Generalized *N*-Dimensional Principal Component Analysis and Efficient Representation of Medical Volumes

Yen-Wei CHEN † and Rui XU †

† College of Information Science and Eng., Ritsumeikan Univ. 1-1-1 Nojihigashi, Kusatsu, Shiga, 525-8577 Japan
E-mail: † chen@is.ritsumei.ac.jp

Abstract There is a growing interest in multi-dimensional image processing, such as medical volume image processing, hyperspectral image processing. In this paper, we propose a novel approach called generalized N-dimensional principal component analysis (GND-PCA) for efficient multi-dimensional data representation and modeling. In GND-PCA, the multi-dimensional data is treated as a tensor. The optimal subspaces on each mode are simultaneously calculated by minimizing the square error between the original tensor and the reconstructed tensor based on the subspace. Experiments on medical MRI dataset show that the proposed GND-PCA can represent the multi-dimensional data more efficiently compared to conventional PCA and recently proposed ND-PCA.

Keyword *N*-Dimensional Principal Component Analysis , Generalized, Tensor , High-order SVD , Efficient Representation, Medical Volume

1. Introduction

Principal component analysis (PCA) is an important technique for efficient data representation and modeling. PCA is an orthogonal linear transform that projects the data into a new coordinate system (subspace) with bases where the data varies the most. The bases are determined by the eigenvectors of the covariance matrix corresponding to the largest eigenvalues. The magnitude of the eigenvalues corresponds to the variance of the data along the eigenvector bases [1, 2]. Since the classical PCA is the method for 1-dimensional (1D) vector data, when PCA is applied to multi-dimensional data (e.g. 2D image or 3D volume), the multi-dimensional data should be initially unfolded to a long 1D vector. Such unfolding process will introduce several problems: (1) the feature vector is in high dimensional vector space resulting in huge computation cost and bad performance on generalization; (2) lost of spatial information. Yang et al proposed a new method called 2-dimensional principal component analysis (2D-PCA) to overcome the above problems [3]. This method is to calculate the bases in the column-mode subspace of the 2D image instead of finding the basis in the long unfolding vector subspace. Therefore, the 2D data can be directly used in the training without the unfolding vector preprocessing. 2D-PCA not only makes the calculation of the bases efficiently but also can accurately represent the 2D data, however its drawback is that it needs more coefficients to represent the 2D data than PCA because 2D PCA is a unilateral projection (right multiplication) scheme. Kong et al. proposed a generalized 2-dimensional PCA (G2D-PCA) [4], which is a bilateral projection scheme, to simultaneously calculate the basis of the row- and column-mode subspaces, so it can represent the 2D data not only accurately but efficiently. Recently, inspired from the work of 2D-PCA, a method called N-dimensional PCA (ND-PCA) was proposed for higher-dimensional data representation [5]. In ND-PCA, the higher-dimensional data is treated as the higher-order tensor witch is directly trained to obtain the bases on one-mode subspace by multi-linear algebra based tool called higher-order singular value decomposition (HOSVD) [6]. It was applied on the 3D facial scanning data. Since ND-PCA only compresses the data on one-mode subspace, it is also suffered from the problem that the data can not represented efficiently, similar to the problem of 2D-PCA.

Inspired from the works of G2D-PCA and ND-PCA, we proposed a new method called generalized N-dimensional principal component analysis (GND-PCA). The high–dimensional data is treated as a series of higher-order tensors and the optimal subspace on each mode are simultaneously calculated by minimizing the square error between the original tensor and the reconstructed tensor based on the subspace with an iteration algorithm. Experiments on medical MRI dataset show that the proposed GND-PCA can represent the

multi-dimensional data more efficiently compared to the conventional PCA and ND-PCA.

The paper is organized as follows: related works such as 2D-PCA, G2D-PCA and ND-PCA are briefly summarized in Sec.2; the proposed GND-PCA is presented in Sec.3; experimental results on medical MRI dataset and multi-angle view & illumilation facial dataset are presented in Sec.4; and conclusions are given in Sec.5.

2. Related Works

2.1. SVD and PCA

Suppose a series of D-dimensional vectors with zero-mean, $\mathbf{a}_i, i=1,2,\cdots,M$ are given and $\mathbf{A}=[\mathbf{a}_1,\mathbf{a}_2,\cdots,\mathbf{a}_M]$ is a $D\times M$ matrix, where M is the number of samples. PCA is based on eigenvalue decomposition of the covariance matrix $\mathbf{Cov}$, as Eq.(1).

$$\mathbf{Cov}=\mathbf{A}\mathbf{A}^T=\mathbf{W}\boldsymbol{\Sigma}\mathbf{W}^T \qquad (1)$$

where $\boldsymbol{\Sigma}$ is a diagonal matrix corresponding of eigen values and $\mathbf{W}$ is a $D\times D$ matrix, whose column vectors $\mathbf{w}_i, i=1,2,\cdots,D$ are eigenvectors of $\mathbf{Cov}$. The leading J eigenvectors, where $J\le D$ construct the subspace and the vector $\mathbf{a}\in\mathbf{R}^D$ can be represented by its coefficient vector, $\mathbf{b}=[\mathbf{w}_1,\mathbf{w}_2,\cdots,\mathbf{w}_J]^T\cdot\mathbf{a}\in\mathbf{R}^J$.

On the other hand, a matrix $\mathbf{A}\in\mathbf{R}^{D\times M}$ can be decomposed by SVD as

$$\mathbf{A}=\mathbf{U}\mathbf{S}\mathbf{V}^T \qquad (2)$$

where $\mathbf{U}\in\mathbf{R}^{D\times D}$ and $\mathbf{V}\in\mathbf{R}^{M\times M}$ are orthogonal matrices. $\mathbf{S}\in\mathbf{R}^{D\times M}$ is a diagonal matrix corresponding of singular values. From Eq.(2),

$$\mathbf{A}\mathbf{A}^T=\mathbf{U}\mathbf{S}\mathbf{V}^T\mathbf{V}\mathbf{S}^T\mathbf{U}^T=\mathbf{U}\mathbf{S}^2\mathbf{U}^T \qquad (3)$$

It is clear that the squares of the non-zero singular values of $\mathbf{A}$ are equal to the non-zero eigenvalues of $\mathbf{A}\mathbf{A}^T=\mathbf{Cov}$ and the columns of $\mathbf{U}$ (left singular vectors) are eigenvectors of $\mathbf{Cov}$. Thus we just need to apply SVD to $\mathbf{A}$ to get the principal orthogonal vectors (bases).

2.2. HO-SVD and ND-PCA

A N-th order tensor $\mathcal{A}\in\mathbf{R}^{I_1\times I_2\times\cdots\times I_N}$ is defined as a multi-array with N indices. The space of the tensor is comprised by the N mode space. The tensor $\mathcal{A}$ can be unfolded to $\mathbf{A}_{(n)}\in\mathbf{R}^{I_n\times(I_1\cdot I_2\cdots I_{n-1}\cdot I_{n+1}\cdots I_N)}$ which is called as mode-n matrix, where $0<n\le N$. Unfolding of a 3rd order tensor is shown in Fig.1. The tensor $\mathcal{A}$ can be decomposed by the higher-order SVD (HO-SVD), which is also known as Tucker decomposition [6], as

$$\mathcal{A}=\mathcal{B}\times_1\mathbf{U}^{(1)}\times_2\mathbf{U}^{(2)}\times\cdots\times_N\mathbf{U}^{(N)} \qquad (4)$$

where $\mathbf{U}^{(n)}\in\mathbf{R}^{I_n\times I_n}$ is a unitary matrix, $\mathcal{B}\in\mathbf{R}^{I_1\times I_2\times\cdots\times I_N}$ is the core tensor, and $\times_n$ is the mode-n product. As shown in Eq.(5), $\mathcal{B}\times_n\mathbf{U}^{(n)}$ can be rewritten as a matrix multiplication and the result C is also a N-th order tensor.

$$\mathcal{B}\times_n\mathbf{U}^{(n)}=\mathbf{U}^{(n)}\mathbf{B}_{(n)}=\mathbf{C}_{(n)}=C \qquad (5)$$

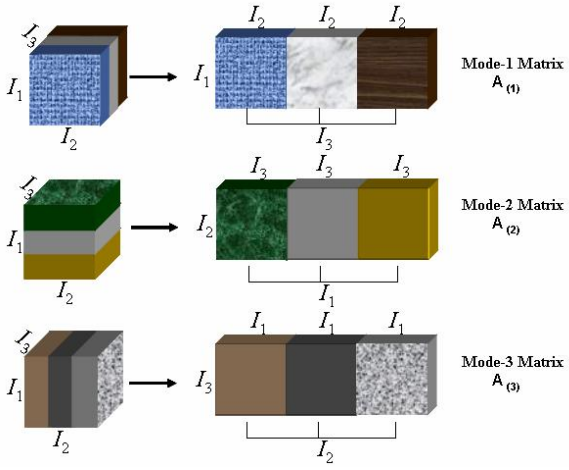

Fig.1 Example of unfolding the 3rd order tensor

HO-SVD has been applied to ND-PCA[5] with applications to 3D facial scanning data, representation of face with multiple-modes [7,8] and robust face recognition[9]. In ND-PCA, the N-dimensional dataset are directly treated as N-th order tensors $\mathcal{A}_i\in\mathbf{R}^{I_1\times I_2\times\cdots\times I_N}$, $i=1,2,\cdots,M$. In a similar manner described in Sec.2.1, instead of calculating the covariance tensor, we just need to construct a new tensor $\mathcal{X}=[\mathcal{A}_1,\mathcal{A}_2,\cdots,\mathcal{A}_M]\in\mathbf{R}^{I_1\times I_2\times\cdots\times I_N\times M}$, and apply HO-SVD on its mode-n subspace. The first leading J eigenvectors $\mathbf{u}_1^{(n)},\mathbf{u}_2^{(n)},\cdots,\mathbf{u}_J^{(n)}$, where $J<I_n$, are the bases on the mode-n subspace $\mathbf{U}^{(n)}$. The N-dimensional data $\mathcal{A}\in\mathbf{R}^{I_1\times I_2\times\cdots\times I_N}$ can be compactly represented by a tensor $\mathcal{B}=\mathcal{A}\times_n\mathbf{U}^{(n)T}\in\mathbf{R}^{I_1\times I_2\times\cdots\times I_{n-1}\times J\times I_{I+1}\times\cdots\times I_N}$, whose components are the projections (coefficients) onto the mode-n subspace.

3. Generalized ND-PCA

Though ND-PCA can make the calculation of the bases efficiently and can accurately represent the multi-dimensional data. As well as 2D-PCA, ND-PCA is also a unilateral projection scheme and only compress the data on the mode-*n* subspace. So ND-PCA needs lots of components to represent the multi-dimensional data. In this paper, we propose a generalized *N*-dimensional PCA (GND-PCA) to simultaneously calculate the basis on each mode subspace [10].

The basic idea of GND-PCA is that we want to reconstruct the original *N*-th order tensor $\mathcal{A} \in \mathbf{R}^{I_1 \times I_2 \times \cdots \times I_N}$ with a lower rank core tensor $\mathcal{B} \in \mathbf{R}^{J_1 \times J_2 \times \cdots \times J_N}$, where $J_n < I_n$, and try to find a set of optimal matrices $\mathbf{U}^{(n)} \in \mathbf{R}^{I_n \times J_N}$, $n = 1,2,\cdots,N$ with orthogonal column for each mode. The reconstruction of *N*-th order tensor $\mathcal{A}$ can be expressed as $\hat{\mathcal{A}} = \mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times \cdots \times_N \mathbf{U}^{(N)}$. Illustration for 3rd order tensor reconstruction is shown in Fig.2.

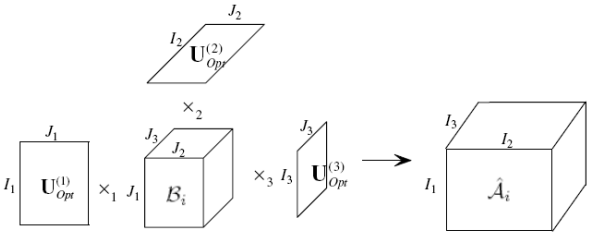

Fig.2 Illustration of reconstructing a 3rd order tensor

The optimal orthogonal matrices $\mathbf{U}^{(n)}$ can be determined by minimizing a cost function as Eq.(6).

$$S = \sum_{i=1}^{M} \left\| \mathcal{A}_i - \hat{\mathcal{A}}_i \right\|^2 = \sum_{i=1}^{M} \left\| \mathcal{A}_i - \mathcal{B}_i \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times \cdots \times_N \mathbf{U}^{(N)} \right\|^2 \quad (6)$$

In Eq.(6), only the samples $\mathcal{A}_i \in \mathbf{R}^{I_1 \times I_2 \times \cdots \times I_N}$, ($i = 1,2,\cdots,M$) are known, *M* is the number of samples.

Theorem 1: *Given fixed N matrices* $\boldsymbol{U}^{(n)}$*, the tensors* $\mathcal{B}_i$ *that minimize the cost function of.(6) are given by*

$$\mathcal{B}_i = \mathcal{A}_i \times_1 \mathbf{U}^{(1)^T} \times_2 \mathbf{U}^{(2)^T} \times \cdots \times_N \mathbf{U}^{(N)^T} \quad (7)$$

Since the proof of Theory 1 is simple, it is omitted here. From Theorem 1, we can obtain Theorem 2 [10].

Theorem 2: *If the tensors* $\mathcal{B}_i$ *are given as Eq.(7), minimization of the cost function of Rq.(6) is equal to maximization of the following cost function:*

$$S' = \sum_{i=1}^{M} \left\| \mathcal{A}_i \times_1 \mathbf{U}^{(1)^T} \times_2 \mathbf{U}^{(2)^T} \times \cdots \times_N \mathbf{U}^{(N)^T} \right\|^2 \quad (8)$$

There is no close-form solution to simultaneously resolve the matrices $\mathbf{U}^{(n)}$ for the cost function S', however the explicit solution for one matrix can be obtained if the other matrices are fixed [10]. So we use an iteration algorithm to simultaneously calculate the optimal matrices $\mathbf{U}_{opt}^{(1)}, \mathbf{U}_{opt}^{(2)}, \cdots, \mathbf{U}_{opt}^{(N)}$, which are able to maximize the cost function S'. This algorithm is summarized in Algorithm 1. In Algorithm 1, we terminate the iteration when the cost of Eq.(8) is not significantly changed in two consecutive times.

Algorithm 1 Iteration Algorithm to Compute the N Matrices $\mathbf{U}_{opt}^{(1)}, \mathbf{U}_{opt}^{(2)}, \ldots, \mathbf{U}_{opt}^{(N)}$

IN: a series of N-th order tensors, $\mathcal{A}_i \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}, i = 1,2,\ldots,M$.
OUT: N Matrices $\mathbf{U}_{opt}^{(n)} \in \mathbb{R}^{I_n \times J_n}$ $(J_n < I_n, n = 1,2,\ldots,N)$ with orthogonal column vectors.

1. Initial values: $k = 0$ and $\mathbf{U}_0^{(n)}$ whose columns are determined as the first J_n leading eigenvectors of the matrices $\sum_{i=1}^{M} (\mathbf{A}_{i(n)} \cdot \mathbf{A}_{i(n)}^T)$.
2. Iterate for k until convergence
   - Maximize $S' = \sum_{i=1}^{M} \| \mathcal{C}_i \times_1 \mathbf{U}^{(1)^T} \|^2, \mathcal{C}_i = \mathcal{A}_i \times_2 \mathbf{U}_k^{(2)^T} \times \ldots \times_N \mathbf{U}_k^{(N)^T}$
     Solution: $\mathbf{U}^{(1)}$ whose columns are determined as the first J_1 leading eigenvectors of $\sum_{i=1}^{M} (\mathbf{C}_{i(1)} \cdot \mathbf{C}_{i(1)}^T)$
     Set $\mathbf{U}_{k+1}^{(1)} = \mathbf{U}^{(1)}$.
   - Maximize $S' = \sum_{i=1}^{M} \| \mathcal{C}_i \times_2 \mathbf{U}^{(2)^T} \|^2, \mathcal{C}_i = \mathcal{A}_i \times_1 \mathbf{U}_{k+1}^{(1)^T} \times_3 \mathbf{U}_k^{(3)^T} \times \ldots \times_N \mathbf{U}_k^{(N)^T}$
     Solution: $\mathbf{U}^{(2)}$ whose columns are determined as the first J_2 leading eigenvectors of $\sum_{i=1}^{M} (\mathbf{C}_{i(2)} \cdot \mathbf{C}_{i(2)}^T)$
     Set $\mathbf{U}_{k+1}^{(2)} = \mathbf{U}^{(2)}$.
     ...
   - Maximize $S' = \sum_{i=1}^{M} \| \mathcal{C}_i \times_n \mathbf{U}^{(n)^T} \|^2, \mathcal{C}_i = \mathcal{A}_i \times_1 \mathbf{U}_{k+1}^{(1)^T} \times \ldots \times_{n-1} \mathbf{U}_{k+1}^{(n-1)^T} \times_{n+1} \mathbf{U}_k^{(n+1)^T} \times \ldots \times_N \mathbf{U}_k^{(N)^T}$
     Solution: $\mathbf{U}^{(n)}$ whose columns are determined as the first J_n leading eigenvectors of $\sum_{i=1}^{M} (\mathbf{C}_{i(n)} \cdot \mathbf{C}_{i(n)}^T)$
     Set $\mathbf{U}_{k+1}^{(n)} = \mathbf{U}^{(n)}$.
     ...
   - Maximize $S' = \sum_{i=1}^{M} \| \mathcal{C}_i \times_N \mathbf{U}^{(N)^T} \|^2, \mathcal{C}_i = \mathcal{A}_i \times_1 \mathbf{U}_{k+1}^{(1)^T} \times \ldots \times_{N-1} \mathbf{U}_{k+1}^{(N-1)^T}$
     Solution: $\mathbf{U}^{(N)}$ whose columns are determined as the first J_N leading eigenvectors of $\sum_{i=1}^{M} (\mathbf{C}_{i(N)} \cdot \mathbf{C}_{i(N)}^T)$
     Set $\mathbf{U}_{k+1}^{(N)} = \mathbf{U}^{(N)}$.

   $k = k + 1$
3. Set $\mathbf{U}_{opt}^{(1)} = \mathbf{U}_k^{(1)}$, $\mathbf{U}_{opt}^{(2)} = \mathbf{U}_k^{(2)}$, …, $\mathbf{U}_{opt}^{(N)} = \mathbf{U}_k^{(N)}$.

4. Experimental Results

The proposed GND-PCA is applied to medical MR volumes. We use eighteen MR T1-weighted 3D images (volumes) of Vanderbilt database [11]. These eighteen volumes are collected from different patients, and their dimensions are $256 \times 256 \times 26$. We choose one volume as the template and align the other seventeen volumes onto the template by

similarity-transformation based rigid registration. A 3D similarity-transformation has seven parameters, three for translations, three for rotation angles and one for scaling factor [10]. Such a registration can eliminate global difference but keep the local differences for the modeling. Three registered MR volumes are shown in Fig.3, which are used as training samples.

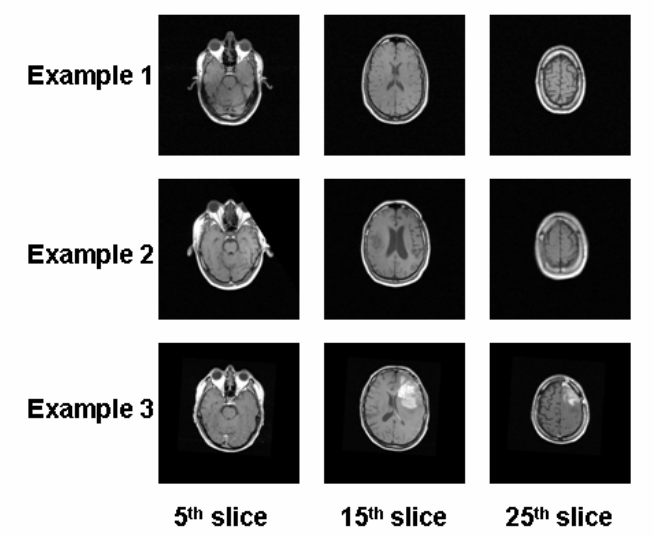

Fig.3 Examples of the registered MR volumes

The medical volumes are treated as a series of the 3rd order tensors. The leave-one-out experiment is done to test the generalization ability of GND-PCA. We use seventeen volumes as samples to learn the optimal subspaces and the left-untrained one is used as a test. In training process, the iteration is terminated when there is no dramatic change of the cost function in two consecutive times. The convergence of the training for $50\times50\times15$ mode-subspace bases is shown in Fig.4. It can be seen that the convergence is fast. Usually two times of iteration is enough.

One typical result is shown in Fig.5. The test volume is reconstructed from $50\times50\times15$ and $75\times75\times20$ mode-subspace bases, respectively. The corresponding compressing rates are 2.2% and 6.6%, respectively. It can be seen that the quality of the reconstructed images become better and better as increasing the mode-subspace bases, especially for the tumor region (the bright region in lower right). It should be noted that the training samples do not have similar tumors around that position.

In order to make a comparison, the same experiments

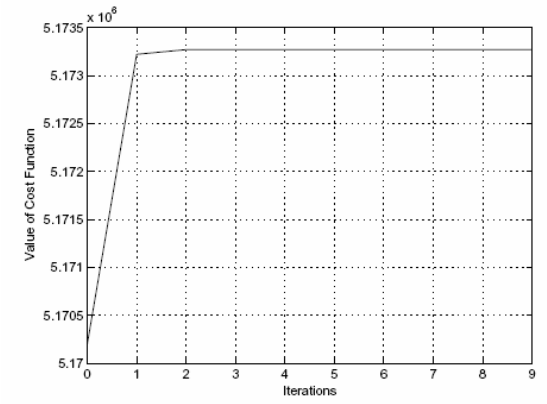

Fig.4 Convergence of GND-PCA

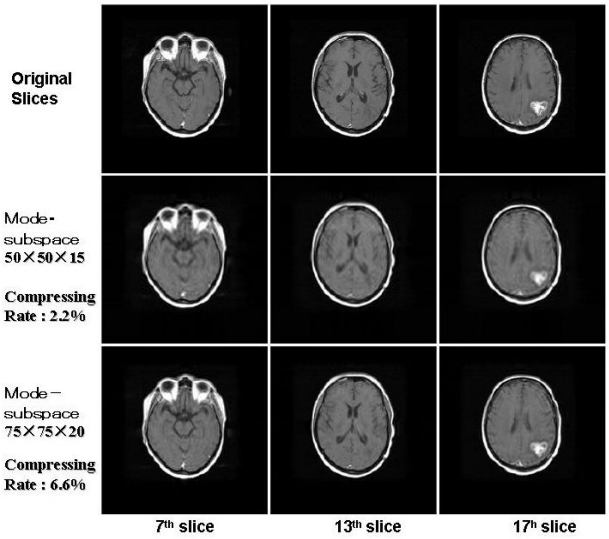

Fig.5 Reconstructed results with GND-PCA bases

are also done with classical PCA (1D-PAC) and ND-PCA. In classical 1D-PAC, the volume image should be first unfolded into a vector with a huge dimension of 1703936. So only the eigenface method [2] can be used to calculate the PCA subspace. The reconstructed result is shown in Fig.6. Since in the eigenface method, only 16 bases are available which are too few compared to the dimension of 1703936, the test volume can not be reconstructed well as shown in Fig.6. So it is clear that if the samples for training are limited, the classical PCA can not used for modeling or efficient representation of the multi-dimensional data because of its bad performance on generalization.

The reconstructed volumes by ND-PCA[5] in the leave-one-out testing experiments are shown in Fig.7. The compression rate is about 11.7%, which is corresponding to $100\times100\times20$ in GND-PCA. It can be seen that the results of ND-PCA are better than the results of classical 1D-PCA (Fig.6). But they are more blurred compared to the results of our GND-PCA (the case of $75\times75\times20$ in Fig.5).

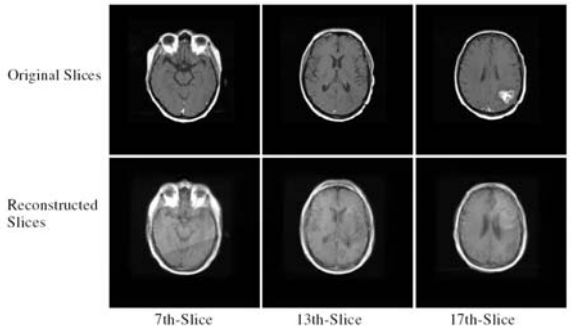

Fig.6 Reconstructed results with classical 1D-PCA bases

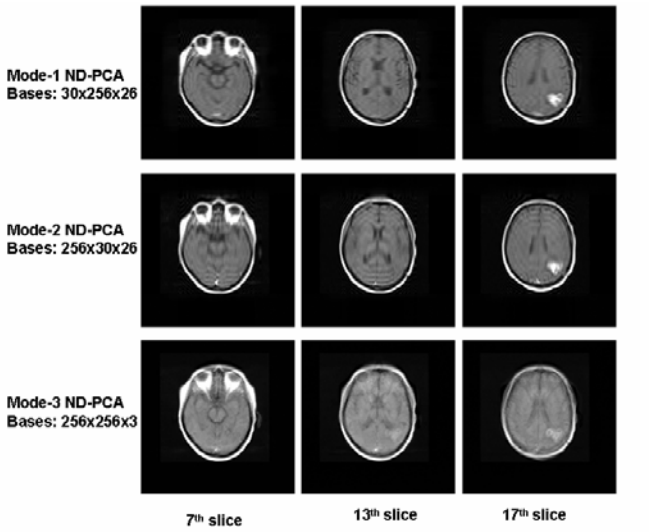

Fig.7 Reconstructed results with ND-PCA basis

In order to make a quantitative comparison, normalized correlation (*NC*) of the original volume $I(x, y, z)$ and the reconstructed volume $\hat{I}(x, y, z)$, which is defined as Eq.(9) and is used as a quantitative measure, are shown in Fig.8. The compressing rate in each method is the same (11.7%). It can be seen that the normalized correlation for GND-PCA is higher than ND-PCA and conventional PCA.

$$NC = \frac{\sum_{x,y,z} I(x, y, z) \cdot \hat{I}(x, y, z)}{\sqrt{\sum_{x,y,z} I^2(x, y, z)} \cdot \sqrt{\sum_{x,y,z} \hat{I}^2(x, y, z)}} \quad (9)$$

5. Conclusion

We proposed a novel approach called generalized *N*-dimensional principal component analysis (GND-PCA) for efficient multi-dimensional data representation and modeling. ND-PCA can be considered as a special case of GND-PCA. The effectiveness and representation ability of GND-PCA have been demonstrated by experiments on medical MR volume dataset.

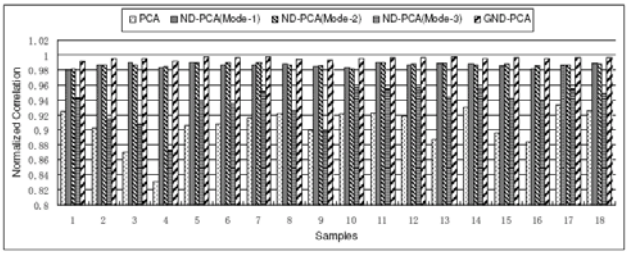

Fig.8 Comparison of results for PCA, ND-PCA, GND-PCA

This work was supported in part by the Grand-in Aid for Scientific Research from the Japanese Ministry for Education, Science, Culture and Sports under the Grand No. 19500161.

References

[1] I.T. Jolliffe, Principal Component Analysis. Springer, 2002.

[2] M. Turk, A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, Vol.3, No.1, pp. 71-86, 1991.

[3] J. Yang, et al., "To-Dimensional PCA: A New Approach to Appearance-based Face Representation and Recognition", IEEE Trans.on PAMI, Vol.26, No.1, pp. 131-137, 2004.

[4] H. Kong, et. Al., "Generalized 2D principal component analysis for face image representation and recognition ",Neural Networks, Vol.18, pp.585-594, 2005.

[5] H.C.Yu and M.Bennamoun, "1D-PCA,2D-PCA to nD-PCA", Proc. of ICPR'06,2006.

[6] L.D.Lathauwer, B.D.Moor and J.Vandewalle, "A Multilinear Singular Value Decomposition", SIAM Journal of Matrix Analysis and Application, Vol.21, No.4, pp.1253-1278, 2001.

[7] M.A.O.Vasilescu and D.Terzopoulos, "Multilinear Analysis of Image Ensembles: Tensorface", Proc. of European Conference on Computer Vision, pp.447-460,2002.

[8] M.A.O.Vasilescu and D.Terzopoulos, "Multilinear Subspace Analysis of Image Ensemble", Proc. of IEEE Conf. On Computer Vision and Pattern Recognition, Vol.2, pp.93-99, 2003.

[9] M.A.O.Vasilescu and D.Terzopoulos, "Multilinear Image Analysis for Facial Recognition", Proc. of ICPR'02, 2002

[10] R.Xu and Y.W.Chen, "Appearance models for medical volumes with few samples by generalized 3D-PCA", Lecture Notes in Computer Science, Springer, Vol.4984, pp.821-830, 2008.

[11] J. West et al., "Comparison and evaluation of retrospective intermodality brain image registration techniques," J. Comput. Assist. Tomogr., Vol.21, pp.554-566, 1997.