NON-NEGATIVE TENSOR FACTORIZATION USING ALPHA AND BETA DIVERGENCES

Andrzej CICHOCKI[1*], *Rafal ZDUNEK*[1†], *Seungjin CHOI*[2], *Robert PLEMMONS*[3] *and Shun-ichi AMARI*[1]

[1] Brain Science Institute, RIKEN, Wako-shi, Saitama 351-0198, JAPAN,
[2] Pohang University of Science and Technology, KOREA,
[3] Wake Forest University, USA

ABSTRACT

In this paper we propose new algorithms for 3D tensor decomposition/factorization with many potential applications, especially in multi-way Blind Source Separation (BSS), multidimensional data analysis, and sparse signal/image representations. We derive, compare and implement in MATLAB NTFLAB Toolbox three classes of algorithms: Multiplicative, Fixed Point Alternating Least Squares (FPALS) and Alternating Interior-Point Gradient (AIPG) algorithms. Some of the proposed algorithms are characterized by improved robustness, efficiency and convergence rates and can be applied for various distributions of data and additive noise.

Index Terms— Algorithms, Learning systems, Linear approximation, Signal representations, Feature extraction.

1. MODELS AND PROBLEM FORMULATION

Tensors (also known as n-way arrays or multidimensional arrays) are used in a variety of applications ranging from neuroscience and psychometrics to chemometrics [6,8,9,17-19]. Nonnegative matrix factorization (NMF), Non-negative tensor factorization (NTF) and parallel factor analysis PARAFAC models with non-negativity constraints have been recently proposed as promising sparse and quite efficient representations of signals, images, or general data [2-7,10-13]. From a viewpoint of data analysis, NTF is very attractive because it takes into account spacial and temporal correlations between variables more accurately than 2D matrix factorizations, such as NMF, and it provides usually sparse common factors or hidden (latent) components with physiological meaning and interpretation [9,15]. In most applications, especially in neuroscience (EEG, fMRI), the standard NTF or PARAFAC models were used [15,16]. In this paper we consider more general model referred to as 3D NTF2 model (in analogy to the Parafac2 model [17]) (see Fig. 1). A given tensor $\underline{\boldsymbol{X}} \in \mathbb{R}_+^{I\times T\times K}$ is decomposed to a set of matrices $\boldsymbol{S}$, $\boldsymbol{D}$ and $\{\boldsymbol{A}_1, \boldsymbol{A}_2, ..., \boldsymbol{A}_K\}$ with nonnegative entries. Here and elsewhere, $\mathbb{R}_+$ denotes the nonnegative orthant with appropriate dimensions. The three-way NTF2 model can be described as

$$\boldsymbol{X}_k = \boldsymbol{A}_k \boldsymbol{D}_k \boldsymbol{S} + \boldsymbol{E}_k, \qquad (k = 1, 2, \ldots, K) \tag{1}$$

where $\boldsymbol{X}_k = \boldsymbol{X}_{:,:,k} = [x_{itk}]_{I\times T} \in \mathbb{R}_+^{I\times T}$ are frontal slices of $\underline{\boldsymbol{X}} \in \mathbb{R}_+^{I\times T\times K}$, K is the number of frontal slices, $\boldsymbol{A}_k = [a_{irk}]_{I\times R} \in \mathbb{R}_+^{I\times R}$ are the basis (mixing matrices), $\boldsymbol{D}_k \in \mathbb{R}_+^{R\times R}$ is a diagonal matrix that holds the k-th row of the $\boldsymbol{D} \in \mathbb{R}_+^{K\times R}$ in its main diagonal, and $\boldsymbol{S} = [s_{rt}]_{R\times T} \in \mathbb{R}_+^{R\times T}$ is a matrix representing sources (or hidden components or common factors), and $\boldsymbol{E}_k = \boldsymbol{E}_{:,:,k} \in \mathbb{R}^{I\times T}$ is the k-th frontal slice of a tensor $\underline{\boldsymbol{E}} \in \mathbb{R}^{I\times T\times K}$ representing error or noise depending upon the application. The objective is to estimate the set of matrices $\{\boldsymbol{A}_k\}, (k, \ldots, K)$, $\boldsymbol{D}$ and $\boldsymbol{S}$, subject to some non-negativity constraints and other possible natural constraints such as sparseness and/or smoothness. Since the diagonal matrices $\boldsymbol{D}_k$ are scaling matrices they can usually be absorbed by the matrices $\boldsymbol{A}_k$ by introducing column-normalized matrices $\boldsymbol{A}_k := \boldsymbol{A}_k \boldsymbol{D}_k$, so usually in BSS applications the matrix $\boldsymbol{S}$ and the set of scaled matrices $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_K$ need only to be estimated. However, in such a case we may loose the uniqueness of the NTF representation ignoring scaling and permutation ambiguities. The uniqueness still can be achieved by imposing nonnegativity, sparsity and other constraints. The above NTF2 model is similar to the well known PARAFAC2 model with non-negativity constraints and Tucker models [6,15,17]. In the special case, when all matrices $\boldsymbol{A}_k$ are identical, the NTF2 model can be simplified to the ordinary PARAFAC model with the non-negativity constraints described $\boldsymbol{X}_k = \boldsymbol{A}\boldsymbol{D}_k\boldsymbol{S} + \boldsymbol{E}_k,\ k = 1, \ldots, K$ or equivalently $x_{itk} = \sum_r a_{ir} s_{rt} d_{kr} + e_{itk}$ or $\underline{\boldsymbol{X}} = \sum_r \boldsymbol{a}_r \otimes \boldsymbol{s}_r^T \otimes \boldsymbol{d}_r + \underline{\boldsymbol{E}}$, where $\boldsymbol{s}_r$ are rows of $\boldsymbol{S}$ and $\boldsymbol{a}_r$, $\boldsymbol{d}_r$ are columns of $\boldsymbol{A}$ and $\boldsymbol{D}$, respectively and $\otimes$ means outer product of vectors [8,9]. Throughout this paper, we use the following notation: the rt-th element of the matrix $\boldsymbol{S}$ is denoted by s_{rt}, $x_{itk} = [\boldsymbol{X}_k]_{it}$ means the it-th element of the k-th frontal slice matrix $\boldsymbol{X}_k$, $\bar{\boldsymbol{A}} = [\boldsymbol{A}_1; \boldsymbol{A}_2; \ldots; \boldsymbol{A}_K] \in \mathbb{R}_+^{KI\times R}$ is a column-wise unfolded matrix of the slices $\boldsymbol{A}_k$, $\bar{a}_{pr} = [\bar{\boldsymbol{A}}]_{pr}$. Analogously, $\bar{\boldsymbol{X}} = [\boldsymbol{X}_1; \boldsymbol{X}_2; \ldots; \boldsymbol{X}_K] \in \mathbb{R}_+^{KI\times T}$ is the column-wise unfolded matrix of the slices $\boldsymbol{X}_k$ and $\bar{x}_{pt} = [\bar{\boldsymbol{X}}]_{pt}$.

[*] On leave from Warsaw University of Technology, Poland

[†] On the leave from Institute of Telecommunications, Teleinformatics, and Acoustics, Wroclaw University of Technology, Poland

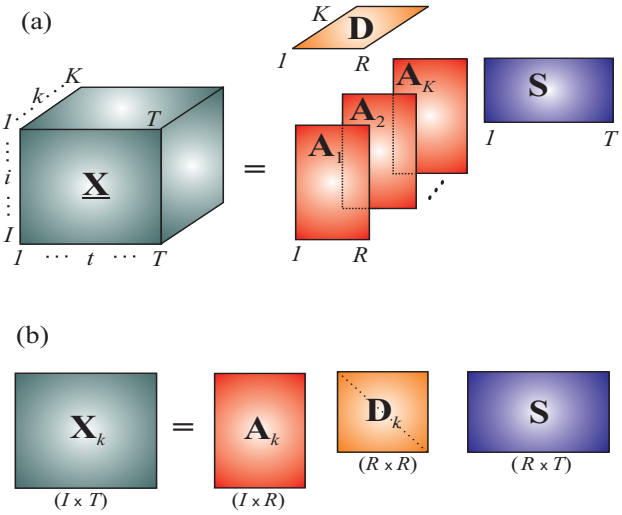

Fig. 1. (a) NTF2 model in which 3D tensor is decomposed to set of nonnegative matrices: $\{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_K\}$, $\boldsymbol{D}$, $\boldsymbol{S}$. (b) Equivalent representation in which frontal slices of tensor are factored by set of matrices (tensor $\underline{\boldsymbol{E}}$ representing error is omitted for simplicity).

2. ALPHA AND BETA DIVERGENCES AND ASSOCIATED ALGORITHMS

To deal with the model (1) efficiently we adopt several approaches from constrained optimization and multi-criteria optimization, where we minimize simultaneously several cost functions using alternating switching between sets of parameters. Alpha and Beta divergences are two complimentary generalized cost functions which can be applied for NMF and NTF [1,3,4,5,20].

2.1. Alpha Divergence

Let us consider a flexible and general class of the cost functions, called α-divergence [1,3,4]:

$$D_A^{(\alpha)}(\bar{\boldsymbol{X}}||\bar{\boldsymbol{A}}\boldsymbol{S}) = \frac{\sum_{pt}(\bar{x}_{pt}^{\alpha}[\bar{\boldsymbol{A}}\boldsymbol{S}]_{pt}^{1-\alpha} - \alpha\bar{x}_{pt} + (\alpha-1)[\bar{\boldsymbol{A}}\boldsymbol{S}]_{pt})}{\alpha(\alpha-1)}$$

$$D_{Ak}^{(\alpha)}(\boldsymbol{X}_k||\boldsymbol{A}_k\boldsymbol{S}) = \\ = \frac{\sum_{it}(x_{itk}^{\alpha}[\boldsymbol{A}_k\boldsymbol{S}]_{it}^{1-\alpha} - \alpha x_{itk} + (\alpha-1)[\boldsymbol{A}_k\boldsymbol{S}]_{it})}{\alpha(\alpha-1)}$$

We note that as special cases of α-divergence for $\alpha = 2, 0.5, -1$, we obtain the Pearson's, Hellinger's and Neyman's chi-square distances, respectively, while for the cases $\alpha = 1$ and $\alpha = 0$ the divergence has to be defined by the limits: $\alpha \to 1$ and $\alpha \to 0$, respectively. When these limits are evaluated one obtains for $\alpha \to 1$ the generalized Kullback-Leibler divergence (I-divergence) and for $\alpha \to 0$ the dual generalized KL divergence [1,3,4].

Instead of applying the standard gradient descent method, we use the nonlinearly transformed gradient approach as generalization of the exponentiated gradient (EG)[4] :

$$\Phi(a_{irk}) \leftarrow \Phi(a_{irk}) - \eta_{irk}\frac{\partial D_{A_k}^{(\alpha)}(\boldsymbol{X}_k||\boldsymbol{A}_k\boldsymbol{S})}{\partial\Phi(a_{irk})}, \quad (2)$$

$$\Phi(s_{rt}) \leftarrow \Phi(s_{rt}) - \eta_{rt}\frac{\partial D_{A}^{(\alpha)}(\bar{\boldsymbol{X}}||\bar{\boldsymbol{A}}\boldsymbol{S})}{\partial\Phi(s_{rt})}, \quad (3)$$

where $\Phi(x)$ is a suitably chosen function.
It can be shown that such a nonlinear scaling or transformation provides a stable solution and the gradients are much better behaved in the space Φ. In our case, we employ $\Phi(x) = x^{\alpha}$, which leads directly to the new learning algorithm (for $\alpha \neq 0$) (the rigorous proof of local convergence similar to this given by Lee and Seung [12] is omitted due to a lack of space):

$$a_{irk} \leftarrow a_{irk}\left(\frac{\sum_{t=1}^{T}(x_{itk}/[\boldsymbol{A}_k\boldsymbol{S}]_{it})^{\alpha}\, s_{rt}}{\sum_{t=1}^{T} s_{rt}}\right)^{1/\alpha}, \quad (4)$$

$$s_{rt} \leftarrow s_{rt}\left(\frac{\sum_{p=1}^{KI}\bar{a}_{pr}\,\left(\bar{x}_{pt}/[\bar{\boldsymbol{A}}\boldsymbol{S}]_{pt}\right)^{\alpha}}{\sum_{p=1}^{KI}\bar{a}_{pr}}\right)^{1/\alpha}. \quad (5)$$

2.2. Beta Divergence

Regularized beta divergence for the NTF2 problem can be defined as follows:

$$D^{(\beta)}(\bar{\boldsymbol{X}}||\bar{\boldsymbol{A}}\boldsymbol{S}) = \sum_{pt}(\bar{x}_{pt}\,\frac{\bar{x}_{pt}^{\beta} - [\bar{\boldsymbol{A}}\boldsymbol{S}]_{pt}^{\beta}}{\beta(\beta+1)}$$
$$+[\bar{\boldsymbol{A}}\boldsymbol{S}]_{pt}^{\beta}\frac{[\bar{\boldsymbol{A}}\boldsymbol{S}]_{pt} - \bar{x}_{pt}}{\beta+1}) + \alpha_S\|\boldsymbol{S}\|_{L1}, \quad (6)$$

$$D_k^{(\beta)}(\boldsymbol{X}_k||\boldsymbol{A}_k\boldsymbol{S}) = \sum_{it}(x_{itk}\,\frac{x_{itk}^{\beta} - [\boldsymbol{A}_k\boldsymbol{S}]_{it}^{\beta}}{\beta(\beta+1)}$$
$$+[\boldsymbol{A}_k\boldsymbol{S}]_{it}^{\beta}\frac{[\boldsymbol{A}_k\boldsymbol{S}]_{it} - x_{itk}}{\beta+1}) + \alpha_{A_k}\|\boldsymbol{A}_k\|_{L1}, \quad (7)$$
$$k = 1, \ldots, K, \;\; t = 1, 2, \ldots, T, \;\; i = 1, 2, \ldots, I,$$

where α_S and α_{A_k} are small positive regularization parameters which control the degree of sparseness of the matrices $\boldsymbol{S}$ and $\boldsymbol{A}_k$, respectively, and the $L1$-norms defined as $||\boldsymbol{S}||_{L1} = \sum_{rt}|s_{rt}|$ and $||\boldsymbol{A}_k||_{L1} = \sum_{ir}|a_{irk}|$ are introduced to enforce sparse representations of the solutions. It is interesting to note that for $\beta = 1$, we obtain the squared Euclidean distances expressed by the Frobenius norms $\|\boldsymbol{X}_k - \boldsymbol{A}_k\boldsymbol{S}\|_F^2$, while for the singular cases, $\beta = 0$ and $\beta = -1$, the beta divergence has to be defined as limiting cases as $\beta \to 0$ and $\beta \to -1$, respectively. When these limits are evaluated one gets for $\beta \to 0$ the generalized Kullback-Leibler divergence (called I-divergence) and for $\beta \to -1$ we obtain the Itakura-Saito distance. The choice of the parameter β depends on the statistical distribution of the data and the beta

divergence corresponds to the Tweedie models. For example, the optimal choice of the parameter for the normal distribution is $\beta = 1$, for the gamma distribution is $\beta \to -1$, for the Poisson distribution $\beta \to 0$, and for the compound Poisson $\beta \in (-1, 0)$. By minimizing the beta divergence, we have derived various kinds of NTF algorithms: Multiplicative based on the standard gradient descent, Exponentiated Gradient (EG), Projected Gradient (PG), Alternating Interior-Point Gradient (AIPG), or Fixed Point Alternating Least Squares (FPALS) algorithms. For example, in order to derive a flexible multiplicative learning algorithm, we compute the gradient of (6)-(7) with respect to elements of matrices $s_{rt} = s_r(t) = [\boldsymbol{S}]_{rt}$ and $a_{irk} = [\boldsymbol{A}_k]_{ir}$ and performing simple mathematical manipulations we obtain the multiplicative update rules:

$$a_{irk} \leftarrow a_{irk} \frac{[\sum_{t=1}^{T}(x_{itk}/[\boldsymbol{A}_k\boldsymbol{S}]_{it}^{1-\beta})\, s_{rt} - \alpha_{A_k}]_{\varepsilon}}{\sum_{t=1}^{T}[\boldsymbol{A}_k\boldsymbol{S}]_{it}^{\beta}\, s_{rt}}, \quad (8)$$

$$s_{rt} \leftarrow s_{rt} \frac{[\sum_{p=1}^{KI} \bar{a}_{pr}\, (\bar{x}_{itk}/[\bar{\boldsymbol{A}}\boldsymbol{S}]_{it}^{1-\beta}) - \alpha_S]_{\varepsilon}}{\sum_{p=1}^{KI} \bar{a}_{pr}\, [\bar{\boldsymbol{A}}_k\boldsymbol{S}]_{it}^{\beta}}, \quad (9)$$

where $[x]_{\varepsilon} = \max\{\varepsilon, x\}$ with a small $\varepsilon = 10^{-16}$ is introduced in order to avoid zero and negative values.

In the special case for $\beta = 1$ we can derive an alternative algorithm, called FPALS (Fixed Point Alternating Least Squares) algorithm (see [5] for detail)

$$\boldsymbol{A}_k \leftarrow \left[(\boldsymbol{X}_k\boldsymbol{S}^T - \alpha_{A_k}\boldsymbol{E}_A)(\boldsymbol{S}\boldsymbol{S}^T)^{+}\right]_{\varepsilon}, \quad (10)$$

$$\boldsymbol{S} \leftarrow \left[(\bar{\boldsymbol{A}}^T\bar{\boldsymbol{A}})^{+}(\bar{\boldsymbol{A}}^T\bar{\boldsymbol{X}} - \alpha_S\boldsymbol{E}_S)\right]_{\varepsilon}, \quad (11)$$

where $[\boldsymbol{A}]^{+}$ denotes Moore-Penrose pseudo-inverse of $\boldsymbol{A}$ and $\boldsymbol{E}_A \in \mathbb{R}^{I\times R}$, $\boldsymbol{E}_S \in \mathbb{R}^{R\times T}$ are matrices with all entries one. The above algorithm can be considered as a nonlinear projected Alternating Least Squares (ALS) or nonlinear extension of EM-PCA algorithm.

Furthermore, using the Alternating Interior-Point Gradient (AIPG) approach [14], another efficient algorithm has been developed and implemented [5]:

$$\boldsymbol{A}_k \leftarrow \boldsymbol{A}_k - \eta_{A_k}\boldsymbol{P}_{A_k}, \quad (12)$$

$$\boldsymbol{S} \leftarrow \boldsymbol{S} - \eta_S\boldsymbol{P}_S, \quad (13)$$

where $\boldsymbol{P}_{A_k} = \left(\boldsymbol{A}_k \oslash (\boldsymbol{A}_k\boldsymbol{S}\boldsymbol{S}^T)\right) \odot \left((\boldsymbol{A}_k\boldsymbol{S} - \boldsymbol{X}_k)\boldsymbol{S}^T\right)$, $\boldsymbol{P}_S = \left(\boldsymbol{S} \oslash (\bar{\boldsymbol{A}}^T\bar{\boldsymbol{A}}\boldsymbol{S})\right) \odot \left(\bar{\boldsymbol{A}}^T(\bar{\boldsymbol{A}}\boldsymbol{S} - \bar{\boldsymbol{X}})\right)$ and operators $\odot$ and $\oslash$ mean component-wise multiplication and division, respectively. The learning rates η_{A_k} and η_S are selected in this way to ensure the steepest descent, and on the other hand, to maintain non-negativity. Thus, $\eta_S = \min\{\tau\hat{\eta}_S, \eta_S^*\}$ and $\eta_{A_k} = \min\{\tau\hat{\eta}_{A_k}, \eta_{A_k}^*\}$, where $\tau \in (0, 1)$, $\hat{\eta}_S = \{\eta : \boldsymbol{S} - \eta\boldsymbol{P}_S\}$ and $\hat{\eta}_{A_k} = \{\eta : \boldsymbol{A}_k - \eta\boldsymbol{P}_{A_k}\}$ ensure non-negativity, and

$$\eta_{A_k}^* = \frac{\mathrm{vec}(\boldsymbol{P}_{A_k})^T\mathrm{vec}(\boldsymbol{A}_k\boldsymbol{S}\boldsymbol{S}^T - \boldsymbol{X}_k\boldsymbol{S}^T)}{\mathrm{vec}(\boldsymbol{P}_{A_k}\boldsymbol{S})^T\mathrm{vec}(\boldsymbol{P}_{A_k}\boldsymbol{S})}, \quad (14)$$

$$\eta_S^* = \frac{\mathrm{vec}(\boldsymbol{P}_S)^T\mathrm{vec}(\bar{\boldsymbol{A}}^T\bar{\boldsymbol{A}}\boldsymbol{S} - \bar{\boldsymbol{A}}^T\bar{\boldsymbol{X}})}{\mathrm{vec}(\boldsymbol{A}_k\boldsymbol{P}_S)^T\mathrm{vec}(\boldsymbol{A}_k\boldsymbol{P}_S)} \quad (15)$$

are the adaptive steepest descent learning rates.

3. SIMULATION RESULTS

All the NMF algorithms discussed in this paper have been extensively tested for many difficult benchmarks for signals and images with various statistical distributions and also for real EEG data. We found the best performance can be obtained with the AIPG, FPALS and the algorithm (8)-(9) for $\beta = 1$.

Due to space limitation, we present here only one simulation Example: Six natural highly correlated images are mixed by randomly generated 3D tensor $\underline{\boldsymbol{A}} \in \mathbb{R}_{+}^{12\times 6\times 25}$. The observed mixed data are collected in 3D tensor $\underline{\boldsymbol{X}} \in \mathbb{R}_{+}^{12\times 64^2\times 25}$. The exemplary results are shown in Fig.2.

4. CONCLUSIONS AND DISCUSSION

In this paper we proposed generalized and flexible cost functions (controlled by a single parameter alpha or beta) that allows us to derive a family of robust and efficient NTF algorithms. The optimal choice of a free parameter of a specific cost function depends on a statistical distribution of data and additive noise, thus various criteria and algorithms (updating rules) should be applied for estimating the basis matrices $\boldsymbol{A}_k$ and the source matrix $\boldsymbol{S}$, depending on *a priori* knowledge about the statistics of noise or errors. It is worth to mention that we can use three different strategies to estimate common factors (the source matrix $\boldsymbol{S}$). In the first approach, presented in this paper, we use two different cost functions: A global cost function (using unfolded column-wise matrices: $\bar{\boldsymbol{X}}$, $\bar{\boldsymbol{S}}$ for frontal slices of 3D tensors) to estimate the common factors $\boldsymbol{S}$, i.e., the source matrix $\boldsymbol{S}$; and local cost functions to estimate the slices $\boldsymbol{A}_k$, $(k = 1, 2, ..., K)$. However, instead of using the unfolding matrices for the NTF2 model, in order to estimate $\boldsymbol{S}$, we can use, average matrices defined as $\bar{\boldsymbol{X}} = \sum_k \boldsymbol{X}_k \in \mathbb{R}^{I\times T}$ and $\bar{\boldsymbol{A}} = \sum_k \boldsymbol{A}_k \in \mathbb{R}^{I\times R}$. Furthermore, it is also possible to apply a different approach by using only set of local cost functions, e.g., $D_k(\boldsymbol{X}_k||\boldsymbol{A}_k\boldsymbol{S}) = 0.5|\boldsymbol{X}_k - \boldsymbol{A}_k\boldsymbol{S}||_F^2$. In such a case, we estimate $\boldsymbol{A}_k$ and $\boldsymbol{S}$ cyclically by applying alternating minimization (similar to row-action projection of the Kaczmarz algorithm). We found that such approaches also work quite well for the NTF2 model. The advantage of the last approach is that the all updates learning rules are local (slice by slice) and algorithms are generally faster for large data, (especially, if $K >> 1$).

Obviously, 3D NTF models can be transformed to a 2D non-negative matrix factorization (NMF) problem by unfold-

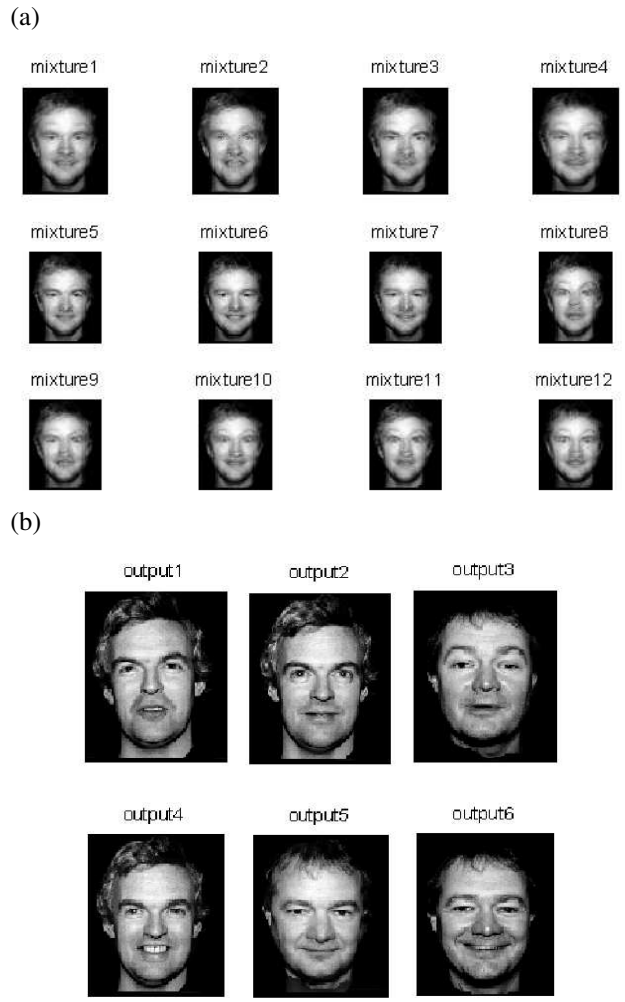

Fig. 2. Example 1: (a) Observed mixed images (only subset of the observed images are shown); (b) Estimated source images using the FPALS algorithm (SIR = 20.8, 22.7, 21.5, 23.1, 20.4, 24.2 [dB], respectively).

ing (matricizing) tensors. However, it should be noted that such a 2D model is not exactly equivalent to the NMF2 model, since in practice we often need to impose different additional constraints for each slice. In other words, the NTF2 model should not be considered as equal to a standard 2-way NMF of a single unfolded 2-D matrix. The profiles of the stacked (column-wise unfolded) $\bar{\boldsymbol{A}}$ are often not treated as single profiles and the constraints are usually applied independently to each $\boldsymbol{A}_k$ sub-matrix that form the stacked $\bar{\boldsymbol{A}}$. Moreover, the NTF2 is considered as a dynamical process, where the data analysis is performed several times under different conditions (multi-start initializations, multilayer or recurrent implementation, Monte Carlo analysis, selection of additional natural constraints, etc.) to get full information about the available data and/or discover some inner structures in the data.

We have been motivated to develop proposed NTF algorithm by using them in three areas of data analysis (especially, EEG data) and signal/image processing: (i) multi-way blind source separation, (ii) model reductions and selection and (iii) sparse image coding. The proposed models can be further extended by imposing additional, natural constraints such as smoothness, continuity, closure, unimodality, local rank, selectivity, and/or by taking into account a prior knowledge about specific 3D, or more generally, multi-way data. Obviously, there are many challenging open issues remaining, such as global convergence, optimal choice of parameters and uniqueness of a solution when additional constraints are imposed.

5. REFERENCES

[1] S. Amari, *D*ifferential-Geometrical Methods in Statistics, 1985 Springer Verlag.

[2] M. Berry, M. Browne, A. Langville, P. Pauca, and R. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization", *Computational Statistics and Data Analysis*, 2006 http://www.wfu.edu/ plemmons/papers.htm.

[3] A. Cichocki, R. Zdunek, and S. Amari, "Csiszar's divergences for non-negative matrix factorization: Family of new algorithms" ICA-2006, Springer LNCS 3889, 32-39, 2006.

[4] A. Cichocki, S. Amari, R. Zdunek, R. Kompass, G. Hori, and Z. He, "Extended SMART algorithms for non-negative matrix factorization", Springer LNAI 4029, 548–562 2006.

[5] A. Cichocki and R. Zdunek, "NTFLAB for Signal and Image Processing", Tech. rep., Laboratory for Advanced Brain Signal Pro- cessing, BSI RIKEN, Saitama, Japan, http://www.bsp.brain.riken.jp (2006).

[6] L. De Lathauwer, and P. Comon (Editors), Workshop on Tensor Decompositions and Applications, CIRM, Marseille, France, 2005 http://www.etis.ensea.fr/ wtda/

[7] I. Dhillon and S. Sra, "Generalized nonnegative matrix approximations with Bregman divergences". In: NIPS -Neural Information Proc. Systems, Vancouver Canada, 2005.

[8] T. Hazan, S. Polak, and A. Shashua, "Sparse image coding using a 3D non-negative tensor factorization", in Proc. of ICCV, 50–57, 2005.

[9] M. Heiler and C. Schnoerr, "Controlling sparseness in Non-Negative Tensor Factorization", ECCV 2006, http://www.cvgpr.uni-mannheim.de/heiler/

[10] P. Hoyer, "Non-negative matrix factorization with sparseness constraints" *Journal of Machine Learning Research* 5, 1457–1469, 2004

[11] M. Kim, and S. Choi, "Monaural music source separation: Nonnegativity, sparseness, and shift-invariance", ICA-2006, Springer LNCS 3889, pp. 617-624, 2006.

[12] D.D. Lee, and H.S. Seung, "Learning of the parts of objects by non-negative matrix factorization". *Nature*, 401, 788-791, 1999.

[13] H. Li, T. Adali, and D.E. Wang, "Non-negative matrix factorization with orthogonality constraints for chemical agent detection in Raman spectra", In: IEEE Workshop on Machine Learning for Signal Processing, Mystic USA, 2005.

[14] M. Merritt, and Y. Zhang, "Interior-point gradient method for large-scale totally nonnegative least squares problems" *Journal of Optimization Theory and Applications*, 126(1), 193–202, 2005.

[15] M. Morup, L.K. Hansen, C.S. Herrmann, J. Parnas and S.M. Arnfred, "Parallel factor analysis as an exploratory tool for wavelet transformed event-related EEG", *NeuroImage*, 29(3):938–947, 2006.

[16] F. Miwakeichi, E. Martinez-Montes, P.A. Valdes-Sosa, N. Nishiyama, H. Mizuhara, and Y. Yamaguchi, "Decomposing EEG data into spacetime frequency components using parallel factor analysis", *NeuroImage*, 22, 1035-1045, 2004.

[17] A. Smilde, R. Bro, and P. Geladi, *Multi-way Analysis: Applications in the Chemical Sciences*, Wiley, NY, 2004.

[18] M.A.O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensorfaces" In Proc. European Conf. on Computer Vision (ECCV), pages 447-460, Copenhagen, Denmark, 2002.

[19] S.A. Vorobyov, Y. Rong, N.D. Sidiropoulos, and A.B. Gershman, "Robust iterative fitting of multilinear models", *IEEE Trans. Signal Processing*, vol. 53, no. 8, pp. 2678-2689, 2005.

[20] R. Kompass, "A generalized divergence measure for nonnegative matrix factorization", Neural Computation, 2006 (in print).