SEQUENTIAL UPDATING ALGORITHM FOR EXTRACTING THE BASIS OF KARHUNEN-LOEVE TRANSFORMATION

Qu Yanyun[1,2], Zheng Nanning[1], Li Cuihua[2], Yuan Zejian[1]

1. The Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University
2. Computer Science Department, Xiamen University

ABSTRACT

Karhunen-Loeve transformation (KLT) is a popular method for dimensional reduction and feature extraction in image analysis, signal processing, automatic control systems, and so on, while the drawback of the KLT is expensive computation. In this paper, we propose a novel updating algorithm for KLT, the rank-k updating algorithm, which has advantages especially for image sequences: it is faster than batch algorithm, it can handle the dynamic database, and it does not save the entire database. Furthermore it makes the active learning and recognition possible in computer vision. Finally we analyze the computational complexity and error of the algorithm. We also show its application in face analysis. The experimental results demonstrate the efficiency of our algorithm.

1. INTRODUCTION

Karhunen-Loeve transformation (KLT), which is also called principal component analysis (PCA), is a popular technology for dimensional reduction or feature extraction in image analysis, signal processing, automatic control systems, etc. However, its computational demand and batch calculation limit its application, especially in face recognition [2][7][5] and viewpoint transformation [1] where the image vector is of high dimension, which made KLT compute very slow.

The core of KLT is the eigendecomposition. The batch algorithms such as [3] [10] require that the database is deterministic, which implies we must store all the input data before proceeding to calculate the basis of KLT. If the data are of m dimension and have n vectors, the required memory is $O(mn)$ which would exceed the resource of common memory if m and n is very large, and the computational complexity is $O(mn^2)$. The updating algorithms are proposed to deal with the problem, which are prevalent in signal processing. Rank-one algorithm is discussed by the researchers [8][6][4], which takes the data one by one and update the basis of KLT.

In this paper, we propose a rank-k updating algorithm motivated by the idea of rank-one algorithm [9]. The algorithm has the common advantages of updating algorithm especially for image sequence: it is faster than the batch algorithm, and it can handle a time sequence when they arrive rather than wait for the end of the sequence. The proposed algorithm is faster than rank-one algorithm that is faster than the batch algorithm [9]. The required memory space is $O(km)$, where m denotes the dimension of the vector and k denotes the number of the appended vectors at every time step.

In section 2, we propose the rank-k updating algorithm. The computational complexity and accuracy of the algorithm are analyzed in section 3. Section 4 provides experimental results. Conclusions are given in section 5.

2. SEQUENTIAL UPDATING ALGORITHM

Let $\{x_1, \quad x_2, \quad \cdots \quad x_n\}, x_i \in R^m$ be a set of n vectors, $\mu = \frac{1}{n}\sum_{k=1}^{n} x_k$ be the mean of the set, $B_n = [x_1 - \mu, x_2 - \mu, \cdots, x_n - \mu]$, and the covariance matrix S_T was defined as: $S_T = B_n B_n^T$. The eigenvectors of S_T are the basis of KLT. Assume that μ is zero, and the singular value decomposition (SVD) of B_n is calculated, denoted by $\{U_{n'}, \Sigma_{n'}, V_{n'}\}$, ($n' \le n$), where the singular values in $\Sigma_{n'}$ is arranged in decreasing order and the singular values which are smaller than the threshold ε are neglected. Let $X_k = [x_{n+1}, x_{n+2}, \cdots x_{n+k}]$, and $\{x_{n+i}\}_{i=1}^{k}$ be the set of the new appended vectors. We compute the SVD of matrix $B_{n+k} = \left[U_{n'} \Sigma_{n'} V_{n'}^T \quad X_k\right]$.

0-7803-8554-3/04/$20.00 ©2004 IEEE.

Let $span(U_{n'}) = \Re$, we want to find more orthogonal vectors from the orthogonal compensate space $\Re^{\perp}$. Define two linear transforms $\hat{U}, \hat{V}$ as below:

$$\hat{U} = \begin{bmatrix} U_{n'} & U_a \end{bmatrix},$$

$$\hat{V} = \begin{bmatrix} V_{n'} & \\ & I_{k\times k} \end{bmatrix},$$

$$U_a H = X_k - U_{n'} U_{n'}^T X_k,$$

where U_a is the unitary matrix and H is the upper triangle matrix. Note that $(I - U_{n'} U_{n'}^T) U_{n'} = 0$. We multiply $\hat{U}^T$, $\hat{V}$ on the left and right of B_{n+k} respectively, which yields that $\begin{bmatrix} \Sigma_{n'} & U_{n'}^T X_k \\ 0 & H \end{bmatrix}$,

then compute the SVD of the above matrix. We propose the rank-k algorithm as below in matlab-like language.

Rank-k Eigenspace Update Algorithm:

Initialization: the first k vectors form the matrix X_0, and calculate the SVD, denoted by $\{U_{old}, \Sigma_{old}, V_{old}\}$

for $i = 2,3,\cdots$

1. $X1 = U_{old}^T X_k$;
2. $U_a H = X_k - U_{old} X1$;
3. Calculate the SVD of matrix

$$A = \begin{bmatrix} \Sigma_{old} & X1 \\ 0 & H \end{bmatrix} = U' \Sigma' V'^T;$$

Find k1 such that $\sigma_{k1} > \varepsilon > \sigma_{k1+1}$;

4. U_{new}=the first k1 columns of $\begin{bmatrix} U_{old} & U_a \end{bmatrix} U'$;

$U_{old} = U_{new}$;

Σ_{old}=the $k1 \times k1$ upper left corner in Σ';

end

Remark 1. If k=1, the algorithm is rank-one algorithm [9]. But the above algorithm omits the computation of V_i, so the computation is less costly.

Remark 2. For the step *Initialization,* we can do the rank-one updating algorithm as below:

1). $U_{old} = x_1 / \|x_1\|$, $\Sigma_{old} = \|x_1\|$

2). Perform the rank-one algorithm until the number accumulates to k.

Remark 3. We perform Gram-Schmidt orthogonalization in step 2. If the diagonal elements in matrix H are smaller than the threshold δ, that is $H(i,i) \le \delta, (1 \le i \le k)$, the i^th column should be deleted corresponding to the matrix $U_a, X1, H$, because it implies that x_{n+i} can be well represented by U_{old}.

3. COMPUTATIONAL COMPLEXITIE AND ACCURACY

Regardless of the initialization, we count the main multiplication computation flops in one iteration of the rank-k algorithm, see as TABLE 1, where m denotes the dimension of vector x_i, n denotes the number of the vector in the database, i denotes the i^th loop, k denotes the number of appended vectors and $p = n/k$. The minimum of total operation is $O(mnk)$ when k is optimal.

In the following, we discuss the error. Two primary error sources are the round-off error incurred in calculating the SVD of matrix A and the error caused by truncating the singular values smaller than the threshold ε. The SVD function in matlab is backward stable, and special processing as remark 3 can keep the rank-k algorithm numerical stable. Now we discuss the truncating error. The error of s^th iteration is $|\sigma_i(B_s) - \sigma_i| \le \|E_s\|_2 \le \varepsilon$, where $\sigma_i(B_s)$ stands for the i^th exact singular value of matrix B_s, σ_i stands for the numerical singular value correspondingly, and E_s stands for the error matrix. So the total error of the rank-k algorithm is $|\sigma_i(B_n) - \sigma_i| \le \sum_{s=1}^{p} \|E_s\|_2 \le p\varepsilon$. We can choose proper threshold ε to control the error.

TABLE 1 Computation Complexity Of Rank-k Updating Algorithm In i^th Iteration And The Total Operation

	Multiplication
Step 1	$O(mik^2)$
Step 2	$O(3/2mk + 1/2mk^2 + mik^2)$
Step 3	$O(i^2k^3 + i^2k^2)$
Step 4	$O(i^2k^2m)$
Total	$O(3/2mn^2 + 5/3mnk + mn^2p/3)$

4. EXPERIMENTAL RESULTS

We performed a set of experiments to demonstrate the efficacy of the proposed algorithm. The proposed algorithms can handle the additional images without recomputing the basis set from scratch. We do not detail the difference between the batch algorithm and the updating algorithm. Some comparisons are given in [9]. The experiments are done in Matlab6.0 on PIII 800 computer.

The first experiment compares efficiency among the rank-one algorithm, the rank-2 and the rank-3 algorithm. In the experiment, Germany MPI(Max-Planck Institute for Biological Cybermtics) face image database was used. In face analysis, the first step is to normalize the face images, which is to registering the eye locations, the upper and bottom bounder of the face images manually. We use 100 female images of the particular database. We take 90 normalized face images as the training set, and the others as the testing set. In Fig. 1, computing times for the three schemes are plotted. As expected, the rank-2 algorithm is much faster than the rank-one, and the rank-3 algorithm is faster than the rank-2 algorithm.

The second experiment compares the average reconstruction error between the rank-one and rank-k ($k = 3$) updating algorithm. We choose ten images from the training set. The average error can be represented as

$$err = \frac{1}{10}\sum_{i=1}^{10}\left\| x_i - Ux_i \right\|_2 .$$

As shown in Fig. 2, the average residual of rank-3 algorithm is smaller than that of rank-one algorithm. And the average residual is decreasing as the number of image is increasing.

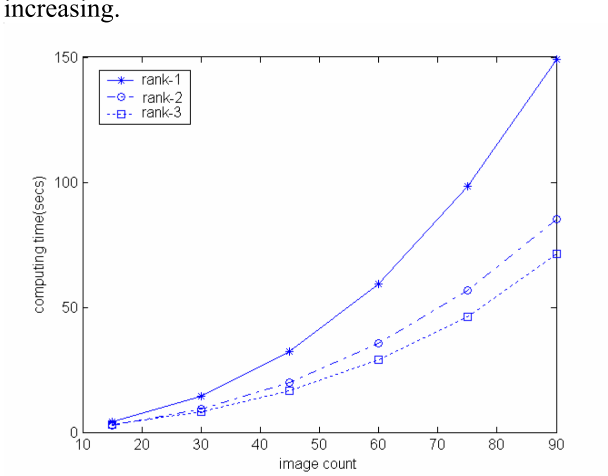


Fig 1. Computing time compare
rank-one denoted by "star solid", rank-2 denoted by "circle dash-dot" , rank-3 denoted by "square dot".

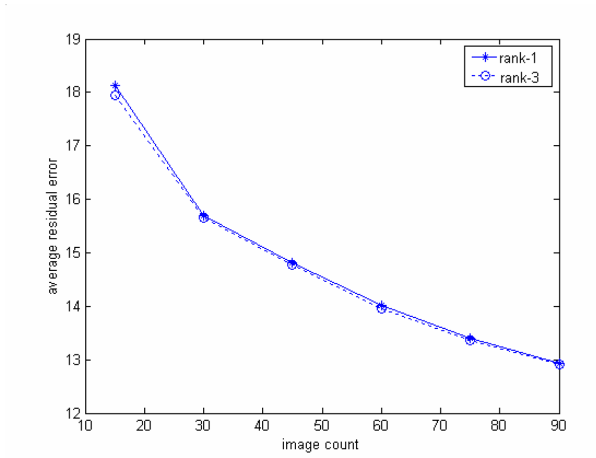


Fig 2. Average residual compare
rank-one denoted by "star solid", rank-3 denoted by "circle dot".

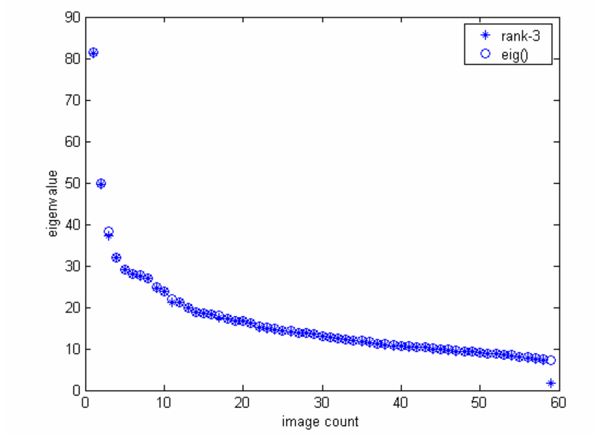
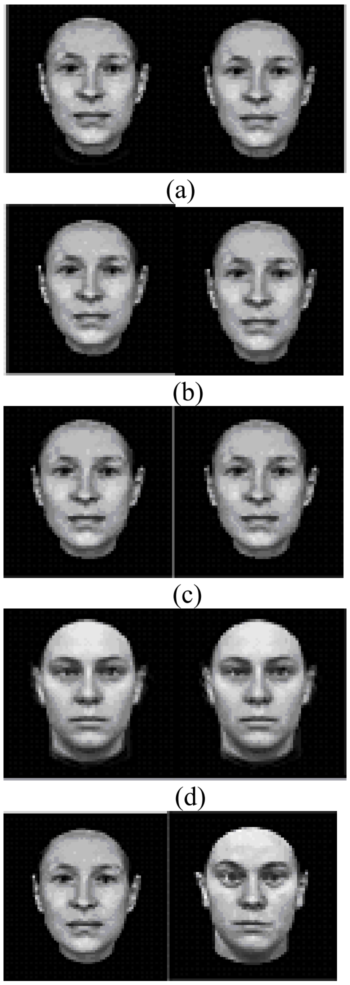


Fig 3. Eigenvalues of 60 image compare;
"star" stands for rank-3 updating, "circle" stands for standard eig().

The third experiment compares the singular values obtained by rank-k ($k = 3$) updating algorithm with those by the classical scheme that is to calculate the eigenvalues of matrix $B^T B$ by matlab function eig(). The image count is 60. It's shown that the singular values are almost consistent in the two schemes, which implies that the presented algorithm is acceptable.

The fourth experiment compares the visible reconstructed image. We take an image in the training set and an image in the testing set as examples. The two images are represented in the eigenspaces calculated by rank-k (k=1, k=3) algorithm. As shown in Fig 4, the image in the training set can be well represented, and the vision effect is better and better with increase in images. The image in the testing set isn't as good as the image in the training set, and the reconstructed image is more like a mean face.



(a) (b) (c) (d) (e) (f)

Fig 4. Visual compare of the reconstructed image

(a) (b) (c) is the reconstructed group of the image which comes from the training set, and each has two images, left of which is by rank-one algorithm, right of which is by rank-3 algorithm (a) is reconstructed by 30 images; (b) is reconstructed by 60 images; (c) is reconstructed by 90 images (d) has two images, which are the reconstruction of the image coming from the testing set by 90 images; (e) is the original image of (a)(b)(c); (f) is the original image of (d).

5. CONCLUSIONS

Karhunen-Loeve transform was widely used to reduce dimension or to extract feature in computer vision, or signal processing. But the computation is costly. In order to improve the efficiency of KLT, we propose a novel algorithm, which is the rank-k updating algorithm. The algorithm presents a way to processing data on-line or active learning. The comparison with the rank-one algorithm shows that our algorithm is faster and achieves similar accuracy level for image reconstruction. The flops of the algorithm are acceptable. We also illustrate the error resulting from the operation truncating the smallest left singular pairs. The experimental results demonstrate the efficiency of the proposed algorithms.

ACKNOWLEDGEMENTS

The research is supported by National Nature Science Foundation of China (No.60175008, No.60175006).

REFERENCE

[1] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition", in Proc. IEEE Conf. CVPR'94, Seattle, pp.84-91.

[2] B.K. Gunturk, A.U. Batur, "Eigenface-domain super-resolution for face recognition", IEEE Trans. Image Processing, vol.12, pp.597-606, May, 2003.

[3] G.H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions", Numer. Math. 1, pp.403-20,1970.

[4] Guanghan Xu, Yongman Cho, and Thomas Kailath, "Application of fast subspace decomposition to signal processing and communication problems", IEEE Trans. Signal Processing, vol. 42, No. 6, pp1453-1461, June, 1994.

[5] L. Sirovich and M. Kirby, "Low dimensional procedure for the characterization of human faces", J. Opt. Soc. Amer. A, vol.4, No.3, pp.519-524,1987.

[6] M. Gu and Stanley C. Eisenstat, "A stable and fast algorithm for updating the singular value decomposition", Research Report YALEU/DCS/RR-966, Yale University, New haven, CT, 1994.

[7] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 19, No. 7, pp.711-720, July, 1997.

[8] P. Comon, G.H. Golub, "Tracking a few extreme singular values and vectors in signal processing", Proceedings of the IEEE, vol. 78, No. 8, pp.1327-1343, August, 1990.

[9] S. Chandrasekaran, B.S. Manjunath, Y.F. Wang, J.Winkeler, H. Zhang, "An eigenspace update algorithm for image analaysis", Graphical models and Image Processing, Vol.59, No.5, pp.321-332, Sept. 1997.

[10] T.F. Chan, "An improved algorithm for computing the singular value decomposition", ACM Trans. Math. Soft.8, pp.72-83,1982.