SYMMETRIC GENERALIZED LOW RANK APPROXIMATIONS OF MATRICES

Kohei Inoue, Kenji Hara, and Kiichi Urahama

Kyushu University, Department of Communication Design Science
4-9-1, Shiobaru, Minami-ku, Fukuoka, 815-8540 Japan

ABSTRACT

Recently, the generalized low rank approximations of matrices (GLRAM) have been proposed for dimensionality reduction of matrices such as images. However, in GLRAM, it is necessary for users to specify the numbers of rows and columns in low rank matrices. In this paper, we propose a method for determining them semiautomatically by symmetrizing GLRAM. Experimental results show that the proposed method can determine the optimal ranks of matrices while achieving competitive approximation performance.

***Index Terms*—** Dimensionality reduction, GLRAM, symmetric GLRAM, matrices

1. INTRODUCTION

Principal component analysis (PCA) and linear discriminant analysis (LDA) are well-known techniques for dimensionality reduction. Since they are based on vectors, matrices such as 2D face images must be transformed into 1D image vectors in advance. However, the resultant vectors usually lead to a high-dimensional vector space, where it is difficult to solve the (generalized) eigenvalue problems for PCA and LDA.

Recently, Yang et al. [1] have proposed 2DPCA, and Ye [2] has proposed generalized low rank approximations of matrices (GLRAM). These methods can handle matrices directly without vectorizing them. Ye [2] proposed an iterative algorithm for GLRAM, which will be summarized in the next section. In GLRAM [2], a matrix A_i is approximated by the low rank matrix $M_i = L^T A_i R$, and Ye's iterative algorithm [2] renews two matrices L and R alternately. On the other hand, Liang and Shi [3] and Liang et al. [4] proposed an analytical algorithm which does not need to iterate the renewal procedure. Liang's analytical algorithm [3, 4] selects the better one from two cases: R calculated with an initialized L and L calculated with an initialized R. However, Hu et al. [5] and Inoue and Urahama [6] showed that Liang's analytical algorithm [3, 4] does not necessarily give the optimal solution of GLRAM. Liu and Chen [7] also proposed a non-iterative algorithm for GLRAM. However, Liu's non-iterative algorithm [7] does not select the better one from the two cases in Liang's analytical algorithm [3, 4] but always outputs the former case. Therefore, Liu's non-iterative algorithm [7] cannot outperform Liang's analytical algorithm [3, 4]. Lu et al. [8] proposed another non-iterative algorithm which calculates L and R independently. However, the same algorithm as Lu's one [8] has been shown in the paper [6] already.

This work was partially supported by the Japan Society for the Promotion of Science under the Grant-in-Aid for Scientific Research (23700212).

In GLRAM [2], it is necessary for users to specify the number of rows l_1 and that of columns l_2 in the low rank matrix M_i. Ye [2] experimentally showed that the good results are obtained when $l_1 = l_2$. Additionally, Liu et al. [9] derived a lower bound of the objective function for GLRAM and showed that the minimization of the lower bound results in $l_1 = l_2$. Ding and Ye [10] have also shown the same lower bound as Liu's one.

In this paper, we propose a method for determining l_1 and l_2 semiautomatically by symmetrizing GLRAM [2]. Although the matrices handled in GLRAM [2] are asymmetric generally, in the proposed method, we construct symmetric matrices from the asymmetric ones to derive symmetric GLRAM. In the proposed method, l_1 and l_2 are semiautomatically determined from the sum $l = l_1 + l_2$, therefore, the users do not need to specify them. Experimental results show that the proposed method achieves better objective function values than the conventional method when l is fixed to a constant.

The rest of this paper is organized as follows: Section 2 summarizes GLRAM [2], Section 3 proposes symmetric GLRAM, Section 4 shows experimental results, and Section 5 concludes this paper.

2. GENERALIZED LOW RANK APPROXIMATIONS OF MATRICES

Let $A_i \in \Re^{r \times c},\ i = 1, \ldots, n$ where $\Re$ denotes the set of real numbers. Then the generalized low rank approximations of matrices $\{A_i\}_{i=1}^n$ (GLRAM) are formulated as follows [2]:

$$\min_{L,\ R,\ \{M_i\}_{i=1}^n} \quad \sum_{i=1}^{n} \left\| A_i - L M_i R^T \right\|_F^2 \tag{1}$$

$$\text{subj.to} \quad L^T L = I_{l_1},\ R^T R = I_{l_2}, \tag{2}$$

where $L \in \Re^{r \times l_1},\ R \in \Re^{c \times l_2}$ for $l_1 < r,\ l_2 < c$, I_{l_1} and I_{l_2} denote the identity matrices of orders l_1 and l_2, and $\| \cdot \|_F$

978-1-4673-0046-9/12/$26.00 ©2012 IEEE

Table 1. Ye's algorithm [2].

Algorithm GLRAM
Input: matrices $\{A_i\}_{i=1}^n$, l_1, and l_2
Output: matrices L, R, and $\{M_i\}_{i=1}^n$

Obtain initial L_0 for L and set $t \leftarrow 1$;
While not convergent
  form the matrix $M_R = \sum_{i=1}^n A_i^T L_{t-1} L_{t-1}^T A_i$;
  compute the l_2 eigenvectors $\{\phi_j^R\}_{j=1}^{l_2}$ of M_R corresponding to the largest l_2 eigenvalues;
  $R_t \leftarrow [\phi_1^R, \ldots, \phi_{l_2}^R]$;
  form the matrix $M_L = \sum_{i=1}^n A_i R_t R_t^T A_i^T$.
  compute thte l_1 eigenvectors $\{\phi_j^L\}_{j=1}^{l_1}$ of M_L corresponding to the largest l_1 eigenvalues;
  $L_t \leftarrow [\phi_1^L, \ldots, \phi_{l_1}^L]$;
  $t \leftarrow t+1$;
EndWhile
$L \leftarrow L_{t-1}$;
$R \leftarrow R_{t-1}$;
For i from 1 to n
  $M_i \leftarrow L^T A_i R$;
EndFor

denotes the Frobenius norm. If L and R are given, then the optimal M_i is obtained by $M_i = L^T A_i R$. From

$$\sum_{i=1}^n \left\|A_i - L M_i R^T\right\|_F^2 = \sum_{i=1}^n \|A_i\|_F^2 - \sum_{i=1}^n \left\|L^T A_i R\right\|_F^2, \tag{3}$$

and that $\sum_{i=1}^n \|A_i\|_F^2$ is a constant with respect to L and R, the above minimization problem (1) may be rewritten as

$$\max_{L,\, R} \quad \sum_{i=1}^n \left\|L^T A_i R\right\|_F^2. \tag{4}$$

Ye's algorithm [2] for this problem is summarized in Table 1, in which l_1 and l_2 need to be specified by hand. Ye [2] experimentally showed that the good results are obtained when $l_1 = l_2$. Liu et al. [9] also derived the same result as Ye's one [2] from the minimization of a lower bound of the objective function of GLRAM.

3. SYMMETRIC GLRAM

In the above GLRAM [2], given matrices $\{A_i\}_{i=1}^n$ are asymmetric generally. In this section, we construct symmetric matrices from the asymmetric ones $\{A_i\}_{i=1}^n$ as follows:

$$S_i = \begin{pmatrix} O_{c,c} & A_i^T \\ A_i & O_{r,r} \end{pmatrix}, \quad i = 1, \ldots, n, \tag{5}$$

and then propose a low rank approximation method for symmetric matrices $\{S_i\}_{i=1}^n$, where $O_{c,c}$ denotes a $c \times c$ zero matrix. The symmetric GLRAM for $\{S_i\}_{i=1}^n$ becomes

$$\max_U \quad \sum_{i=1}^n \left\|U^T S_i U\right\|_F^2 \tag{6}$$

$$\text{subj.to} \qquad U^T U = I_l, \tag{7}$$

where $U \in \Re^{(r+c)\times l}$, I_l denotes the identity matrix of order l, and $l < r + c$. Let $F(U)$ be the objective function in (6). Then we find that

$$F(U) = \sum_{i=1}^n \mathrm{tr}\left(U^T S_i U U^T S_i U\right) = \sum_{i=1}^n \mathrm{tr}\left[\left(U U^T S_i\right)^2\right], \tag{8}$$

from which the Lagrange function for (6) with (7) is given by

$$\mathcal{L}(U, \Lambda) = \sum_{i=1}^n \mathrm{tr}\left[\left(U U^T S_i\right)^2\right] - 2\mathrm{tr}\left[\Lambda\left(U^T U - I_l\right)\right], \tag{9}$$

where $\Lambda \in \Re^{l\times l}$ is a symmetric matrix of which the elements are the Lagrange multipliers and tr denotes the matrix trace. Then we have the necessary conditions for optimality:

$$\frac{1}{4}\frac{\partial \mathcal{L}}{\partial U} = \sum_{i=1}^n S_i U U^T S_i U - U\Lambda = O_{(r+c),l}, \tag{10}$$

$$\frac{\partial \mathcal{L}}{\partial \Lambda} = U^T U - I_l = O_{l,l}. \tag{11}$$

From (10), we have

$$\sum_{i=1}^n S_i U U^T S_i U = U\Lambda, \tag{12}$$

and, from (11), we have $U^T U = I_l$, which is no less than the constraint in (7). Based on (12), we propose an algorithm in Table 2, where input data are matrices $\{A_i\}_{i=1}^n$ and the rank l or the number of columns in U. While Ye's algorithm [2] in Table 1 needs both l_1 and l_2 for L and R respectively, the proposed algorithm in Table 2 needs only l for U.

The details of the algorithm in Table 2 are as follows: First we form symmetric matrices $\{S_i\}_{i=1}^n$ defined by (5) (Line 1). Next we compute the l eigenvectors $\{\tilde{\phi}_j\}_{j=1}^l$ corresponding to the largest l eigenvalues of $\sum_{i=1}^n S_i^2$ and then initialize U as $U_0 = [\tilde{\phi}_1, \ldots, \tilde{\phi}_l]$, and initialize the number of iterations, t, as $t = 1$ (Line 2). Then, for example, U after t iterations is expressed as U_t. In the iterative procedure, we first form $M = \sum_{i=1}^n S_i U_{t-1} U_{t-1}^T S_i$ and then compute the l eigenvectors $\{\phi_j\}_{j=1}^l$ corresponding to the largest l eigenvalues of M to form $U_t = [\phi_1, \ldots, \phi_l]$. We repeat this procedure until the convergence condition described below is satisfied (Lines 3-8). We used the convergence condition as $\frac{\mathrm{RMSE}^{(t-1)} - \mathrm{RMSE}^{(t)}}{\mathrm{RMSE}^{(t-1)}} < \epsilon$ for $t = 1, 2, \ldots$, where $\mathrm{RMSE}^{(t)}$ denotes the root mean square error $\mathrm{RMSE}^{(t)} = \sqrt{\frac{1}{n}\sum_{i=1}^n \left\|S_i - U_{t-1}U_{t-1}^T S_i U_t U_t^T\right\|_F^2}$ after t iterations

Table 2. The proposed algorithm

Algorithm Symmetric GLRAM
Input: matrices $\{A_i\}_{i=1}^{n}$ and l
Output: matrices L, R, and $\{M_i\}_{i=1}^{n}$
1. Form symmetric matrices $\{S_i\}_{i=1}^{n}$.
2. Obtain initial U_0 for U and set $t \leftarrow 1$;
3. While not convergent
4. form the matrix $M = \sum_{i=1}^{n} S_i U_{t-1} U_{t-1}^T S_i$;
5. compute the l eigenvectors $\{\phi_j\}_{j=1}^{l}$ of M corresponding to the largest l eigenvalues;
6. $U_t \leftarrow [\phi_1, \dots, \phi_l]$;
7. $t \leftarrow t + 1$;
8. EndWhile
9. $U \leftarrow U_{t-1}$;
10. $L = [\ ]$;
11. $R = [\ ]$;
12. For j from 1 to l
13. $u \leftarrow U(1 : c, j)$;
14. $v \leftarrow U(c + 1 : r + c, j)$;
15. If $\|u\| \geq \|v\|$
16. $R \leftarrow [R, u]$;
17. Else
18. $L \leftarrow [L, v]$;
19. EndIf
20. EndFor
21. For i from 1 to n
22. $M_i \leftarrow L^T A_i R$;
23. EndFor

and ϵ is a positive constant, provided that $\mathrm{RMSE}^{(0)} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|S_i - U_0 U_0^T S_i U_0 U_0^T\|_F^2}$. We express the converged U_t as U (Line 9). Then we make L and R from U as follows: First we initialize L and R to empty arrays (Lines 10, 11). Let u be a vector of which the elements are the first c elements in the jth column of U (Line 13) and let v be a vector of which the elements are the rest r elements in the jth column of U (Line 14). If $\|u\| \geq \|v\|$ (Line 15) then add u into the last column of R (Line 16), or else add v into the last column of L (Line 18). For $j = 1, \dots, l$, we repeat this procedure (Lines 12-20). Since the diagonal blocks of S_i are zero matrices as shown in (5), the jth column $u_j \in \Re^{r+c}$ of $U = [u_1, \dots, u_l]$ has the form like $[u_{1,j}, \dots, u_{c,j}, 0, \dots, 0]^T = [u^T, 0, \dots, 0]^T$ or $[0, \dots, 0, u_{c+1,j}, \dots, u_{r+c,j}]^T = [0, \dots, 0, v^T]^T$. The lines 15-19 in Table 2 describe the procedure for extracting the nonzero elements u or v. Finally, we compute the low rank approximation of A_i by $M_i = L^T A_i R$ (Lines 21-23).

4. EXPERIMENTAL RESULTS

In this section, we show experimental results on the ORL face image database [11]. Fig. 1 shows face images in the ORL database [11]. The ORL database [11] contains face images of 40 persons. For each person, there are 10 different face images. In our experiments, we used the first 5 images per person, i.e., $n = 5 \times 40 = 200$. The height and width of an image are $r = 112$ and $c = 92$ pixels, respectively.

Fig. 1. Face images in the ORL face database [11].

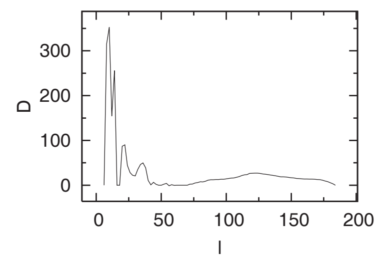

Fig. 2. Difference D vs. $l = l_1 + l_2$.

In Ye's GLRAM [2], it is shown that the good results are obtained when

$$l_1 = l_2 \tag{13}$$

is satisfied [2, 9]. Thus, we call the GLRAM with the constraint (13) the constrained GLRAM (CGLRAM), and compare it with the proposed method.

Let $L_Y \in \Re^{r\times h}$ and $R_Y \in \Re^{c\times h}$ be the matrices L and R obtained by CGLRAM, where $h = \frac{l_1+l_2}{2}$, and let $L_O \in \Re^{r\times l_1}$ and $R_O \in \Re^{c\times l_2}$ be that by the proposed method. Then we evaluate the value of $D = \sum_{i=1}^{n}(\left\|L_O^T A_i R_O\right\|_F^2 - \left\|L_Y^T A_i R_Y\right\|_F^2)$, that is, the difference between the two objective function values. If $D > 0$, then the objective function value obtained by the proposed method is larger than that by CGLRAM. The value of D is shown in Fig. 2, where the vertical axis denotes D and the horizontal axis denotes $l = 2h = l_1 + l_2$. In this figure, D is positive in almost all range of l, and therefore the objective function value by the proposed method is larger than or equal to that by CGLRAM. Since the proposed method accepts different values for l_1 and l_2, the objective function value may be different from that by CGLRAM. The values of l_1 and l_2 is shown in Fig. 3, where the proposed method and CGLRAM are denoted by the solid and the broken lines, respectively.

Additionally, in CGLRAM, the value of $l = l_1 + l_2 = 2h$ is restricted to even numbers, and therefore we cannot select odd numbers for l. On the other hand, in the proposed method, we can select both even and odd numbers for l. The

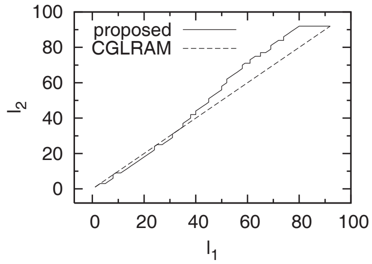

Fig. 3. l_2 vs. l_1.

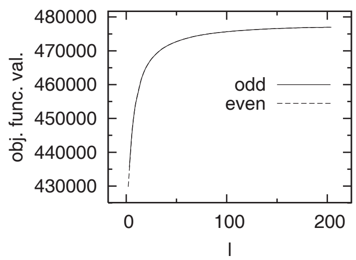

Fig. 4. Variation in the objective function value.

objective function value for the proposed method is shown in Fig. 4, where the solid and the broken lines correspond to the parity of l, i.e., odd and even numbers, respectively. The overlap between the solid and the broken lines in this figure shows that the proposed method achieves comparable performance when l is an odd number, with that when l is an even number. Finally, the reconstructed images $\tilde{A}_i = LM_iR^T$ are shown in Fig. 5, where the leftmost images are the original ones and the corresponding reconstructed images for $l = 5,\ 10,\ 15,\ \ldots, 45$ are arranged to their right.

Thus, in the proposed method, only l is needed to compute the low rank approximations of matrices instead of l_1 and l_2 for GLRAM [2]. Furthermore, while $l = l_1 + l_2$ in CGLRAM is restricted to even numbers, the proposed method accept both even and odd numbers for l.

5. CONCLUSION

In this paper, we proposed a method for determining semiautomatically the numbers of rows and columns in low rank matrices in the generalized low rank approximations of matrices (GLRAM) by symmetrizing GLRAM, and experimentally showed that the proposed method achieves larger objective function value than the conventional GLRAM (CGLRAM) which uses the same numbers of rows and columns. Additionally, while the total number of rows and columns in CGLRAM is restricted to even numbers, the proposed method accepts both even and odd numbers of rows and columns of low rank matrices.

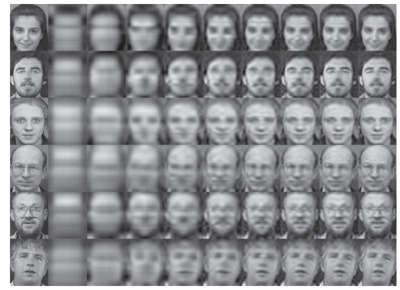

Fig. 5. Original images (the leftmost column) and their reconstructed images.

6. REFERENCES

[1] J. Yang, D. Zhang, A. F. Frangi, and J. Yang, "Two-dimensional pca: A new approach to appearance-based face representation and recognition," *IEEE Trans. PAMI*, vol. 26, pp. 131–137, 2004.

[2] J. Ye, "Generalized low rank approximations of matrices," *Mach. Learn.*, vol. 61, pp. 167–191, 2005.

[3] Z. Liang and P. Shi, "An analytical algorithm for generalized low-rank approximations of matrices.," *Pattern Recogn.*, vol. 38, pp. 2213–2216, 2005.

[4] Z. Liang, D. Zhang, and P. Shi, "The theoretical analysis of glram and its applications," *Pattern Recogn.*, vol. 40, pp. 1032–1041, 2007.

[5] Y. Hu, H. Lv, and X. Zhang, "Comments on "an analytical algorithm for generalized low-rank approximations of matrices"," *Pattern Recogn.*, vol. 41, pp. 2133–2135, 2008.

[6] K. Inoue and K. Urahama, "Equivalence of non-iterative algorithms for simultaneous low rank approximations of matrices," in *Proc. CVPR*, 2006, vol. 1, pp. 154–159.

[7] J. Liu and S. Chen, "Non-iterative generalized low rank approximation of matrices," *Pattern Recogn. Lett.*, vol. 27, pp. 1002–1008, 2006.

[8] C. Lu, W. Liu, and S. An, "A simplified glram algorithm for face recognition," *Neurocomput.*, vol. 72, pp. 212–217, 2008.

[9] J. Liu, S. Chen, Z.-H. Zhou, and X. Tan, "Generalized low-rank approximations of matrices revisited," *Trans. Neur. Netw.*, vol. 21, pp. 621–632, 2010.

[10] C. H. Q. Ding and J. Ye, "2-dimensional singular value decomposition for 2d maps and images," in *SDM*, 2005.

[11] F. Samaria and A. Harter, "Parameterisation of a stochastic model for human face identification," in *IEEE Workshop on Applications of Computer Vision*, 1994.