

Robust Simultaneous Low Rank Approximation of Tensors

Kohei Inoue, Kenji Hara, and Kiichi Urahama

Kyushu University, Fukuoka 815-8540, Japan
{k-inoue,hara,urahama}@design.kyushu-u.ac.jp

Abstract. We propose simultaneous low rank approximation of tensors (SLRAT) for the dimensionality reduction of tensors and modify it to the robust one, i.e., the robust SLRAT. For both the SLRAT and the robust SLRAT, we propose iterative algorithms for solving them. It is experimentally shown that the robust SLRAT achieves lower reconstruction error than the SLRAT when a dataset contains noise data. We also propose a method for classifying sets of tensors and call it the subspace matching, where both training data and testing data are represented by their subspaces, and each testing datum is classified on the basis of the similarity between subspaces. It is experimentally verified that the robust SLRAT achieves higher recognition rate than the SLRAT when the testing data contain noise data.

1 Introduction

Dimensionality reduction is an important topic in image processing, pattern recognition, computer vision and data mining researches. Recently, Yang et al. [1] presented two-dimensional principal component analysis (2DPCA) for reducing the dimensions of matrices. In the 2DPCA, each matrix does not need to be transformed into a vector prior to the dimensionality reduction. However, the 2DPCA is approximately equivalent to the traditional PCA operated on the row vectors of matrices [2,3,4]. Ye [5] proposed generalized low rank approximation of matrices (GLRAM). Different from the 2DPCA, the GLRAM reduces the dimensions of both rows and columns of matrices. Inoue and Urahama [6] showed a relationship between the GLRAM and the other non-iterative algorithms. Ding et al. [7] provided the error analysis of these methods and derived error bounds similar to Eckart-Young theorem which plays critical role in the development and application of singular value decomposition (SVD). Lu et al. [8] proposed a multilinear PCA (MPCA) for tensor object feature extraction and discussed the issues of initialization, convergence and subspace dimensionality determination. Huang and Ding [9] proposed robust tensor factorization using R_1 norm, i.e., rotationally invariant L_1 norm.

In this paper, we propose simultaneous low rank approximation of tensors (SLRAT) which is an extension of the GLRAM to higher-order tensors, and then we modify the SLRAT to its robust version. We also propose a method for classifying sets of tensors, which we call the subspace matching. The proposed

subspace matching method calculates the similarity between subspaces of sets of tensors and classifies each set of tensors on the basis of the similarity. Therefore, using the subspace matching, we can calculate the similarity between two sets of different cardinalities. Experimental results on the ORL face image database, which is a widely used face image database, show the effectiveness of the proposed methods.

The rest of this paper is organized as follows: In Section 2, we propose the SLRAT and derive an iterative solution algorithm. In Section 3, we modify the SLRAT to its robust version and derive an iterative solution algorithm. In Section 4, we propose the subspace matching method for classifying sets of tensors. Experimental results are shown in Section 5, where examples of image reconstruction and face recognition are shown. Section 6 summarizes the main results of this paper.

2 Simultaneous Low Rank Approximation of Tensors

In this section, we propose simultaneous low rank approximation of tensors (SLRAT). The notations used in this paper follow De Lathauwer et al. [10,11] and Bader and Kolda [12] mainly.

Let $\mathcal{A}_i = [a_{i_1 \dots i_N i}] \in \mathbb{R}^{I_1 \times \dots \times I_N}$ for $i = 1, \dots, M$, where $a_{i_1 \dots i_N i}$ is the (i_1, \dots, i_N) element of \mathcal{A}_i and $\mathbb{R}^{I_1 \times \dots \times I_N}$ denotes an $(I_1 \times \dots \times I_N)$ -dimensional real space. Then the simultaneous low rank approximation of tensors (SLRAT) is formulated as follows:

$$\min_{U, \{\mathcal{B}_i\}_{i=1}^M} \sum_{i=1}^M \|\mathcal{A}_i - \mathcal{B}_i \times \{U\}\|_F^2 \quad (1)$$

$$\text{subj.to } U^{(n)T} U^{(n)} = I_{R_n}, \quad n = 1, \dots, N, \quad (2)$$

where $U = \{U^{(1)}, \dots, U^{(N)}\}$ is a set of $U^{(n)} = [u_{i_n i'_n}^{(n)}] \in \mathbb{R}^{I_n \times R_n}$ for $i_n = 1, \dots, I_n$, $i'_n = 1, \dots, R_n$ and $n = 1, \dots, N$, and $\mathcal{B}_i \times \{U\} = \mathcal{B}_i \times_1 U^{(1)} \dots \times_N U^{(N)}$ [12], where $\mathcal{B}_i = [b_{i'_1 \dots i'_N i}] \in \mathbb{R}^{R_1 \times \dots \times R_N}$ and $\mathcal{B}_i \times_n U^{(n)} = [\sum_{i'_n=1}^{R_n} b_{i'_1 \dots i'_N i} u_{i_n i'_n}^{(n)}] \in \mathbb{R}^{R_1 \times \dots \times R_{n-1} \times I_n \times R_{n+1} \times \dots \times R_N}$ is the n -mode product of \mathcal{B}_i and $U^{(n)}$ [10,11]. $\|\cdot\|_F$ denotes the Frobenius norm and I_{R_n} is the $R_n \times R_n$ identity matrix. We assume that $R_n \leq I_n$ for $n = 1, \dots, N$. Let $E(U, \{\mathcal{B}\})$ be the objective function in Eq. (1). Then it follows from $\partial E / \partial \mathcal{B}_i = 0$ that

$$\mathcal{B}_i = \mathcal{A}_i \times \{U^T\}, \quad i = 1, \dots, M, \quad (3)$$

where $U^T = \{U^{(1)T}, \dots, U^{(M)T}\}$. By substituting Eq. (3) into E we find that

$$E(U) = \sum_{i=1}^M \|\mathcal{A}_i\|_F^2 - \tilde{E}(U), \quad (4)$$

where

$$\tilde{E}(U) = \sum_{i=1}^M \|\mathcal{A}_i \times \{U^T\}\|_F^2. \quad (5)$$

Since $\sum_{i=1}^M \|\mathcal{A}_i\|_F^2$ is a constant, we may rewrite Eq. (1) as follows:

$$\max_U \tilde{E}(U). \quad (6)$$

$\tilde{E}(U)$ can be written in the form

$$\tilde{E}(U) = \sum_{i=1}^M \left\| (\mathcal{A}_i \times_{-n} \{U^T\}) \times_n U^{(n)T} \right\|_F^2 \quad (7)$$

$$= \sum_{i=1}^M \left\| U^{(n)T} \tilde{A}_{i(n)} \right\|_F^2 \quad (8)$$

$$= \text{tr} \left(U^{(n)T} \tilde{A}_{(n)} U^{(n)} \right), \quad (9)$$

where tr denotes the matrix trace and $\tilde{A}_{(n)} = \sum_{i=1}^M \tilde{A}_{i(n)} \tilde{A}_{i(n)}^T$ where $\tilde{A}_{i(n)}$ is the mode- n matricizing [12] or the matrix unfolding [10,11] of

$$\mathcal{A}_i \times_{-n} \{U^T\} = \mathcal{A}_i \times_1 U^{(n)T} \cdots \times_{n-1} U^{(n-1)T} \times_{n+1} U^{(n+1)T} \cdots \times_N U^{(N)T}. \quad (10)$$

Thus, from Eq. (9), we see that if $U^{(1)}, \dots, U^{(n-1)}, U^{(n+1)}, \dots, U^{(N)}$ are fixed, then the optimal $U^{(n)}$ is a matrix whose columns are the principal eigenvectors of $\tilde{A}_{(n)}$. Consequently, we obtain an iterative algorithm as follows:

[SLRAT]

Step 0 (Initialization): Initialize $U^{(n)}$ for $n = 1, \dots, n$ as $U^{(n,0)} = [v_1^{(n)}, \dots, v_{R_n}^{(n)}]$ where $v_1^{(n)}, \dots, v_{R_n}^{(n)}$ are the eigenvectors of $\sum_{i=1}^M A_{i(n)} A_{i(n)}^T$ corresponding to the largest R_n eigenvalues, where $A_{i(n)}$ is the mode- n matricizing [12] or the matrix unfolding [10,11] of \mathcal{A}_i . Initialize the iteration counter t as $t = 0$. Initialize the root mean squared error (RMSE) at $t = 0$ as $\text{RMSE}^{(0)} = \sqrt{\frac{1}{M} \sum_{i=1}^M \|\mathcal{A}_i\|_F^2}$.

Step 1: For $n = 1, \dots, N$, compute the eigenvectors $u_1^{(n,t+1)}, \dots, u_{R_n}^{(n,t+1)}$ of $\sum_{i=1}^M \tilde{A}_{i(n,t)} \tilde{A}_{i(n,t)}^T$ corresponding to the largest R_n eigenvalues and form $U^{(n,t+1)} = [u_1^{(n,t+1)}, \dots, u_{R_n}^{(n,t+1)}]$, where $\tilde{A}_{i(n,t)}$ is the mode- n matricizing [12] or the matrix unfolding [10,11] of $\mathcal{A}_i \times_{-n} \{U_n^{(t)}\}$ for $U_n^{(t)} = \{U^{(1,t+1)}, \dots, U^{(n-1,t+1)}, U^{(n,t)}, \dots, U^{(N,t)}\}$.

Step 2: Compute the RMSE as

$$\text{RMSE}^{(t+1)} = \sqrt{\frac{1}{M} \sum_{i=1}^M \left\| \mathcal{A}_i - \tilde{\mathcal{A}}_i^{(t+1)} \right\|_F^2}, \quad (11)$$

where $\tilde{\mathcal{A}}_i^{(t+1)} = \mathcal{B}_i^{(t+1)} \times \{U^{(t+1)}\}$ for $\mathcal{B}_i^{(t+1)} = \mathcal{A}_i \times \{U^{(t+1)T}\}$ and $U^{(t+1)} = \{U^{(1,t+1)}, \dots, U^{(N,t+1)}\}$. If $(\text{RMSE}^{(t)} - \text{RMSE}^{(t+1)})/\text{RMSE}^{(t)} < \epsilon$ for $\epsilon > 0$ then proceed to the next step, otherwise increase t by 1 and go to Step 1.

Step 3: Output $U^* = U_{t+1}$ and $\mathcal{B}_i^* = \mathcal{A}_i \times \{U^{*T}\}$ for $i = 1, \dots, M$.

Note that the SLRAT is reduced to the generalized low rank approximation of matrices (GLRAM) presented by Ye [5] when $N = 2$, i.e., the objective function in Eq. (1) is reduced to

$$\sum_{i=1}^M \left\| A_i - B_i \times_1 U^{(1)} \times_2 U^{(2)} \right\|_F^2 = \sum_{i=1}^M \left\| A_i - U^{(1)} B_i U^{(2)T} \right\|_F^2, \quad (12)$$

where $A_i \in \mathbb{R}^{I_1 \times I_2}$ and $B_i \in \mathbb{R}^{R_1 \times R_2}$ are the second-order tensors or the matrices. The right hand side of Eq. (12) coincides with the objective function of the GLRAM.

3 Robust Simultaneous Low Rank Approximation of Tensors

The SLRAT described in the previous section is formulated as a minimization of the sum of the Frobenius norm. Therefore, The SLRAT is not robust to noise data. In this section, we modify Eq. (1) as follows:

$$\min_{U, \{\mathcal{B}_i\}_{i=1}^M} \sum_{i=1}^M \rho(\|A_i - \mathcal{B}_i \times \{U\}\|_F), \quad (13)$$

where $\rho(x)$ is the Welsch's function commonly used in robust statistics [13]:

$$\rho(x) = 1 - e^{-\alpha x^2}, \quad (14)$$

where $\alpha > 0$. Let $F(U, \{\mathcal{B}_i\})$ be the objective function in Eq. (13). Then we have

$$F(U, \{\mathcal{B}_i\}) = M - \sum_{i=1}^M e^{-\alpha \|A_i - \mathcal{B}_i \times \{U\}\|_F^2}. \quad (15)$$

Since M is a constant, we may rewrite Eq. (13) as follows:

$$\max_{U, \{\mathcal{B}_i\}_{i=1}^M} \tilde{F}(U, \{\mathcal{B}_i\}), \quad (16)$$

where

$$\tilde{F}(U, \{\mathcal{B}_i\}) = M - F(U, \{\mathcal{B}_i\}) = \sum_{i=1}^M e^{-\alpha \|A_i - \mathcal{B}_i \times \{U\}\|_F^2}. \quad (17)$$

Then it follows from $\partial \tilde{F} / \partial \mathcal{B}_i = 0$ that

$$\mathcal{B}_i = \mathcal{A}_i \times \{U^T\}, \quad i = 1, \dots, M, \quad (18)$$

where $U^T = \{U^{(1)T}, \dots, U^{(N)T}\}$. Also, we have

$$\frac{\partial \tilde{F}}{\partial U^{(n)}} = 2\alpha \sum_{i=1}^M \left(A_{i(n)} - U^{(n)} \tilde{B}_{i(n)} \right) \tilde{B}_{i(n)}^T e^{-\alpha \|A_i - \mathcal{B}_i \times \{U\}\|_F^2}, \quad (19)$$

where $A_{i(n)}$ and $\tilde{B}_{i(n)}$ are the mode- n matricizing [12] or the matrix unfolding [10,11] of \mathcal{A}_i and $\mathcal{B}_i \times_{-n} \{U\} = \mathcal{B}_i \times_1 U^{(1)} \dots \times_{n-1} U^{(n-1)} \times_{n+1} U^{(n+1)} \dots \times_N U^{(N)}$, respectively. From $\partial \tilde{F} / \partial U^{(n)} = 0$, we have

$$U^{(n)} = \sum_{i=1}^M A_{i(n)} \tilde{B}_{i(n)}^T e^{-\alpha \|A_i - \mathcal{B}_i \times \{U\}\|_F^2} \left(\sum_{i=1}^M \tilde{B}_{i(n)} \tilde{B}_{i(n)}^T e^{-\alpha \|A_i - \mathcal{B}_i \times \{U\}\|_F^2} \right)^{-1}. \quad (20)$$

Since the right hand side of Eq. (20) contains $U^{(n)}$ in $\{U\}$, we cannot solve Eq. (20) with respect to $U^{(n)}$ analytically. Instead, we solve Eq. (20) by an iterative algorithm. First, we initialize $U^{(n)}$ as $U^{(n,0)} = [v_1^{(n)}, \dots, v_{R_n}^{(n)}]$ where $v_1^{(n)}, \dots, v_{R_n}^{(n)}$ are the eigenvectors of $\sum_{i=1}^M A_{i(n)} A_{i(n)}^T$ corresponding to the largest R_n eigenvalues. Next, we update $U^{(n)}$ as

$$\tilde{U}^{(n,t+1)} = \sum_{i=1}^M A_{i(n)} \tilde{B}_{i(n,t)}^T e^{-\alpha \|A_i - \tilde{\mathcal{A}}_i^{(t)}\|_F^2} \left(\sum_{i=1}^M \tilde{B}_{i(n,t)} \tilde{B}_{i(n,t)}^T e^{-\alpha \|A_i - \tilde{\mathcal{A}}_i^{(t)}\|_F^2} \right)^{-1}, \quad (21)$$

where t is the number of iterations, $\tilde{\mathcal{A}}_i^{(t)} = \mathcal{B}_i^{(t)} \times \{U^{(t)}\}$ for $\mathcal{B}_i^{(t)} = \mathcal{A}_i \times \{U^{(t)T}\}$ and $U^{(t)} = \{U^{(1,t)}, \dots, U^{(N,t)}\}$, and $\tilde{B}_{i(n,t)}$ is the mode- n matricizing [12] or the matrix unfolding [10,11] of $\mathcal{B}_i^{(t)} \times_{-n} \{U^{(t)}\}$. Since Eq. (21) can be written as

$$\tilde{U}^{(n,t+1)} = U^{(n,t)} + \frac{1}{2\alpha} \frac{\partial \tilde{F}^{(t)}}{\partial U^{(n,t)}} \left(\sum_{i=1}^M \tilde{B}_{i(n,t)} \tilde{B}_{i(n,t)}^T e^{-\alpha \|A_i - \tilde{\mathcal{A}}_i^{(t)}\|_F^2} \right)^{-1}, \quad (22)$$

this iterative algorithm can be interpreted as a gradient method [14]. Lastly, we orthogonalize $\tilde{U}^{(n,t+1)}$ as $U^{(n,t+1)} = \text{orth}(\tilde{U}^{(n,t+1)})$, where $\text{orth}(\cdot)$ is an orthogonalization function, in order to satisfy the constraints in Eq. (2). The above procedure is repeated until it converges. The proposed iterative algorithm is summarized as follows:

[Robust SLRAT]

Step 0 (Initialization): Initialize $U^{(n)}$ for $n = 1, \dots, N$ as $U^{(n,0)} = [v_1^{(n)}, \dots, v_{R_n}^{(n)}]$ where $v_1^{(n)}, \dots, v_{R_n}^{(n)}$ are the eigenvectors of $\sum_{i=1}^M A_{i(n)} A_{i(n)}^T$ corresponding to the largest R_n eigenvalues. Initialize the iteration counter t as $t = 0$. Initialize the root mean squared error (RMSE) at $t = 0$ as $\text{RMSE}^{(0)} = \sqrt{\frac{1}{M} \sum_{i=1}^M \|A_i\|_F^2}$.

Step 1: Compute $\tilde{U}^{(n,t+1)}$ for $n = 1, \dots, N$ using Eq. (21).

Step 2: Orthogonalize $\tilde{U}^{(n,t+1)}$ for $n = 1, \dots, N$ as $U^{(n,t+1)} = \text{orth}(\tilde{U}^{(n,t+1)})$.

Compute $\mathcal{B}_i^{(t+1)} = \mathcal{A}_i \times \{U^{(t+1)}\}^T$ for $i = 1, \dots, M$.

Step 3: Compute the RMSE as

$$\text{RMSE}^{(t+1)} = \sqrt{\frac{1}{M} \sum_{i=1}^M \left\| \mathcal{A}_i - \tilde{\mathcal{A}}_i^{(t+1)} \right\|_F^2}. \quad (23)$$

If $(\text{RMSE}^{(t)} - \text{RMSE}^{(t+1)})/\text{RMSE}^{(t)} < \epsilon$ for $\epsilon > 0$ then proceed to the next step, otherwise increase t by 1 and go to Step 1.

Step 4: Output $U^* = U^{(t+1)}$ and $\mathcal{B}_i^* = \mathcal{B}_i^{(t+1)}$ for $i = 1, \dots, M$.

In our implementation, we used MATLAB orth function in Step 2 of the above procedure.

4 Subspace Matching for Classifying Sets of Tensors

In this section, we propose a method for classifying sets of tensors.

Let $\mathcal{A}_{i_c} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ for $i_c = 1, \dots, M_c$ and $c = 1, \dots, C$ be a set of tensors for training, where C is the number of classes. Then, for each class c , we solve the following optimization problem:

$$\min_{U_c, \{\mathcal{B}_{i_c}\}_{i=1}^{M_c}} \sum_{i_c=1}^{M_c} \rho(\|\mathcal{A}_{i_c} - \mathcal{B}_{i_c} \times \{U_c\}\|_F) \quad (24)$$

$$\text{subj.to } U_c^{(n)T} U_c^{(n)} = I_{R_n}, \quad n = 1, \dots, N, \quad (25)$$

where $U_c = \{U_c^{(1)}, \dots, U_c^{(N)}\}$ is a set of $U_c^{(n)} \in \mathbb{R}^{I_n \times R_n}$ for $n = 1, \dots, N$, and $\mathcal{B}_{i_c} \in \mathbb{R}^{R_1 \times \dots \times R_N}$. Let $U_c^* = \{U_{c*}^{(1)}, \dots, U_{c*}^{(N)}\}$ be the solution of Eq. (24) with (25). Then we store U_c^* for $c = 1, \dots, C$ and use them for classifying test sets of tensors to be classified.

Let $\mathcal{A}_i \in \mathbb{R}^{I_1 \times \dots \times I_N}$ for $i = 1, \dots, M$ be a set of tensors to be classified. Then we solve the optimization problem in Eq. (13) with Eq. (2). Let $U^* = \{U_*^{(1)}, \dots, U_*^{(N)}\}$ be the solution of the optimization problem. Then we classify the set of tensors into the c^* th class selected by the following rule:

$$c^* = \arg \max_{c \in \{1, \dots, C\}} S(U_c^*, U^*), \quad (26)$$

where $S(U_c^*, U^*)$ is a similarity between U_c^* and U^* , and is defined as follows:

$$S(U_c^*, U^*) = \max_{\{x_{i_n}\}, \{y_{i_n}\}} \sum_{n=1}^N \sum_{i_n=1}^{R_n} x_{i_n}^T \left| U_{c*}^{(n)T} U_*^{(n)} \right|_{\text{abs}} y_{i_n} \quad (27)$$

$$\text{subj.to } x_{i_n}^T x_{i'_n} = \delta_{i_n i'_n}, \quad y_{i_n}^T y_{i'_n} = \delta_{i_n i'_n}, \quad (28)$$

$$x_{i_n j_n} \in \{0, 1\}, \quad y_{i_n j_n} \in \{0, 1\}, \quad (29)$$

where $x_{i_n} = [x_{i_n j_n}] \in \mathbb{R}^{R_n}$ and $y_{i_n} = [y_{i_n j_n}] \in \mathbb{R}^{R_n}$ for $j_n = 1, \dots, R_n$, δ is the Kronecker delta, and $|A|_{\text{abs}}$ denotes the absolute value of a matrix A [15]. The optimization problem in Eq. (27), (28) and (29) is no less than the assignment problem. We compute the optimal solution using Munkres algorithm [16].

An advantage of the proposed classification method is that the proposed method can accept various numbers of tensors in each set for training or testing, because the sizes of U_c and U are independent of M_c and M , respectively.

5 Experimental Results

In this section, we experimentally evaluate the performance of the robust SLRAT on the ORL face image database. The ORL database [17] contains face images of 40 persons. For each person, there are 10 different images. That is, the total number of the images in the database is 400. The size of each image is 112×92 pixels, i.e., $I_1 = 112$, $I_2 = 92$.

The reconstruction errors for $R_1 = R_2 = 20$, $R_1 = R_2 = 30$ and $R_1 = R_2 = 40$ are shown in Fig. 2(a), (b) and (c), respectively.

5.1 Image Reconstruction

We first demonstrate the robustness of the proposed robust SLRAT using the ORL face images. Some example images are shown in Fig. 1, where the first three persons in the database are selected and presented in Fig. 1(a), (b) and (c), respectively. For each person, the original images are shown in the first row, in which a noise image is added (the rightmost image). The reconstructed images with the conventional and the robust SLRATs for $R_1 = R_2 = 30$ are shown in the second and the third rows for each person in Fig. 1(a), (b) and (c), respectively. The reconstructed images with the conventional SLRAT (the second rows) are disturbed by the added noise images. On the other hand, the reconstructed images with the robust SLRAT (the third rows) are less sensitive to the noise images than that of the conventional SLRAT.

We set $\alpha = 10^{-6}$ for the robust SLRAT and $\epsilon = 10^{-6}$ for both the conventional and the robust SLRATs.

The errors are measured by the RMSE for the face images except the noise images. In each figure, the horizontal axis denotes the number of noise images per person, and the vertical axis denotes the RMSE. The robust and the conventional SLRATs are denoted by solid lines with “+” marks and broken lines with “×” marks, respectively. The RMSEs for the robust SLRAT are lower than that for the conventional SLRAT. Although the RMSEs for the conventional SLRAT increase with the number of noise images, the RMSEs for the robust SLRAT are almost constant.

5.2 Face Recognition

We next show the experimental results of face recognition by the subspace matching method described in Sec. 4. From the ORL database [17], we select the first



(a) First person



(b) Second person



(c) Third person

Fig. 1. Example images: for each person, input images, reconstructed images with the conventional and the robust SLRATs are shown in the top, middle and bottom rows, respectively

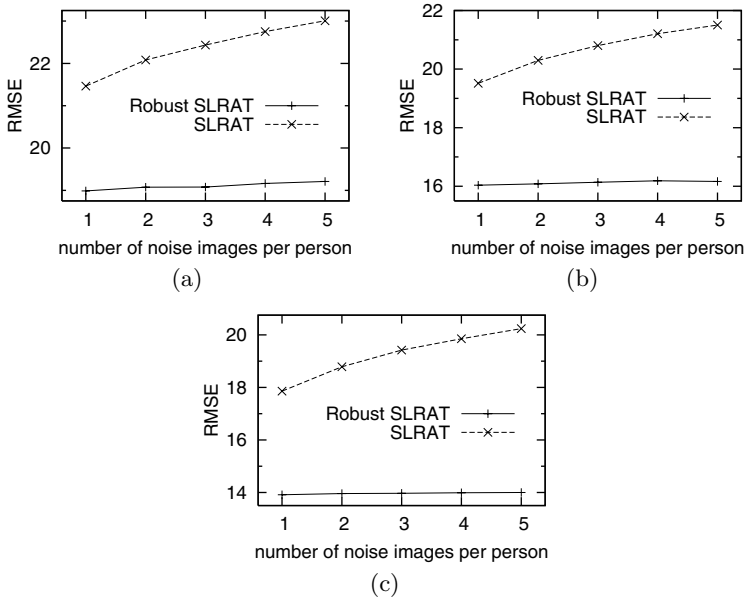


Fig. 2. Reconstruction errors: (a) $R_1 = R_2 = 20$, (b) $R_1 = R_2 = 30$, (c) $R_1 = R_2 = 40$

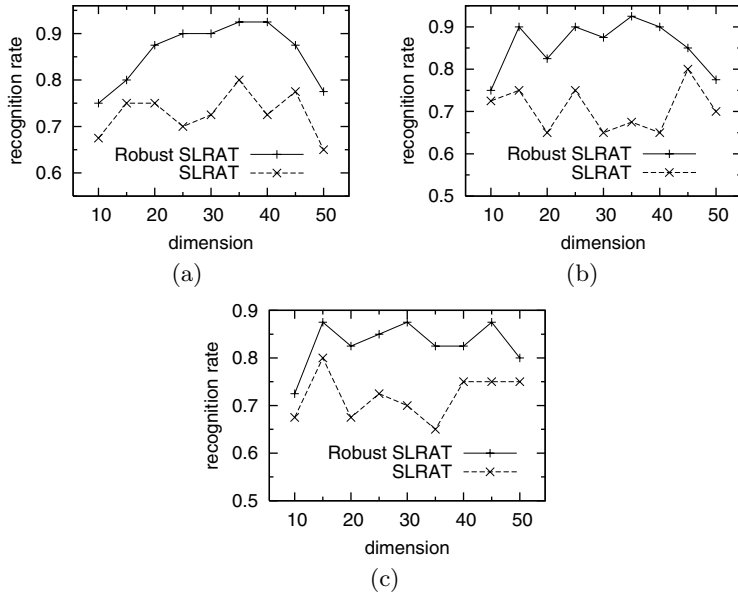


Fig. 3. Recognition rates: (a) $K = 1$, (b) $K = 2$, (c) $K = 3$

five images per person for training, and the remaining five images for testing. The number of classes coincides with that of persons in the database, i.e., $C = 40$. In the training phase, for each class c , we compute U_c^* for $c = 1, \dots, C$ and store them. In the testing phase, in order to verify the robustness of the proposed robust SLRAT, we include noise images in the set of test images. For each person, we add $K = 1, 2, 3$ noise images and then compute U^* . Each set of test images with several noise images is classified on the basis of the classification rule described in Eq. (26).

The recognition rates for $K = 1, 2$ and 3 are shown in Fig. 3(a), (b) and (c), respectively. In each figure, the horizontal axis denotes the reduced dimension R of each mode. We set R_1 and R_2 as $R_1 = R_2 = R$ for simplifying our experiments. The vertical axis denotes the recognition rate. The robust and the conventional SLRATs are denoted by solid lines with “+” marks and broken lines with “×” marks, respectively. The recognition rates for the robust SLRAT are higher than that for the conventional SLRAT.

6 Conclusion

In this paper, we have formulated simultaneous low rank approximation of tensors (SLRAT) as an optimization problem and modified it to the robust one. For both the SLRAT and the robust SLRAT, we proposed iterative algorithms for solving them. It is experimentally shown that the robust SLRAT achieves lower reconstruction errors than the conventional SLRAT. We also proposed a method for classifying sets of tensors, the subspace matching, where both training data and testing data are represented by their subspaces. The similarity between two subspaces are calculated by using Munkres algorithm [16] for assignment problems. The proposed classification method is applied to the example of face recognition on the ORL database [17]. It is experimentally verified that the robust SLRAT achieves higher recognition rates than the conventional SLRAT.

Acknowledgment

This work was partially supported by Grant-in-Aid for Young Scientists (B) No. 20700165.

References

1. Yang, J., Zhang, D., Frangi, A.F., Yang, J.: Two-dimensional PCA: A new approach to appearance-based face representation and recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 26, 131–137 (2004)
2. Wang, L., Wang, X., Zhang, X., Feng, J.: The equivalence of two-dimensional PCA to line-based PCA. *Pattern Recognition Letters* 26, 57–60 (2005)
3. Gao, Q.: Is two-dimensional PCA equivalent to a special case of modular PCA? *Pattern Recognition Letters* 28, 1250–1251 (2007)

4. Zhang, D., Chen, S., Liu, J.: Representing image matrices: Eigenimages versus eigenvectors. In: Wang, J., Liao, X.-F., Yi, Z. (eds.) ISNN 2005. LNCS, vol. 3497, pp. 659–664. Springer, Heidelberg (2005)
5. Ye, J.: Generalized low rank approximations of matrices. *Machine Learning* 61, 167–191 (2005)
6. Inoue, K., Urahama, K.: Equivalence of non-iterative algorithms for simultaneous low rank approximations of matrices. In: *IEEE Proc. CVPR*, pp. 154–159 (2006)
7. Ding, C., Huang, H., Luo, D.: Tensor Reduction Error Analysis – Applications to Video Compression and Classification. In: *Proc. CVPR* (2008)
8. Lu, H., Plataniotis, K.N., Venetsanopoulos, A.N.: MPCA: Multilinear principal component analysis of tensor objects. *IEEE Trans. Pattern Anal. Mach. Intell.* 19, 18–39 (2008)
9. Huang, H., Ding, C.: Robust Tensor Factorization Using R1-Norm. In: *Proc. CVPR* (2008)
10. De Lathauwer, L., De Moor, B., Vandewalle, J.: A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.* 21, 1253–1278 (2000)
11. De Lathauwer, L., De Moor, B., Vandewalle, J.: On the best rank-1 and rank- (R_1, R_2, \dots, R_N) approximation of higher-order tensors. *SIAM J. Matrix Anal. Appl.* 21, 1324–1342 (2000)
12. Bader, B.W., Kolda, T.G.: Algorithm 862: MATLAB tensor classes for fast algorithm prototyping. *ACM Trans. Math. Software* 32, 635–653 (2006)
13. Huber, P.J.: *Robust Statistics*. Wiley, Chichester (1981)
14. Ortega, J.M., Rheinboldt, W.G.: *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, NY (1970)
15. Lütkepohl, H.: *Handbook of Matrices*. John Wiley & Sons, Chichester (1996)
16. Munkres, J.: Algorithms for the Assignment and Transportation Problems. *Journal of the Society of Industrial and Applied Mathematics* 5, 32–38 (1957)
17. Samaria, F., Harter, A.: Parameterisation of a stochastic model for human face identification. In: *Proc. 2nd IEEE Workshop on Appl. Comput. Vision* (1994)