'}, u_2^{'}, \cdots, u_k^{1}$, and exchanged $v_1 v_2 \cdots v_k$ and $v_1^{'}, v_2^{'}, \cdots, v_k^{'}$. Then , we reconstruct the image according to formula (2) . The result is as shown in fig. 1. We can see a large amount of information is presented in the closest rank-l of the vector of singular value .

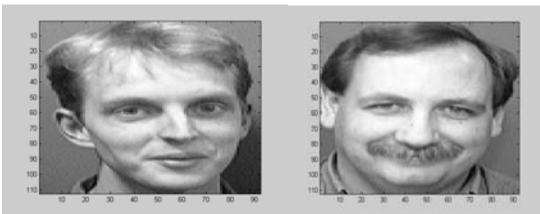

Fig 1 (a): The original image A, B

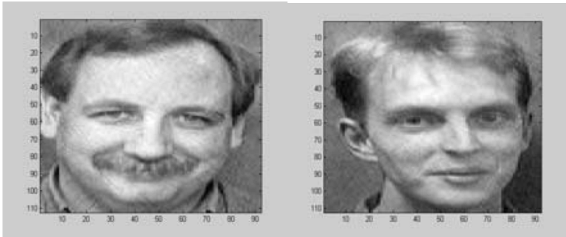

Fig 1(b): The reconstructive image A,B after exchange some columns

Unfortunately, computing an SVD of a very large dataset is an impractical affair, requiring complete data, run-time quadratic in the dataset size, and in-memory storage of entire dataset. Our focus is to develop algorithm that allows the model to be incrementally computed. We first use bath method to compute a suitably sized model and then use the incremental method to build incrementally upon that.

3. Incremental SVD Algorithms

Oja and Karhunen [11] demonstrated an incremental solution to find the first eigenvector from data arriving in the form of serial data items presented as vectors. Sanger [12] later generalized this for finding the first N eigenvectors with the Generalized Hebbian Algorithm. The algorithm converges on the exact eigen decomposition of the data with a probability of one. Sanger's formulation is :

$$c_{ij}(t+1) = c_{ij}(t) + \gamma(y_i(t)x_j(t) - y_i(t)\sum_{k \le i} c_{kj}(t)y_k(t)) \qquad (4)$$

In the above, c_{ij} is an individual element in the current eigenvector, x_j is the input vector and y_i is the activation i.e. $c_i \cdot x_j$, the dot product of the input vector with the i th eigenvector. γ is the learning rate.

Weng et al.[13] demonstrate the efficacy of this approach and extended it as follow:

$$v(n) = \frac{n-1-l}{n} n(n-1) + \frac{1+l}{n} u(n)u(n)^T \frac{v(n-1)}{\|v(n-1\|} \qquad (5)$$

Where $v(n)$ is the n th step estimate of v .where the positive parameter l is called the amnesic parameter. They reported that the amnesic parameter can further improve the convergence rate.

We extend this idea to build a system where we first compute a suitably sized model use batch method and then build incrementally upon that as new data arrives.

3.1 The Algorithm.

First we must make clear the difference between batch and incremental method for singular value decomposition. A bath method computes an Eigen space using all observations simultaneously. An incremental method computes an Eigen space model by successively updating an earlier model as new observations become available. Out operators is that we first compute a suitably sized model use batch method. Then we incrementally update that model as new observations become available.

Suppose we get k singular values use batch method:

$$A_n^{\ l} = \sum_{k=1}^{l} u_k s_k v_k^T \tag{6}$$

Suppose that new sample vectors are acquired sequentially, $a(n+1), a(n+2), \cdots$ possibly infinite. Each $a(n)$, is a d-dimensional vector and d can be as large as 5,000 and beyond. we can assume that a(n) has a zero mean (the mean may be incrementally estimated and subtracted out) . $M = AA^T = E\{a(n)a^T(n)\}$ is the d ×d covariance matrix , which is neither known nor allowed to be estimated as an intermediate result .

By definition, an eigenvector x of matrix M satisfies

$$\lambda x = M x \tag{7}$$

Where λ is the corresponding eigenvalue. By replacing the unknown M with the sample covariance matrix and replacing the x of (7) with its estimate $x(i)$ at each time step i, we obtain an illuminating expression for $v = \lambda x$:

$$v(n) = \frac{1}{n} \sum_{i=1}^{n} a(i) a^T(i) x(i) \tag{8}$$

Where $v(n)$ is the nth step estimate of v. Once we have the estimate of v, it's easy to get the eigenvector and the eigenvalue since $\lambda = \|v\|$ and $x = v / \|v\|$.

Now, the problem is how to estimate $x(i)$ in (8). Considering $x = v/\|v\|$, we may choose $x(i)$ as $v(i-1)/\|v(i-1)\|$, which leads to the following incremental expression :

$$v(n) = \frac{n-1}{n} v(n-1) + \frac{1}{n} a(n) a^T(n) \frac{v(n-1)}{\|v(n-1)\|}, \tag{9}$$

To begin with, we set $v(n) = oldv$, where $oldv$ is computed by a suitably sized batch SVD. Thus for incremental estimation as new date arrives, (9) is written in a recursive form,

$$v(n) = \frac{n-1-l}{n} v(n-1) + \frac{1+l}{n} a(n) a^T(n) \frac{v(n-1)}{\|v(n-1\|} \tag{10}$$

Where $\frac{n-1-l}{n}$ is the weight for the last estimation and $\frac{1+l}{n}$ is the weight for the new data, the positive parameter l is called the amnesic parameter . A mathematical proof of the derivation of this expression (10) can be founded in [10].

Now, the question is how to calculate the left singular vectors U and right singular vectors V $A = U \sum V^T$ as new date arrives. As we know calculating the SVD consists of finding the eigenvalues and eigenvectors of AA^T and $A^T A$. So we update the earlier U vectors using (10) but in the following form:

$$u(n) = \frac{n-1-l}{n} u(n-1) + \frac{1+l}{n} a(n) a^T(n) \frac{u(n-1)}{\|u(n-1)\|} \tag{11}$$

$$a_2(n) = a_1(n) - a_1^T(n) \frac{u_1(n)}{\| u_1(n) \|} \frac{u_1(n)}{\| u_1(n) \|}, \tag{12}$$

The expression (10) has the effect of keeping the eigenvectors normalized.

Note that updating the earlier V vectors is not need to alter the direction, as new date arrives in rows and not in line. So the normality of the earlier V is not changed. And only the values of V are changed. Then we use the formula (11) to update the values of V.

$$v(n) = \frac{n-1-l}{n} v(n-1) + \frac{1+l}{n} a(n) a^T(n) \frac{v(n-1)}{\|v(n-1\|} \quad (13)$$

In this way , the orthogonality is always enforced when the convergence is reached .

3.2 Algorithm Summary

Combining the mechanisms discussed above, we have the candid SVD algorithm as follows.

1. Use the a batch SVD with QR method to compute the first l dominant eigenvectors, $A_n^l = \sum_{k=1}^{l} u_k s_k v_k^T$ directly .

2. When new data $a(n+1), a(n+2)\cdots$ arrives, do the followings steps.

for $j = n+1, n+2, n+3, \cdots$.

$a_1(j) = a(j)$.

3. For $i = 1, 2, \cdots, \min(l, j)$ do,

(a) if $i = j$, initialize the i th eigenvector as

$u_i(n) = u_i(j)$.

$v_i(n) = v_i(j)$.

(b) Otherwise

$$u(n) = \frac{n-1-l}{n} u(n-1) + \frac{1+l}{n} a(n) a^T(n) \frac{u(n-1)}{\|u(n-1)\|}$$

$$a_{i+1}(n) = a_i(n) - a_i^T(n) \frac{u_i(n)}{\|u_i(n)\|} \frac{u_i(n)}{\|u_i(n)\|}$$

$$v_i(n) = \frac{n-i-k}{n} v_i(n-1) + \frac{1+k}{n} a_i(n) a_i(n)^T \frac{v_i(n-1)}{\| v_i(n-1) \|},$$

$\lambda = \|V\|$.

4. Then $\sum = diag(sqrt(\lambda))$. Thus $A = U \Sigma V^T$.

3.3 Complexity Analysis

We reported experiments to study the efficiency of the new algorithm. First we presented our results on the ORL face data set. The ORL database contains 400 grayscale images of 40 persons. Each person has 10 images , each having a resolution of 92 x 112 and 256 gray levels . For this experiment, we use SVD but instead of computing the SVD model (decomposition of matrix A into matrices U , S ,and V)for all samples , we use a threshold size to build an initial model and then use BISVD to incrementally compute the SVD model for additional samples.

Our goal is to select a basis size that is small enough to produce good precision quality. If we start with a very small basis size, the entire model computation can be very fast, but due to the non-orthogonal spaces the precision quality may not be good. On the other hand a large basis size will defeat the purpose of incremental model building. We determined the optimal basis size through experiments, where we first fix a basis size and compute the SVD model by projecting the rest of samples using incremental technique. We start with a model of $l = 10$ and go up to 30 with an increment of 5 at each step. We apply the correlation between the estimated unit eigenvector v and the one $\tilde{v}$ computed by the batch method to evaluate the quality of precision. The dot product of v and $\tilde{v}$ is represented by their inner product of $v.\tilde{v}$, Thus, the larger the correlation, the better. We observed that the correlation improved as we increased the basis size. We noticed that after the basis size crossed 30, the improvement in the correlation became relatively small. So we select 30 to be our threshold basis size. Fig 1 shows the result of the correlation . From fig 1, we can see that correlation of the 1-8th eigen is close to one. But the value of 9-10th is about 0.6. The Eigenvalue computed by our algorithm and by batch method is showing in fig 2. From fig 2, we can see two values tallied well, thus illustrated the precision of our algorithm is feasible. We apply our new method to some faces, as showing in fig 3. From fig 3, we noted that the precision of our method is not very high as batch method, but our method save computation and memory. The comparative of the time of computing SVD of face between our method and batch method is showing in table 1. Showing in Fig 4 is the results of the data computed by our new method $A' = USV^T$ minus the data original image A. We observed that the value is close to zero. Thus proved the efficiency of the new ways in an other way.

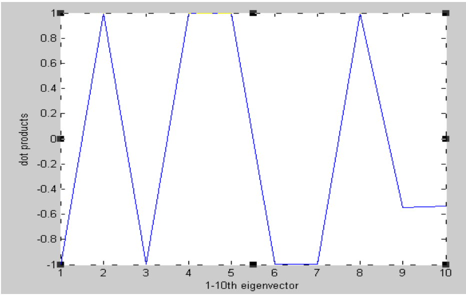

Fig 1: The dot products of $v.\tilde{v}$

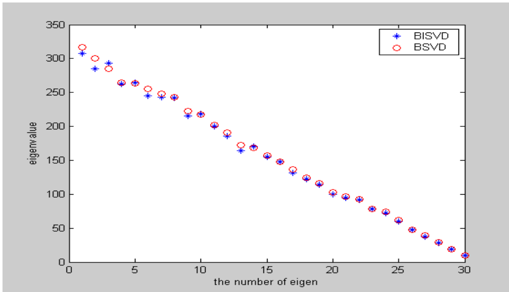

Fig 2 : The eigenvalues computed by our BISVD and batch SVD

Table1: The Average Execution Time for Calculating SVD

Batch SVD	BISVD
76s	0.93s

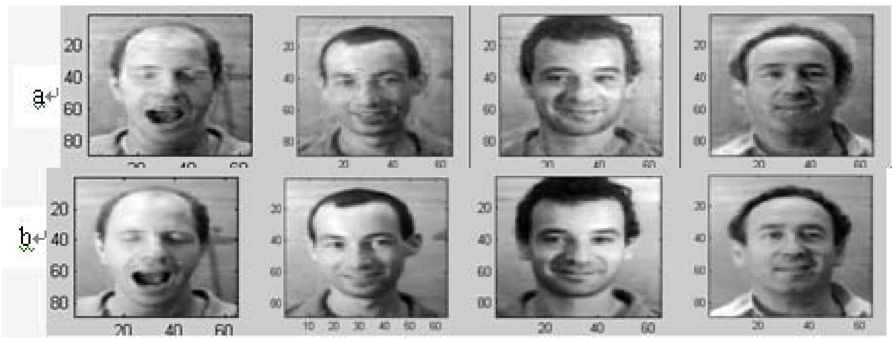

Fig 3: a is the reconstruct image (applied our algorithm), b is the originate face

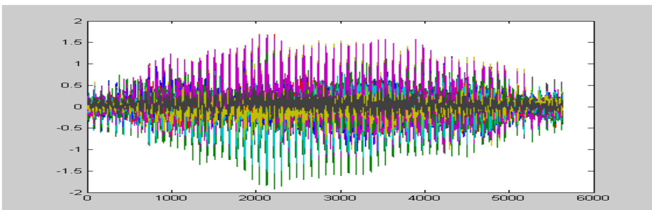

Fig 4: The results of the data computed by our new method $A' = USV^T$ minus the data original image A

4. Conclusion

An incremental approach to approximating the singular value decomposition of a correlation matrix has been presented. Using the incremental approach means that singular value decomposition is an option in situations where data takes the form of single serially-presented observations from an unknown matrix. The method is particularly appropriate the classifier of faces, where datasets are often too large to be processed by traditional methods, and situations where the dataset is unbounded, For example, in the eigenface method, a moderate gray image of 64 rows and 88 columns results in a d-dimensional vector with $d = 5,632$. The symmetric covariance matrix requires elements, which amounts to 15,862,528 entries! The approach produces preliminary estimations of the top vectors, meaning that some samples become available early in the training process. By avoiding matrix multiplication, data of high dimensionality can be processed. The efficiency of our method has been discussed here. Future work will include an evaluation on much larger corpora.

Acknowledgments

The authors are grateful to the anonymous reviewers for their high quality reviews and valuable comments. This project is supported by Scientific Research Fund of Hunan Provincial Education Department (11C0009).

Reference

[1] M. Kirby, L. Sirovich,"Application of the K-L procedure for the characterization of human faces", IEEE Transactions on Pattern Analysis and Machine Intelligence vol.12 ,no.1, pp.103–108,1990.
[2] M.A. Turk, A.P. Pentland, "Eigenfaces for recognition", Journal of Cognitive Neuroscience vol.3,no.1, pp.71–86,1991.
[3] Peter N. Belhumeur, Joao P. Hespanha, and David J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection", IEEE Transactions on Pattern Analysis and Machine Intelligence vol.19,no.7, pp. 711–720,1997.
[4] Wen-Hui Yang,Dao-Qing Dai,Hong Yan , "Feature extraction and uncorrelated discriminant analysis for high-dimensional data", IEEE Transactions on Knowledge and Data Engineering,vol.20 no.5,pp.601–614,2008.
[5] Neil Muller, Lourenço Magaia and B. M. Herbst, "Singular value decomposition eigenfaces and 3D reconstructions", SIAM Review,vol.46 ,no.3 ,pp.518–545,2004.
[6] D.Q. Dai, H. Yan, "Matrix decomposition for feature generation from high dimensional data", in: E.A. Zoeller (Ed.), Pattern Recognition Theory and Application, Nova Science Publishers, New York, 2007.
[7] Peter Hall, David Marshall, Ralph Martin, "Adding and subtracting eigenspaces with eigenvalue decomposition and singular value decomposition", Image and Vision Computing vol.20,pp.1009–1016,2002.
[8] Matthew Brand, "Incremental singular value decomposition of uncertain data with missing values", Lecture Notes in Computer Science, Vol.2350/2002,pp. 707-720,2002,
[9] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl, " Incremental Singular Value Decomposition Algorithms for Highly Scalable Recommender Systems", Fifth International Conference, pp.261- 273, 2002,
[10] Rong Hu, Weihong Xu, Jianping Wang, "Candid Covariance-Free Incremental Singular Value Decomposition", Journal of Information and Computational Science, Vol.7,no.13,pp.2607-2612,2010.
[11] Erkki Oja, Juha Karhune, "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix", J. Math. Analysis and Application, vol.106, pp.69–84,1985.
[12] Terence D. Sanger, "Optimal unsupervised learning in a single-layer linear feed forward neural network". Neural Networks, vol.2, pp.459–473,1989.

[13] Juyang Weng, Yilu Zhang, Wey Shiuan Hwang, “Candid covariance-free incremental principal component analysis”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.25, pp. 1034–1040,2003.

[14] Neil Muller, Lourenco. Magaia, B.M. Herbst, “Singular value decomposition eigenfaces, and 3D reconstructions”, SIAM Review, vol.46, pp.518–545,2004.

[15] Chuan-Xian Ren, Dao-Qing Dai , “Incremental learning of bidirectional principal components for face recognition”, Pattern Recognition, vol.43, pp.318-330, 2010

[16] Chuan-Xian Ren, Dao-Qing Dai, “Incremental learning of bidirectional principal components for face recognition”, Pattern Recognition, vol.43, pp. 318 – 330, 2010.

[17]Miao Cheng, Bin Fang, Yuan Yan Tang, Taiping Zhang, Jing Wen, “Incremental Embedding and Learning in the Local Discriminant Subspace With Application to Face Recognition”, IEEE Transation on Systems, Man, and Cybernetics, Vol 40, pp. 580-591, 2010.

[18] WANG Hai-xing, MA Wen-jie, "Improved ACA for Optimal Loading Problem of Multi-Category Goods", IJACT: International Journal of Advancements in Computing Technology, Vol. 3, No. 10, pp. 364 ~ 370, 2011

[19] Yongliang Xiao, Shaoping Zhu, Weizhong Luo, , Wenbin Liu, , Qian He, "An Efficient Feature Extraction Method for Shot Boundary Detection", IJACT: International Journal of Advancements in Computing Technology, Vol. 3, No. 10, pp. 172 ~ 179, 2011