NONNEGATIVE TUCKER DECOMPOSITION WITH ALPHA-DIVERGENCE

Yong-Deok Kim [§], *Andrzej Cichocki* [†], *Seungjin Choi* [§]

[§] Department of Computer Science, POSTECH, Korea
{karma13,seungjin}@postech.ac.kr
[†] Brain Science Institute, RIKEN, Japan
cia@brain.riken.jp

ABSTRACT

Nonnegative Tucker decomposition (NTD) is a recent multiway extension of nonnegative matrix factorization (NMF), where nonnegativity constraints are incorporated into Tucker model. In this paper we consider α-divergence as a discrepancy measure and derive multiplicative updating algorithms for NTD. The proposed multiplicative algorithm includes some existing NMF and NTD algorithms as its special cases, since α-divergence is a one-parameter family of divergences which accommodates KL-divergence, Hellinger divergence, χ^2 divergence, and so on. Numerical experiments on face images show how different values of α affect the factorization results under different types of noise.

Index Terms— α-divergence, nonnegative matrix factorization, tensor factorization, Tucker models.

1. INTRODUCTION

Nonnegative matrix factorization (NMF) is a widely-used multivariate analysis of nonnegative data [1] which has many potential applications in machine learning, pattern recognition, and signal processing. Successful applications of NMF result from its ability to learn a parts-based representation through a matrix factorization, $\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S}$, where $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_l] \in \mathbb{R}_+^{m\times l}$ is a data matrix, $\boldsymbol{A} \in \mathbb{R}_+^{m\times r}$ is a basis matrix, and $\boldsymbol{S} \in \mathbb{R}_+^{r\times l}$ is an encoding variable matrix.

In many real applications, data has a multiway structure. Exemplary data are video stream (rows, columns, RGB color coordinates, time), EEG in neuroscience (channels, frequency, time, samples), network flow (source ip, destination ip, source port, destination port, time), bibliographic data (keywords, papers, authors, journals), and so on. Conventional methods preprocess multiway data, putting them into a matrix. Recently, there has been a great deal of research on multilinear analysis which conserves the original multiway structure of the data. Motivated by multiway extensions of SVD, NMF was also extended to 2D-NMF [2], nonnegative tensor factorization (NTF) [3, 4], NTF2 [5], higher-order NMF [6], and nonnegative Tucker decomposition (NTD) [7] which are based on Tucker2, CANDECOMP/PARAFAC [8, 9], PARAFAC2, and Tucker model [10], respectively. Recently NTF was shown to be useful in feature extraction for continuous EEG classification [11].

LS error function or KL-divergence has been widely used as a discrepancy measure in NMF, NTF, and NTD. Recently various divergence measures such as Csiszár's f-divergences, α-divergences, and Bregman divergences, were considered as discrepancy measures in NMF [12, 13] and NTF [5].

In this paper we further elaborate our recent work on NTD [7], considering the α-divergence. We develop multiplicative updating algorithms, referred to as 'α-NTD', which iteratively minimize the α-divergence between nonnegative data tensor and Tucker model. Empirically we investigate the role of α on image de-noising by α-NTD, confirming that the parameter α is related to the characteristics of a learning machine, where the α-divergence of q from p ($D_\alpha[p||q]$) emphasizes the part where p is small as α increases [14].

2. BACKGROUND

2.1. α-Divergence

Let us consider two unnormalized distributions $p(x)$ and $q(x)$ associated with a random variable x. The α-divergence [15, 16, 17], that belongs to Csiszár's f-divergence [18], is a parametric family of divergence functional, defined by

$$D_\alpha[p||q] = \frac{1}{\alpha(1-\alpha)} \int \alpha p + (1-\alpha) q - p^\alpha q^{1-\alpha}\, d\mu, \qquad (1)$$

where $\alpha \in (-\infty, \infty)$ and μ is the Lebesque measure. As in KL divergence, α-divergence is zero if $p = q$ and positive otherwise. This property follows from the fact that α-divergence (1) is convex with respect to p and q. The α-divergence includes KL-divergence ($KL[q||p]$ for $\alpha \to 0$ or $KL[p||q]$ for $\alpha \to 1$), Hellinger divergence ($\alpha = \frac{1}{2}$), and χ^2-divergence ($\alpha = 2$), as its special cases.

Considering α-divergence as an error measure, the objective function for NMF is given by

$$D_\alpha[\boldsymbol{X}||\boldsymbol{A}\boldsymbol{S}] = \frac{\sum_{i,j} \alpha X_{ij} + (1-\alpha)[\boldsymbol{A}\boldsymbol{S}]_{ij} - X_{ij}^\alpha [\boldsymbol{A}\boldsymbol{S}]_{ij}^{1-\alpha}}{\alpha(1-\alpha)}.$$

This objective function is iteratively minimized by the following multiplicative updating algorithms:

$$\boldsymbol{S} \leftarrow \boldsymbol{S} \odot \left\{ \frac{\boldsymbol{A}^\top (\boldsymbol{X}/[\boldsymbol{A}\boldsymbol{S}])^{\cdot\alpha}}{\boldsymbol{A}^\top \boldsymbol{1}\boldsymbol{1}^\top} \right\}^{\cdot\frac{1}{\alpha}}, \qquad (2)$$

$$\boldsymbol{A} \leftarrow \boldsymbol{A} \odot \left\{ \frac{(\boldsymbol{X}/[\boldsymbol{A}\boldsymbol{S}])^{\cdot\alpha} \boldsymbol{S}^\top}{\boldsymbol{1}\boldsymbol{1}^\top \boldsymbol{S}^\top} \right\}^{\cdot\frac{1}{\alpha}}, \qquad (3)$$

where $\odot$ is Hadamard product, $/$ is element-wise division, $\boldsymbol{1} = [1, \ldots, 1]^\top$, and $[\boldsymbol{X}]^{\cdot\alpha} = [X_{ij}^\alpha]$. Updating rules (2) and (3), referred to as 'α-NMF', can be easily derived using the same technique as used in [1].

2.2. Nonnegative Tucker Decomposition

An N-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ has N indices $(i_1, i_2, \ldots, i_N)$ and its elements are denoted by $\mathcal{X}_{i_1 i_2 \ldots i_N}$ where $1 \le i_n \le I_n$. The mode-n matricization of $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ rearranges the elements of $\boldsymbol{\mathcal{X}}$ to form the matrix $\boldsymbol{X}_{(n)} \in \mathbb{R}^{I_n \times I_{n+1} I_{n+2} \cdots I_N I_1 I_2 \cdots I_{n-1}}$, where $I_{n+1} I_{n+2} \cdots I_N I_1 I_2 \cdots I_{n-1}$ is in a cyclic order. We follow the standardized notation and convention in Kiers' work [19].

The mode-n product of a tensor $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_n \times \cdots \times J_N}$ by a matrix $\boldsymbol{A}^{(n)} \in \mathbb{R}^{I_n \times J_n}$ is defined by

$$\begin{aligned} &\left[\boldsymbol{\mathcal{S}} \times_n \boldsymbol{A}^{(n)}\right]_{j_1 \cdots j_{n-1} i_n j_{n+1} \cdots j_N} \\ &\quad = \sum_{j_n=1}^{J_n} \mathcal{S}_{j_1 \ldots j_{n-1} j_n j_{n+1} \cdots j_N} A_{i_n j_n}, \end{aligned} \tag{4}$$

leading to a tensor $\boldsymbol{\mathcal{S}} \times_n \boldsymbol{A}^{(n)} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times I_n \times \cdots \times J_N}$. With the mode-$n$ product, the matrix factorization $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$ is written as $\boldsymbol{X} = \boldsymbol{S} \times_1 \boldsymbol{U} \times_2 \boldsymbol{V}$.

Nonnegative Tucker decomposition (NTD) seeks a decomposition of a nonnegative N-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}_+^{I_1 \times I_2 \times \cdots \times I_N}$ as mode products of a nonnegative core tensor $\boldsymbol{\mathcal{S}} \in \mathbb{R}_+^{J_1 \times J_2 \times \cdots \times J_N}$ and N nonnegative mode matrices $\boldsymbol{A}^{(n)} \in \mathbb{R}_+^{I_n \times J_n}$,

$$\boldsymbol{\mathcal{X}} \approx \widehat{\boldsymbol{\mathcal{X}}} = \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{(1)} \times_2 \boldsymbol{A}^{(2)} \cdots \times_N \boldsymbol{A}^{(N)}, \tag{5}$$

which can be written in an element-wise form as

$$\widehat{\mathcal{X}}_{i_1 i_2 \cdots i_N} = \sum_{j_1, j_2, \ldots, j_N} \mathcal{S}_{j_1 j_2 \cdots j_N} A^{(1)}_{i_1 j_1} A^{(2)}_{i_2 j_2} \cdots A^{(N)}_{i_N j_N}. \tag{6}$$

The mode-n matricization of $\boldsymbol{\mathcal{X}}$ in Tucker model (5), is expressed by Kronecker products of the mode-n matricization of the core tensor and mode matrices:

$$\begin{aligned} \boldsymbol{X}_{(n)} &\approx \boldsymbol{A}^{(n)} \boldsymbol{S}_{(n)} \Big[\boldsymbol{A}^{(n-1)} \otimes \cdots \otimes \boldsymbol{A}^{(2)} \otimes \boldsymbol{A}^{(1)} \\ &\qquad \otimes \boldsymbol{A}^{(N)} \otimes \cdots \otimes \boldsymbol{A}^{(n+2)} \otimes \boldsymbol{A}^{(n+1)}\Big]^\top \\ &= \boldsymbol{A}^{(n)} \boldsymbol{S}_{(n)} \boldsymbol{A}^{(\backslash n)\top}, \end{aligned} \tag{7}$$

where $\boldsymbol{S}_{(n)}$ is the mode-n matricization of the core tensor $\boldsymbol{\mathcal{S}}$. The representation (7) plays a crucial role in deriving multiplicative updating algorithms for NTD.

NTD provides a general framework for nonnegative tensor factorization, including NMF, nsNMF, 2D-NMF, and NTF as special cases [7]. Models for NMF, nsNMF, 2D-NMF, and NTF are summarized in Table 1 for easy comparison to NTD (5). In NTD, the core tensor and mode matrices are learned iteratively by multiplicative updates. Various methods such as NMF, nsNMF, 2D-NMF, and NTF emerge from NTD, by imposing some pre-specified structure on the core tensor or mode matrices instead of learning them.

3. α-NONNEGATIVE TUCKER DECOMPOSITION

α-NTD considers the objective function that is the α-divergence of Tucker model $\widehat{\boldsymbol{\mathcal{X}}}$ (given in (5)) from the N-way tensor $\boldsymbol{\mathcal{X}}$ of data:

$$\begin{aligned} D_\alpha[\boldsymbol{\mathcal{X}} \| \widehat{\boldsymbol{\mathcal{X}}}] &= \frac{1}{\alpha(1-\alpha)} \sum_{i_1, i_2, \ldots, i_N} \alpha \mathcal{X}_{i_1 i_2 \cdots i_N} \\ &+ (1-\alpha)\widehat{\mathcal{X}}_{i_1 i_2 \cdots i_N} - \mathcal{X}^{\alpha}_{i_1 i_2 \cdots i_N} \widehat{\mathcal{X}}^{1-\alpha}_{i_1 i_2 \cdots i_N}. \end{aligned} \tag{8}$$

Multiplicative updating algorithms for NTD were elegantly derived in [7] when LS error function or KL-divergence was used as an objective function. We derive multiplicative updates for NTD in a similar way when α-divergence objective function (8) is considered.

We use the following properties in deriving multiplicative updates for α-NTD:

$$\begin{aligned} \operatorname{vec}\left(\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^\top\right) &= (\boldsymbol{V} \otimes \boldsymbol{U}) \operatorname{vec}(\boldsymbol{B}), \\ [\boldsymbol{U} \otimes \boldsymbol{V}]^\top &= \boldsymbol{U}^\top \otimes \boldsymbol{V}^\top, \\ (\boldsymbol{U} \otimes \boldsymbol{V})(\boldsymbol{B} \otimes \boldsymbol{C}) &= \boldsymbol{U}\boldsymbol{B} \otimes \boldsymbol{V}\boldsymbol{C}. \end{aligned}$$

The core idea in developing multiplicative updates for α-NTD is to use the mode-n matricization of Tucker model $\widehat{\boldsymbol{\mathcal{X}}}$, given by

$$\boldsymbol{X}_{(n)} \approx \boldsymbol{A}^{(n)} \boldsymbol{S}_{(n)} \boldsymbol{A}^{(\backslash n)\top} = \boldsymbol{A}^{(n)} \boldsymbol{S}_A^{(n)}, \tag{9}$$

$$\begin{aligned} \operatorname{vec}\left(\boldsymbol{X}_{(n)}\right) &\approx \operatorname{vec}\left(\boldsymbol{A}^{(n)} \boldsymbol{S}_{(n)} \boldsymbol{A}^{(\backslash n)\top}\right) \\ &= \left(\boldsymbol{A}^{(\backslash n)} \otimes \boldsymbol{A}^{(n)}\right) \operatorname{vec}\left(\boldsymbol{S}_{(n)}\right). \end{aligned} \tag{10}$$

Note that these matricized equations have the same form as NMF. Thus, update for the core tensor follows (2) and updates for mode matrices are derived from (3).

We present a detailed derivation of the updating rule only for the core tensor since updates for mode matrices directly emerge from (3) and (9). It follows from (2) and (10) that the n-mode matricized core tensor is updated by

$$\operatorname{vec}\left(\boldsymbol{S}_{(n)}\right) \leftarrow \operatorname{vec}\left(\boldsymbol{S}_{(n)}\right) \odot \boldsymbol{K}_n^{\cdot \frac{1}{\alpha}}, \tag{11}$$

where

$$\boldsymbol{K}_n = \frac{\left[\boldsymbol{A}^{(\backslash n)} \otimes \boldsymbol{A}^{(n)}\right]^\top \left[\operatorname{vec}\left(\boldsymbol{X}_{(n)}\right) / \operatorname{vec}\left(\widehat{\boldsymbol{X}}_{(n)}\right)\right]^{\cdot \alpha}}{\left[\boldsymbol{A}^{(\backslash n)} \otimes \boldsymbol{A}^{(n)}\right]^\top \boldsymbol{1}}. \tag{12}$$

Note that

$$\operatorname{vec}\left(\boldsymbol{X}_{(n)}\right) / \operatorname{vec}\left(\widehat{\boldsymbol{X}}_{(n)}\right) = \operatorname{vec}\left(\left[\boldsymbol{\mathcal{X}} / \widehat{\boldsymbol{\mathcal{X}}}\right]_{(n)}\right).$$

Invoking (9) with this relation leads to

$$\begin{aligned} &\left[\boldsymbol{A}^{(\backslash n)} \otimes \boldsymbol{A}^{(n)}\right]^\top \operatorname{vec}\left(\left[\boldsymbol{\mathcal{X}} / \widehat{\boldsymbol{\mathcal{X}}}\right]_{(n)}\right)^{\cdot \alpha} \\ &\quad = \operatorname{vec}\left(\boldsymbol{A}^{(n)\top} \left[\boldsymbol{\mathcal{X}} / \widehat{\boldsymbol{\mathcal{X}}}\right]_{(n)}^{\cdot \alpha} \boldsymbol{A}^{(\backslash n)}\right) \\ &\quad = \operatorname{vec}\left(\left[(\boldsymbol{\mathcal{X}} / \widehat{\boldsymbol{\mathcal{X}}})^{\cdot \alpha} \times_1 \boldsymbol{A}^{(1)\top} \cdots \times_N \boldsymbol{A}^{(N)\top}\right]_{(n)}\right). \end{aligned}$$

In a similar way, we have

$$\left[\boldsymbol{A}^{(\backslash n)} \otimes \boldsymbol{A}^{(n)}\right]^\top \boldsymbol{1} = \operatorname{vec}\left(\left[\boldsymbol{\mathcal{E}} \times_1 \boldsymbol{A}^{(1)\top} \cdots \times_N \boldsymbol{A}^{(N)\top}\right]_{(n)}\right),$$

where $\boldsymbol{\mathcal{E}}$ is a tensor whose every elements are one. With these calculations, the multiplicative updating rules for the core tensor as well as mode matrices in α-NTD are given by

$$\boldsymbol{\mathcal{S}} \leftarrow \boldsymbol{\mathcal{S}} \odot \left\{ \frac{(\boldsymbol{\mathcal{X}} / \widehat{\boldsymbol{\mathcal{X}}})^{\cdot \alpha} \times_1 \boldsymbol{A}^{(1)\top} \cdots \times_N \boldsymbol{A}^{(N)\top}}{\boldsymbol{\mathcal{E}} \times_1 \boldsymbol{A}^{(1)\top} \cdots \times_N \boldsymbol{A}^{(N)\top}} \right\}^{\cdot \frac{1}{\alpha}} \tag{13}$$

$$\boldsymbol{A}^{(n)} \leftarrow \boldsymbol{A}^{(n)} \odot \left\{ \frac{\left[(\boldsymbol{\mathcal{X}} / \widehat{\boldsymbol{\mathcal{X}}})^{\cdot \alpha}\right]_{(n)} \boldsymbol{S}_A^{(n)\top}}{\boldsymbol{1}\boldsymbol{1}^\top \boldsymbol{S}_A^{(n)\top}} \right\}^{\cdot \frac{1}{\alpha}}, \tag{14}$$

Table 1. Models for NMF, nsNMF, 2D-NMF, and NTF are summarized in the context of tensor factorization. $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{M}$ is a smoothing matrix defined by $(1-\theta)\boldsymbol{I} + \frac{\theta}{R}\mathbf{1}\mathbf{1}^\top$ where the parameter θ satisfies $0 \le \theta \le 1$, and $\boldsymbol{\mathcal{I}}$ is a unit superdiagonal tensor where $i_{i_1 i_2 \cdots i_N} = \delta_{i_1 i_2 \cdots i_N}$. $\boldsymbol{X}_{i_3}$ and $\boldsymbol{S}_{i_3}$ are the i_3th frontal slice of $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{S}}$. i.e. $\boldsymbol{X}_{i_3} = \boldsymbol{\mathcal{X}}_{:,:,i_3}$.

Model	Matrix representation	Tensor representation	Fixed factor
NMF	$\boldsymbol{X} \approx \boldsymbol{A}\boldsymbol{S}$	$\boldsymbol{X} \approx \boldsymbol{I} \times_1 \boldsymbol{A} \times_2 \boldsymbol{S}^\top$	$\boldsymbol{I}$
nsNMF	$\boldsymbol{X} \approx \boldsymbol{A}\boldsymbol{M}\boldsymbol{S}$	$\boldsymbol{X} \approx \boldsymbol{M} \times_1 \boldsymbol{A} \times_2 \boldsymbol{S}^\top$	$\boldsymbol{M}$
2D-NMF	$\boldsymbol{X}_{i_3} \approx \boldsymbol{A}^{(1)}\boldsymbol{S}_{i_3}\boldsymbol{A}^{(2)^\top}, (i_3 = (1, 2, \ldots, I_3)$	$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{(1)} \times_2 \boldsymbol{A}^{(2)} \times_3 \boldsymbol{I}$	$\boldsymbol{I}$
NTF	$\boldsymbol{\mathcal{X}} \approx \sum_{r=1}^{R} \boldsymbol{A}^{(1)}_{:,r} \circ \boldsymbol{A}^{(2)}_{:,r} \circ \cdots \circ \boldsymbol{A}^{(N)}_{:,r}$	$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{I}} \times_1 \boldsymbol{A}^{(1)} \times_2 \boldsymbol{A}^{(2)} \cdots \times_N \boldsymbol{A}^{(N)}$	$\boldsymbol{\mathcal{I}}$

for $n = 1, \ldots, N$. In order to reduce the computational cost, terms in (13) and (14) are computed in the following manner:

$$\begin{aligned}
&\boldsymbol{\mathcal{E}} \times_1 \boldsymbol{A}^{(1)\top} \cdots \times_N \boldsymbol{A}^{(N)\top} = \boldsymbol{A}^{(1)\top}\mathbf{1} \circ \cdots \circ \boldsymbol{A}^{(N)\top}\mathbf{1} \\
&\left[(\boldsymbol{\mathcal{X}}/\widehat{\boldsymbol{\mathcal{X}}})^{.\alpha}\right]_{(n)} \boldsymbol{S}_A^{(n)\top} = \left[(\boldsymbol{\mathcal{X}}/\widehat{\boldsymbol{\mathcal{X}}})^{.\alpha} \times_{m \ne n} \boldsymbol{A}^{(m)\top}\right]_n \boldsymbol{S}_{(n)}^\top \\
&\mathbf{1}^\top \boldsymbol{S}_A^{(n)\top} = \left[\boldsymbol{\mathcal{S}} \times_{m \ne n} \mathbf{1}^\top \boldsymbol{A}^{(m)}\right]_{(n)}^\top, \qquad (15)
\end{aligned}$$

where $\circ$ is the outer product and $\boldsymbol{\mathcal{S}} \times_{m \ne n} \boldsymbol{A}^{(m)} = \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{(1)} \times_2 \cdots \times_{n-1} \boldsymbol{A}^{(n-1)} \times_{n+1} \boldsymbol{A}^{(n+1)} \cdots \times_N \boldsymbol{A}^{(N)}$.

4. NUMERICAL EXPERIMENTS

Our Matlab implementation of α-NTD partly uses the tensor toolbox [20]. We investigate the role of $\alpha \in \{0.5, 1, 2\}$ in the tensor factorization. To this end, we apply α-NMF and α-NTD to a image de-noising task with different types of noise or outliers. We use ORL face DB [21] ($\boldsymbol{\mathcal{X}} \in \mathbb{R}_+^{48\times48\times400}$ collects 48×48 images of 400 people) to generate noise-contaminated images (see Fig. 1) where 3 different types of noise are considered, including pepper (black), salt (white), and pepper & salt (black and white). For each image, 5% of pixels are randomly chosen and then are converted to black or white pixels. In α-NMF, the number of basis is set as 25. In α-NTD, the dimension of the core tensor is set as $24 \times 24 \times 25$. Note that α-NTD require smaller number of parameters (about 38%), compared to α-NMF.

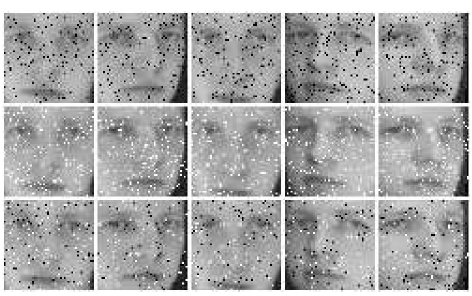

Fig. 1. From top to bottom: images contaminated by pepper (black), salt (white), and pepper & salt (black and white) noise.

Experiments are carried out 20 times independently for each type of noise and each value of $\alpha = \{0.5, 1, 2\}$. As a performance measure, averaged peak-signal-to-noise ratio (PSNR) is used. Higher PSNR values represent better results. The results are shown in Fig. 2 and Table 2. In most of cases, α-NTD is more robust to noise than α-NMF. Interesting observations in these experiments are as follows. The larger α results in the better performance in the case of pepper noise and the smaller α works better in the case of salt noise. In fact, these results are consistent with the characteristics of α-divergence where $D_\alpha[p||q]$ emphasizes the part where p is small as α increases [14].

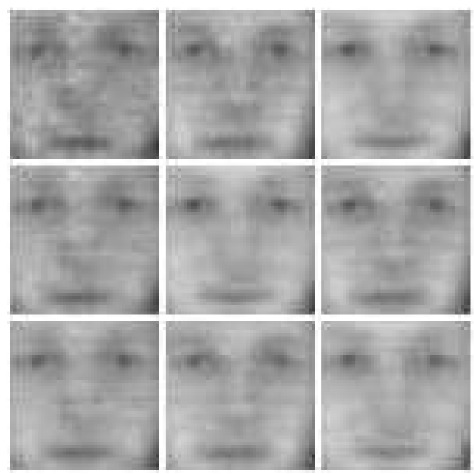

Fig. 2. Reconstructed image by α-NTD. From top to bottom: α is 0.5, 1, and 2. From left to right: pepper (black), pepper & salt (black and white), and salt (white) noise.

In the case of pepper noise, noise-contaminated pixels are replaced by $X_{ij} = 0$ and the associated term in the error function is $\frac{1}{\alpha}\widehat{X}_{ij}$. The larger value of α penalizes such terms, leading to the better performance. In the case of salt noise, noise-contaminated pixels are given as $X_{ij} = 255$ and the associated term in the error function is $\frac{1}{\alpha(1-\alpha)}\{255\alpha + (1-\alpha)\widehat{X}_{ij} - 255^\alpha \widehat{X}_{ij}^{1-\alpha}\}$. This function is shown in Fig. 3, where the smaller value of α de-emphasize the outlier effect.

5. CONCLUSIONS

We have presented a method of nonnegative Tucker decomposition in the case where α-divergence is considered as an error function. We have derived multiplicative updating rules for α-NTD which in-

Table 2. The relationship between types of noise and divergences. In the case of pepper (black) noise, the larger value of α gives better result. In the case of salt (white) noise, the smaller value of α gives better result.

		pepper (black)	pepper & salt	salt (white)
	$\alpha = 0.5$	22.18	25.04	**25.54**
NTD	$\alpha = 1$	24.82	**25.99**	25.09
	$\alpha = 2$	**25.97**	25.49	23.59
	$\alpha = 0.5$	20.30	22.30	**25.43**
NMF	$\alpha = 1$	24.17	**25.27**	24.72
	$\alpha = 2$	**26.31**	22.82	24.85

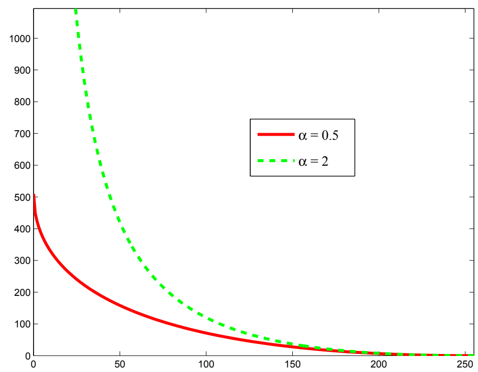

Fig. 3. The behavior of $\frac{1}{\alpha(1-\alpha)}\{255\alpha + (1-\alpha)\widehat{X}_{ij} - 255^{\alpha}\widehat{X}_{ij}^{1-\alpha}\}$ with respect to $\widehat{X}_{ij}$, in the case of salt noise.

clude various existing algorithms such as NMF, nsNMF, NTF, NTF as special cases. We have also investigated the role of α in α-NTD with experiments on image de-noising under 3 different types of noise. Empirical results were well matched with the known characteristics of α-divergence.

Acknowledgments: This work was supported by Korea MIC under ITRC support program supervised by IITA (IITA-2007-C1090-0701-0045) and National Core Research Center for Systems Bio-Dynamics.

6. REFERENCES

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

[2] D. Zhang, S. Chen, and Z. H. Zhou, "Two-dimensional non-negative matrix factorization for face representation and recognition," in *Proceedings of the ICCV-2005 Workshop on Analysis and Modeling of Faces and Gestures*, 2005.

[3] M. Welling and M. Weber, "Positive tensor factorization," *Pattern Recognition Letters*, vol. 22, pp. 1255–1261, 2001.

[4] A. Shashua and T. Hazan, "Non-negative tensor factorization with applications to statistics and computer vision," in *Proceedings of the International Conference on Machine Learning*, Bonn, Germany, 2005.

[5] A. Cichocki, R. Zdunek, S. Choi, R. J. Plemmons, and S. Amari, "Non-negative tensor factorization using alpha and beta divergences," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Honolulu, Hawaii, 2007.

[6] M. Mørup, L. K. Hansen, and S. M. Arnfred, "Sparse higher order non-negative matrix factorization," Tech. Rep., Technical University of Denmark, 2006, http://www.mortenmorup.dk.

[7] Y. -D. Kim and S. Choi, "Nonnegative Tucker decomposition," in *Proceedings of the IEEE CVPR-2007 Workshop on Component Analysis Methods*, Minneapolis, Minnesota, 2007.

[8] J. D. Carroll and J. J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of 'Eckart-Young' decomposition," *Psychometrika*, vol. 35, pp. 283–319, 1970.

[9] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "Exploratory" multi-modal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, pp. 1–84, 1970.

[10] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, pp. 279–311, 1966.

[11] H. Lee, Y. -D. Kim, A. Cichocki, and S. Choi, "Nonnegative tensor factorization for continuous EEG classification," *International Journal of Neural Systems*, vol. 17, no. 4, pp. 305–317, 2007.

[12] A. Cichocki, R. Zdunek, and S. Amari, "New algorithms for non-negative matrix factorization in applications to blind source separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Toulouse, France, 2006.

[13] I. S. Dhillon and S. Sra, "Generalized nonnegative matrix approximations with Bregman divergences," in *Advances in Neural Information Processing Systems*. 2006, vol. 18, MIT Press.

[14] S. Amari, "Integration of stochastic models by minimizing α-divergence," *Neural Computation*, vol. 19, no. 10, pp. 2780–2796, 2007.

[15] S. M. Ali and S. D. Silvey, "A general class of coefficients of divergence of one distribution from another," *Journal of the Royal Statistical Society B*, vol. 28, pp. 131–142, 1966.

[16] S. Amari and H. Nagaoka, *Methods of Information Geometry*, Oxford University Press, 2000.

[17] T. Minka, "Divergence measures and message passing," Tech. Rep. MSR-TR-2005-173, Microsoft Research, 2005.

[18] I. Csiszár, "Information measures: A critical survey," in *Trans. 7th Prague Conference on Information Theory*, 1974, vol. A, pp. 73–86.

[19] H. A. L. Kiers, "Towards a standardized notation and terminology in multiway analysis," *Journal of Chemometrics*, vol. 14, pp. 105–122, 2000.

[20] B. W. Bader and T. G. Kolda, "Algorithm 862: MATLAB tensor classes for fast algorithm prototyping," *ACM Trans. Mathematical Software*, vol. 32, no. 4, pp. 635–653, 2006.

[21] F. Samaria and A. Harter, "Parameterisation of a stochastic model for human face identification," in *Proceedings of the 2nd IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, 1994.