Machine Audition:

Principles, Algorithms and Systems

Wenwu Wang
University of Surrey, UK

Information Science REFERENCE

INFORMATION SCIENCE REFERENCE
Hershey · New York

Director of Editorial Content:	Kristin Klinger
Director of Book Publications:	Julia Mosemann
Acquisitions Editor:	Lindsay Johnston
Development Editor:	Joel Gamon
Publishing Assistants:	Casey Conapitski and Travis Gundrum
Typesetter:	Michael Brehm
Production Editor:	Jamie Snavely
Cover Design:	Lisa Tosheff

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Machine audition : principles, algorithms, and systems / Wenwu Wang, editor.
p. cm.
Includes bibliographical references and index.
Summary: "This book covers advances in algorithmic developments, theoretical frameworks, and experimental research findings to assist professionals who want an improved understanding about how to design algorithms for performing automatic analysis of audio signals, construct a computing system for understanding sound, and to learn how to build advanced human-computer interactive systems"--Provided by publisher. ISBN 978-1-61520-919-4 (hardcover) -- ISBN 978-1-61520-920-0 (ebook) 1. Computational auditory scene analysis. 2. Signal processing. 3. Auditory perception--Computer simulation. I. Wang, Wenwu, 1974- TK7881.4.M27 2010
006.4'5--dc22
2010010161

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this book is new, previously-unpublished material. The views expressed in this book are those of the authors, but not necessarily of the publisher.

Chapter 15

Instantaneous vs. Convolutive Non-Negative Matrix Factorization:

Models, Algorithms and Applications to Audio Pattern Separation

Wenwu Wang
University of Surrey, UK

ABSTRACT

Non-negative matrix factorization (NMF) is an emerging technique for data analysis and machine learning, which aims to find low-rank representations for non-negative data. Early works in NMF are mainly based on the instantaneous model, i.e. using a single basis matrix to represent the data. Recent works have shown that the instantaneous model may not be satisfactory for many audio application tasks. The convolutive NMF model, which has an advantage of revealing the temporal structure possessed by many signals, has been proposed. This chapter intends to provide a brief overview of the models and algorithms for both the instantaneous and the convolutive NMF, with a focus on the theoretical analysis and performance evaluation of the convolutive NMF algorithms, and their applications to audio pattern separation problems.

INTRODUCTION

Since the seminal paper published in 1999 by Lee and Seung, non-negative matrix factorization (NMF) has attracted tremendous research interests over the last decade. The earliest work in NMF is perhaps by (Paatero, 1997) and is then made popular by Lee and Seung due to their elegant multiplicative algorithms (Lee & Seung, 1999, Lee & Seung, 2001). The aim of NMF is to look for latent structures or features within a dataset, through the representation of a non-negative data matrix by a product of low rank matrices. It was found in (Lee & Seung, 1999) that NMF results in a “parts” based representation, due to the nonnegative constraint. This is because only additive operations are allowed in the learning process. Although later works in NMF may have mathematical operations that can lead to negative elements within the low-rank matrices, their

DOI: 10.4018/978-1-61520-919-4.ch015

non-negativity can be ensured by a projection operation (Zdenuk & Cichocki, 2007, Soltuz et al, 2008). Another interesting property with the NMF technique is that the decomposed low-rank matrices are usually sparse, and the degree of their sparseness can be explicitly controlled in the algorithm (Hoyer, 2004). Thanks to these promising properties, NMF has been applied to many problems in data analysis, signal processing, computer vision, and patter recognition, see, e.g. (Lee & Seung, 1999, Pauca et al, 2006, Smaragdis & Brown, 2003, Wang & Plumbley, 2005, Parry & Essa, 2007, FitzGerald et al, 2005, Wang et al, 2006, Zou et al, 2008, Wang et al, 2009, Cichocki et al, 2006b).

In machine audition and audio signal processing, NMF has also found applications in, for example, music transcription (Smaragdis & Brown, 2003, Wang et al, 2006) and audio source separation (Wang & Plumbley, 2005, Parry & Essa, 2007, FitzGerald et al, 2005, FitzGerald et al, 2006, Virtanen, 2007, Wang et al, 2009). In these applications, the raw audio data are usually transformed to the frequency domain to generate the spectrogram, i.e. the non-negative data matrix, which is then used as the input to the NMF algorithm. The instantaneous NMF model given in (Lee & Seung, 1999, Lee & Seung, 2001) has been shown to be satisfactory in certain tasks in audio applications provided that the spectral frequencies of the analyzed signal do not change dramatically over time (Smaragdis, 2004, Smaragdis, 2007, Wang, 2007, Wang et al, 2009). However, this is not a case for many realistic audio signals whose frequencies do vary with time. The main limitation with the instantaneous NMF model is that only a single basis function is used, and therefore is not sufficient to capture the temporal dependency of the frequency patterns within the signal. To address this issue, the convolutive NMF (or similar methods called shifted NMF) model has been introduced (Smaragdis, 2004, Smaragdis, 2007, Virtanen, 2007, FitzGerald et al, 2005, Morup et al, 2007, Schmidt & Morup, 2006, O'Grady & Pearlmutter, 2006, Wang, 2007, Wang et al, 2009). For the convolutive NMF, the data to be analyzed are modelled as a linear combination of shifted matrices, representing the time delays of multiple bases. Several algorithms have been developed based on this model, for example, the Kullback-Leibler (KL) divergence based multiplicative algorithm proposed in (Smaragdis, 2004, Smaragdis, 2007), the squared Euclidean distance based multiplicative algorithm proposed in (Wang, 2007, Wang et al, 2009), the two-dimensional deconvolution algorithms proposed in (Schmidt & Morup, 2006), the logarithmic scaled spectrogram decomposition algorithm in (FitzGerald et al, 2005), and the algorithm based on the constraints of the temporal continuity and sparseness of the signals in (Virtanen, 2007).

This chapter will briefly review the mathematical models for both instantaneous and convolutive NMF, some representative algorithms, and their applications to the machine audition problems, in particular, the problem of audio pattern separation and onset detection. This chapter also aims to serve as complementary material to our previous work in (Wang et al, 2009). To this end, we will provide a theoretical analysis of the convolutive NMF algorithm based on the squared Euclidean distance. These results can be readily extended to the KL divergence based algorithms. Moreover, we will provide several examples in addition to the simulations provided in (Wang et al, 2009). The remainder of the chapter is organised as follows. The next two sections will review the models and the algorithms of instantaneous and convolutive NMF, respectively. Then, we provide the theoretical analysis to the convolutive NMF algorithm based on the squared Euclidean distance. After this, we show some simulations to demonstrate the applicability of the NMF algorithms (both instantaneous and convolutive) to the machine audition problems including audio pattern separation and onset detection. In addition to the performance comparison between the three typical convolutive NMF algorithms, we will further compare their performance based on the relative reconstruction errors and the rejection ratio. Finally, we discuss future research directions in this area.

INSTANTANEOUS NMF

The mathematical model of the instantaneous NMF can be described as follows. Given a non-negative data matrix $\mathbf{X} \in \Re_{+}^{M\times N}$, find two matrices $\mathbf{W} \in \Re_{+}^{M\times R}$ and $\mathbf{H} \in \Re_{+}^{R\times N}$ such that $\mathbf{X} \approx \mathbf{WH}$, where the factorization rank R is generally chosen to be smaller than M(or N), or akin to $(M+N)R < MN$. In other words, NMF aims to map the given data from a higher dimensional space to a lower one. As a result, some redundancies within the data can be reduced and at the same time, some latent features can be extracted. In practice, such data can be an image (Lee & Seung, 1999), the spectrogram of an audio signal (Smaragdis & Brown, 2003), among many others, see e.g. (Berry, 2007) for a recent review. Several cost functions have been used in the literature for finding **W** and **H**, see e.g. (Paatero, 1997, Lee & Seung, 1999, Lee & Seung, 2001, Hoyer, 2004, Cichocki et al, 2006a, Dhillon & Sra, 2006). One frequently used criterion is based on the mean squared reconstruction error defined as follows

$$(\hat{\mathbf{W}}, \hat{\mathbf{H}}) = \arg\min_{\mathbf{W},\mathbf{H}} \frac{1}{2}\left\|\mathbf{X} - \mathbf{WH}\right\|_F^2 \tag{1}$$

where $\left\|*\right\|_F$ denotes the Frobenius norm, and $\hat{\mathbf{W}}$ and $\hat{\mathbf{H}}$ are the estimated optimal values of **W** and **H** (when the algorithm converges). It is also referred to as the squared Euclidean distance based criterion. Another criterion is based on the extended KL divergence,

$$(\hat{\mathbf{W}}, \hat{\mathbf{H}}) = \arg\min_{\mathbf{W},\mathbf{H}} \sum_{m=1}^{M}\sum_{n=1}^{N}\left\{\mathbf{X}\bullet\log\left[\frac{\mathbf{X}}{\mathbf{WH}}\right] - \mathbf{X} + \mathbf{WH}\right\} \tag{2}$$

where ● denotes the element-wise multiplication, and the division also operates in element wise. If we denote **WH** as $\hat{\mathbf{X}}$, then $\hat{\mathbf{X}}$ is the reconstructed data, which should ideally be equal to **X**. In practice, as shown in Equation (1) and (2), the difference between $\hat{\mathbf{X}}$ and **X** is used to find **W** and **H**.

To optimize the above cost functions, Lee and Seung have proposed simple yet efficient multiplicative algorithms based on the variable step-size normalization of each element of **W** and **H** (Lee & Seung, 1999, Lee & Seung, 2001), where **W** and **H** are updated alternately in each iteration, i.e. fixing **W**, updating **H**, then fixing **H** and updating **W**. Based on criterion (1), the update equations for **W** and **H** can be written as

$$\mathbf{H}^{q+1} = \mathbf{H}^{q} \bullet \frac{(\mathbf{W}^{q})^{T}\mathbf{X}}{(\mathbf{W}^{q})^{T}\mathbf{W}^{q}\mathbf{H}^{q}} \tag{3}$$

$$\mathbf{W}^{q+1} = \mathbf{W}^{q} \bullet \frac{\mathbf{X}(\mathbf{H}^{q+1})^{T}}{\mathbf{W}^{q}\mathbf{H}^{q+1}(\mathbf{H}^{q+1})^{T}} \tag{4}$$

where $(*)^T$denotes matrix transpose and q is the iteration number. Similarly, for criterion (2), we have the following update equations

$$\mathbf{H}^{q+1} = \mathbf{H}^{q} \bullet \frac{(\mathbf{W}^{q})^{T}\dfrac{\mathbf{X}}{\mathbf{W}^{q}\mathbf{H}^{q}}}{(\mathbf{W}^{q})^{T}\mathbf{E}} \tag{5}$$

$$\mathbf{W}^{q+1} = \mathbf{W}^{q} \bullet \frac{\dfrac{\mathbf{X}}{\mathbf{W}^{q}\mathbf{H}^{q}}(\mathbf{H}^{q+1})^{T}}{\mathbf{E}(\mathbf{H}^{q+1})^{T}} \tag{6}$$

where $\mathbf{E} \in \Re_{+}^{M \times N}$ is a matrix whose elements are all set to unity.

Since the publication of these multiplicative algorithms, there have been an increasing number of activities in developing new algorithms for NMF. These include using new cost functions (such as Csiszar's divergence, alpha and beta divergence, Cichocki et al, 2006a, Cichocki et al, 2006b, Cichocki et al, 2007), new adaptation algorithms (such as the projected gradient methods, alternating least squares (ALS) method and the conjugate gradient algorithm, Lin, 2007, Zdenuk & Cichocki, 2007, Kim et al, 2007, Wang & Zou, 2008), applying additional constraints (such as sparseness, smoothness, continuity, etc., Hoyer, 2004, Virtanen, 2003, Virtanen, 2007). For a review of recent development on NMF, please refer to e.g. (Albright et al, 2006, Berry et al, 2007).

Here we show the learning rules of ALS algorithm. The ALS method uses the following iterations to update **W** and **H**

$$\mathbf{H}^{q+1} = \left((\mathbf{W}^{q})^{T} \mathbf{W}^{q} \right)^{-1} (\mathbf{W}^{q})^{T} \mathbf{X} \tag{7}$$

$$\mathbf{W}^{q+1} = \mathbf{X} (\mathbf{H}^{q+1})^{T} \left(\mathbf{H}^{q+1} (\mathbf{H}^{q+1})^{T} \right)^{-1} \tag{8}$$

The matrix inverse used in above equations may result in negative elements within **W** and **H**. In practice, the negative elements are projected back to the non-negative orthant. It was found in (Soltuz et al, 2009) that the ALS algorithm has fast convergence rate, however, its convergence performance is not consistent. The algorithm suffers from instability and may diverge in practice. To improve its stability, one can combine the multiplicative algorithm due to (Lee & Seung, 2001) with the ALS algorithm, as suggested in (Soltuz et al, 2009). For example, **W** and **H** can be updated in the following way,

$$\mathbf{H}^{q+1} = \mathbf{H}^{q} \bullet \frac{(\mathbf{W}^{q})^{T} \mathbf{X}}{(\mathbf{W}^{q})^{T} \mathbf{W}^{q} \mathbf{H}^{q}} \tag{9}$$

$$\mathbf{W}^{q+1} = \mathbf{X} (\mathbf{H}^{q+1})^{T} \left(\mathbf{H}^{q+1} (\mathbf{H}^{q+1})^{T} \right)^{-1} \tag{10}$$

The hybrid algorithm provides a fast convergence rate and at the same time offers good convergence stability. More details of the theoretical and numerical analysis of this algorithm can be found in (Soltuz et al, 2009).

CONVOLUTIVE NMF

The instantaneous NMF has limitations in dealing with many non-stationary signals whose frequencies change dramatically over time, since only a single basis is used in the model. To address this issue, Smaragdis extended the standard (instantaneous) NMF model to the convolutive case (Smaragdis, 2004). Rather than using $\hat{\mathbf{X}} = \mathbf{W}\mathbf{H}$, $\hat{\mathbf{X}}$, is represented by a sum of shifted matrix products, i.e.

$$\hat{\mathbf{X}} = \sum_{p=0}^{P-1} \mathbf{W}(p) \overset{p\rightarrow}{\mathbf{H}} \tag{11}$$

where $\mathbf{W}(p) \in \Re_{+}^{M \times R}$, $p = 1, \cdots, P-1$, are a set of bases, and $\overset{p\rightarrow}{\mathbf{H}}$ shifts the columns of **H** by p spots to the right. Similarly, $\overset{\leftarrow p}{\mathbf{H}}$ shifts the columns of **H** by p spots to the left. The shifts are non-circular, which means the elements of the columns shifted in from outside the matrix will be set to zeros (Smaragdis, 2004, Wang, 2007, Wang et al, 2009). For example, suppose

$$\mathbf{H} = \begin{bmatrix} 2 & 1 & 7 \\ 9 & 9 & 3 \\ 4 & 5 & 8 \end{bmatrix}$$

Then

$$\overset{0\rightarrow}{\mathbf{H}} = \begin{bmatrix} 2 & 1 & 7 \\ 9 & 9 & 3 \\ 4 & 5 & 8 \end{bmatrix}, \overset{1\rightarrow}{\mathbf{H}} = \begin{bmatrix} 0 & 2 & 1 \\ 0 & 9 & 9 \\ 0 & 4 & 5 \end{bmatrix},$$

$$\overset{2\rightarrow}{\mathbf{H}} = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 9 \\ 0 & 0 & 4 \end{bmatrix}, \overset{3\rightarrow}{\mathbf{H}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\overset{\leftarrow 0}{\mathbf{H}} = \begin{bmatrix} 2 & 1 & 7 \\ 9 & 9 & 3 \\ 4 & 5 & 8 \end{bmatrix}, \overset{\leftarrow 1}{\mathbf{H}} = \begin{bmatrix} 1 & 7 & 0 \\ 9 & 3 & 0 \\ 5 & 8 & 0 \end{bmatrix},$$

$$\overset{\leftarrow 2}{\mathbf{H}} = \begin{bmatrix} 7 & 0 & 0 \\ 3 & 0 & 0 \\ 8 & 0 & 0 \end{bmatrix}, \overset{\leftarrow 3}{\mathbf{H}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Using the convolutive model (11) and the extended KL divergence based criterion (2), Smaragdis obtained the following multiplicative learning algorithm

$$\mathbf{H}^{q+1} = \mathbf{H}^{q} \bullet \frac{(\mathbf{W}^{q}(p))^{T} \left(\dfrac{\overset{\leftarrow p}{\mathbf{X}}}{\hat{\mathbf{X}}^{q}} \right)}{(\mathbf{W}^{q}(p))^{T} \bullet} \tag{12}$$

$$\mathbf{W}^{q+1}(p) = \mathbf{W}^{q}(p) \bullet \frac{\left(\dfrac{\mathbf{X}}{\hat{\mathbf{X}}^{q}} \right) (\overset{p\rightarrow}{\mathbf{H}^{q+1}})^{T}}{\bullet\, (\overset{p\rightarrow}{\mathbf{H}^{q+1}})^{T}} \tag{13}$$

As in our previous work (Wang, 2009), we refer to this algorithm as ConvNMF-KL in this chapter.

Recently, using the same convolutive model (11), we have derived a new algorithm using the squared Euclidean distance based criterion (1), see details in (Wang et al, 2009). In this algorithm, the update equations for **W**(p) and **H** are given as

$$\mathbf{W}^{q+1}(p) = \mathbf{W}^{q}(p) \bullet \frac{\mathbf{X} \left(\overset{p\rightarrow}{\mathbf{H}^{q}} \right)^{T}}{\hat{\mathbf{X}}^{q} \left(\overset{p\rightarrow}{\mathbf{H}^{q}} \right)^{T}} \tag{14}$$

$$\mathbf{H}^{q+1} = \mathbf{H}^{q} \bullet \frac{(\mathbf{W}^{q+1}(p))^{T} \overset{\leftarrow p}{\mathbf{X}}}{(\mathbf{W}^{q+1}(p))^{T} \overset{\leftarrow p}{\hat{\mathbf{X}}}} \tag{15}$$

The update Equation (15) may lead to a biased estimate of **H**. To address this issue, in practice, **H** can be further modified as

$$\mathbf{H}^{q+1} = \frac{1}{P} \sum_{p=0}^{P-1} \left(\mathbf{H}^{q} \bullet \frac{(\mathbf{W}^{q+1}(p))^{T} \overset{\leftarrow p}{\mathbf{X}}}{(\mathbf{W}^{q+1}(p))^{T} \overset{\leftarrow p}{\hat{\mathbf{X}}}} \right) \tag{16}$$

To improve the computational efficiency of the proposed algorithm, we have introduced a recursive update method for $\hat{\mathbf{X}}$ as follows

$$\hat{\mathbf{X}}^{q} = \hat{\mathbf{X}}^{q} - \mathbf{W}^{q}(p) \overset{p\rightarrow}{\mathbf{H}^{q}} + \mathbf{W}^{q+1}(p) \overset{p\rightarrow}{\mathbf{H}^{q}} \quad (p = 0, \cdots, P-1) \tag{17}$$

The subtraction used in (17) may result in negative values of the elements in $\hat{\mathbf{X}}^{q}$. We use the following projection to prevent this

$$\hat{\mathbf{X}}^{q}_{i,j} = \max(\varepsilon, \hat{\mathbf{X}}^{q}_{i,j}) \tag{18}$$

where $\hat{\mathbf{X}}^{q}_{i,j}$ is the *ij*-th element of the matrix $\hat{\mathbf{X}}$ at iteration q, and ε is a floor constant, and typically, we choose $\varepsilon = 10^{-9}$ in our implementations. Same as in our previous work (Wang et al, 2009), we denote the algorithm described above as ConvNMF-ED. The implementation details of ConvNMF-ED can be found in (Wang et al, 2009) and are omitted in this chapter.

It is worth noting that there is another method considering the convolutive model, i.e. SNMF2D developed in (Schmidt & Morup, 2006, Morup & Schmidt, 2006). This algorithm can be regarded as an extension of Smaragdis's work by considering a two-dimensional deconvolution scheme, together with sparseness constraints (Morup & Schmidt, 2006), where both the extended KL divergence and the least squares criterion are considered. For their least square criterion based approach, denoted as SNMF2D-LS in this chapter, the shifted versions of $\mathbf{W}^q$and $\mathbf{H}^q$ at all time lags $p = 0, \ldots, P - 1$ are used for updating $\mathbf{W}^q(p)$ and $\mathbf{H}^q(p)$, with an individual time lag at each iteration. The update equation for $\mathbf{H}^q$ is given as

$$\mathbf{H}^{q+1} = \mathbf{H}^q \bullet \frac{\left(W^{q+1}(0)\right)^T \overset{\leftarrow p}{\mathbf{X}} + \cdots + \left(W^{q+1}(P-1)\right)^T \overset{\leftarrow p}{\mathbf{X}}}{\left(W^{q+1}(0)\right)^T \overset{\leftarrow p}{\hat{\mathbf{X}}} + \cdots + \left(W^{q+1}(P-1)\right)^T \overset{\leftarrow p}{\hat{\mathbf{X}}}} \tag{19}$$

The advantage of this formulation is the increased sparseness that may be achieved for the decomposition matrices. For audio pattern separation purpose, however, this representation may break the structure of audio objects which makes event or onset detection directly from $\mathbf{W}^q(p)$ and $\mathbf{H}^q$ even more difficult (Wang et al, 2009). Another issue with this formulation is the over-shifting effect (Wang et al, 2009) where the time-frequency signature in the data has been shifted more than it actually requires in the sense of audio object separation. Also the computational load with the above formulation is higher as compared with the ConvNMF-ED algorithm. In this chapter, as a compliment to the results in (Wang et al, 2009), we will show additional simulations and comparisons between the above methods in the subsequent sections.

As in instantaneous NMF, one can also consider sparseness constraint within the convolutive NMF algorithm. For example, we can enforce sparseness constraint on $\mathbf{H}$ using the following cost function,

$$(\hat{\mathbf{W}}, \hat{\mathbf{H}}) = \arg\min_{\mathbf{W},\mathbf{H}} \left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_F^2 + \lambda \sum_{ij} \mathbf{H}_{ij} \tag{20}$$

where $\hat{\mathbf{X}}$ takes the form of (11), and λ is a regularization constant which controls the amount of sparseness constraints. To optimize this cost function, we can use the same expression as (14) for the update of $\mathbf{W}(p)$. However, $\mathbf{H}$ needs to be updated as follows (Wang, 2008)

$$\mathbf{H}^{q+1} = \mathbf{H}^q \bullet \frac{\left((\mathbf{W}^{q+1}(p))^T \overset{\leftarrow p}{\mathbf{X}}\right)}{\left((\mathbf{W}^{q+1}(p))^T \overset{\leftarrow p}{\hat{\mathbf{X}}^q} + \lambda \check{\mathbf{z}}\right)} \tag{21}$$

CONVERGENCE ANALYSIS OF THE CONVOLUTIVE NMF ALGORITHM

The exact convergence analysis of the proposed algorithm would be difficult. However, the overall performance can be approximated by the key updating Equations (14) and (15). As we know that, when $P = 1$, ConvNMF-ED is approximately equivalent to the instantaneous NMF (i.e. the algorithm based on update Equations (3) and (4)). This implies that we can effectively follow the method used in (Lee & Seung, 2001) for the convergence analysis of the proposed algorithm in terms of (14) and (15). Similarly, we have the following lemma.

Lemma 1: The squared Euclidean distance $\Im$ is non-increasing under the learning rules (14) and (15).

Proof: Suppose $G(\mathbf{w}, \mathbf{w}^q)$ is an auxiliary function for $\Im = \frac{1}{2}\left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_F^2$, then according to (Lee & Seung, 2001), the conditions $G(\mathbf{w}, \mathbf{w}^q) \geq \Im(\mathbf{w})$, $G(\mathbf{w}, \mathbf{w}) = \Im(\mathbf{w})$ should be satisfied, and $\Im$ is non-increasing under the update $\mathbf{w}^{q+1} = \arg\min G(\mathbf{w}, \mathbf{w}^q)$, where $\mathbf{w}$ is a vector derived from the matrix **W**(p) by stacking its columns together into one column vector, i.e. the vectorization of matrix **W**(p), and q is again the iteration index. Likewise, **w** can be replaced by the vector **h** in order to prove the convergence property of the learning rule of **H**, where **h** can be obtained in the same way as **w** using vectorization. Correspondingly, $\Im$ is represented as a function of **w**, i.e. $\Im(\mathbf{w})$, instead of **W**(p). To proceed the proof, we need the following derivatives, $\frac{\partial \hat{\mathbf{X}}_{i,j}}{\partial \mathbf{W}_{m,n}(p)}$, $\frac{\partial \Im}{\partial \mathbf{W}_{m,n}(p)}$, and the components of the Hessian tensor

$$\Pi_{m,n,m',n'}(p) = \frac{\partial^2 \Im}{\partial \mathbf{W}_{m,n}(p) \partial \mathbf{W}_{m',n'}(p)},$$

where the sub-script denotes the specific element of a matrix; for example, $\mathbf{W}_{m,n}(p)$ represents the mn-th element of the matrix **W**(p), and the same notation is used for other matrices throughout the chapter.

In terms of Equation (11), we have the following derivative (see the Appendix for its derivation)

$$\frac{\partial \hat{\mathbf{X}}_{i,j}}{\partial \mathbf{W}_{m,n}(p)} = \delta_{i,m} \overset{p\rightarrow}{\mathbf{H}}_{n,j} \tag{22}$$

where $\delta_{i,m}$ is denoted as

$$\delta_{i,m} = \begin{cases} 1, & i = m \\ 0, & i \neq m \end{cases} \tag{23}$$

Similary, according to Equation (1) and (22), we have (see the Appendix for details)

$$\frac{\partial \Im}{\partial \mathbf{W}_{m,n}(p)} = \sum_j (\hat{\mathbf{X}}_{m,j} - \mathbf{X}_{m,j}) \overset{p\rightarrow}{\mathbf{H}}_{n,j} \tag{24}$$

Based on Equations (1) (22) (24), the components of the Hessian tensor $\Pi_{m,n,m',n'}(p)$ can be derived as (refer to the Appendix for details)

$$\Pi_{m,n,m',n'}(p) = \sum_j \delta_{m,m'} \overset{p\rightarrow}{\mathbf{H}}_{n,j} \overset{p\rightarrow}{\mathbf{H}}_{n',j} \tag{25}$$

Let $\lambda = \{m, n\}$, $\lambda' = \{m', n'\}$, so that the tensor $\Pi_{m,n,m',n'}(p)$ can be compressed as a matrix with elements denoted as $\mathbf{T}_{\lambda,\lambda'}(p)$. With these derivatives, we are now ready for the whole proof using a procedure similar to that in (Lee & Seung, 2001, Morup & Schmidt, 2006). First, we can expand $\Im$ in terms of a second order Taylor series, i.e.

$$\Im(\mathbf{w}) = \Im(\mathbf{w}^q) + (\mathbf{w} - \mathbf{w}^q)\frac{\partial \Im}{\partial \mathbf{W}(p)} + \frac{1}{2}(\mathbf{w} - \mathbf{w}^q)^T \mathbf{T}(\mathbf{w} - \mathbf{w}^q) \tag{26}$$

Then, let $G(\mathbf{w}, \mathbf{w}^q)$ take the following form

$$G(\mathbf{w}, \mathbf{w}^q) = \Im(\mathbf{w}^q) + (\mathbf{w} - \mathbf{w}^q)\frac{\partial \Im}{\partial \mathbf{W}(p)} + \frac{1}{2}(\mathbf{w} - \mathbf{w}^q)^T \mathbf{K}(\mathbf{w}^q)(\mathbf{w} - \mathbf{w}^q) \tag{27}$$

where $\mathbf{K}(\mathbf{w}^q)$ is a diagonal matrix defined as

$$\mathbf{K}_{\lambda,\lambda'} = \delta_{\lambda,\lambda'} \frac{(\mathbf{T}\mathbf{w}^q)_\lambda}{(\mathbf{w}^q)_\lambda} \tag{28}$$

According to the same method as in (Lee & Seung, 2001), it is straightforward to show that

$$(\mathbf{w} - \mathbf{w}^q)(\mathbf{K}(\mathbf{w}^q) - \mathbf{T})(\mathbf{w} - \mathbf{w}^q) \geq 0 \tag{29}$$

Therefore, we have $G(\mathbf{w}, \mathbf{w}^q) \geq \Im(\mathbf{w})$ in terms of $G(\mathbf{w}, \mathbf{w}^q) - \Im(\mathbf{w})$ computed from Equations (26) (27). It is also straightforward to prove that $G(\mathbf{w}, \mathbf{w}) = \Im(\mathbf{w})$ in terms of Equations (26) and (27). Finally, we need to show that the learning rules (14) were obtained when the gradient of $G(\mathbf{w}, \mathbf{w}^q)$ with respect to **w** equals to zero, i.e.

$$\frac{\partial G}{\partial \mathbf{w}} = \frac{\partial G}{\partial \mathbf{W}_{m,n}(p)} = 0 \tag{30}$$

According to Equation (25), we have (see the derivation in the Appendix)

$$(\mathbf{Tw})_{m,n} = \left(\hat{\mathbf{X}} \left(\overset{p\rightarrow}{\mathbf{H}} \right)^T \right)_{m,n} \tag{31}$$

Expanding (30) and incorporating (31), we obtain the following element-wise adaptation equation

$$\mathbf{W}_{m,n}^{q+1}(p) = \mathbf{W}_{m,n}^{q}(p) - \frac{\mathbf{W}_{m,n}^{q}(p)}{\sum_j \overset{p\rightarrow}{\mathbf{H}}_{n,j} \hat{\mathbf{X}}_{m,j}} \sum_j (\hat{\mathbf{X}}_{m,j} - \mathbf{X}_{m,j}) \overset{p\rightarrow}{\mathbf{H}}_{n,j} = \mathbf{W}_{m,n}^{q}(p) \frac{\left(\mathbf{X} \left(\overset{p\rightarrow}{\mathbf{H}} \right)^T \right)_{m,n}}{\left(\hat{\mathbf{X}} \left(\overset{p\rightarrow}{\mathbf{H}} \right)^T \right)_{m,n}} \tag{32}$$

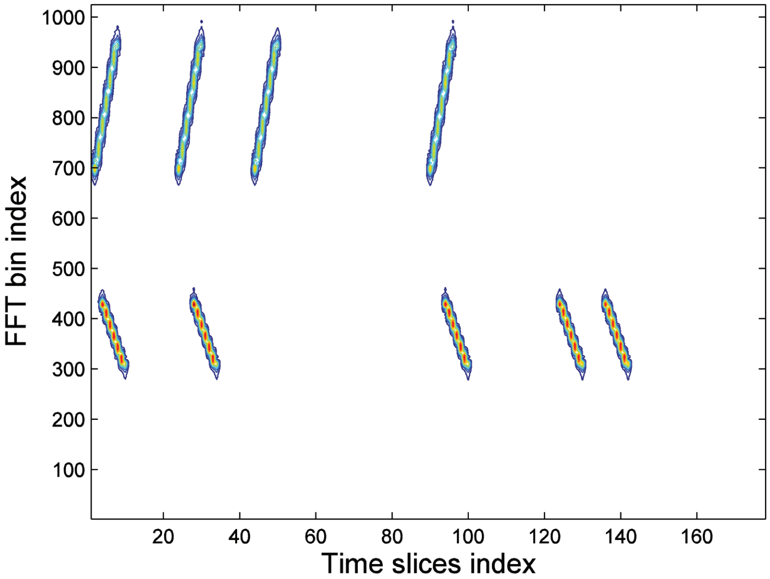
Apparently, Equation (32) is an element-wise operation of (14). Consequently, $\Im$ is non-increasing under the update (32), as it is obtained by the minimization of $G(\mathbf{w}, \mathbf{w}^q)$. Similarly, $\Im$ is non-increasing under the update Equation (15). This concludes the proof of Lemma 1. Obviously, the distance $\Im$ is invariant under these updates if and only if **W**(*p*) and **H** are at a limit point of the distance. However, as in instantaneous NMF, whether any limit point is always stationary remains an open issue, see, e.g. (Lin, 2007) and therefore is an interesting topic for future research.

APPLICATIONS TO AUDIO PATTERN SEPARATION

In this section, we show an example of applying the ConvNMF-ED algorithm to the audio object separation problem using artificially generated audio signals. More application examples to real music audio signals can be found in (Wang, 2007, Wang et al, 2009), and are not included in this chapter. We generate the audio signal in the same way as used in our previous work (Wang, 2007). First, we generated two audio signals, with one containing five repeating patterns whose frequencies changing linearly with time from 320Hz to 270Hz, and the other containing four repeating patterns whose frequencies change linearly from 500Hz to 600Hz. The sampling frequency f_s for both signals is 1500Hz. These two signals were added together to generate a mixture. The length of the signal is 30 seconds. Then, this mixture was transformed into the frequency domain by the procedure described in (Wang et al, 2006, Wang, 2007, Wang et al, 2008, Wang et al, 2009), where the frame length *T* of the fast Fourier transform (FFT) was set to 2048 samples, i.e., the frequency resolution is approximately 0.73Hz. The signal was segmented by a Hamming window with the window size being set to 600 samples (400ms), and the time shift to 250 samples (approximately 167ms), that is, an overlap between the neighboring frames was used. Each segment was zero-padded to have the same size as *T* for the FFT operation. The generated matrix **X** is visualized in Figure 1. Note that the parameters used for generating **X** are identical to those used in (Wang, 2007).

The ConvNMF-ED algorithm was then applied to **X**. In this algorithm, the factorization rank *R* was set to two, i.e., exactly the same as the total number of the signals in the mixture. The matrices **W**(*p*) and **H** were initialized as the absolute values of random matrices with elements drawn from a standardized Gaussian probability density function. *P* was set to six (in order for the object to be separated, *P* should be sufficiently large to cover the length of the object in the signal). All tests were run on a computer whose CPU speed is 1.8GHz. Figure 2 and Figure 3 show $\mathbf{H}^o$ and $\mathbf{W}^o(p)$, i.e. optimal values of **H** and **W**(*p*), respectively. It is clear from these figures that the audio objects

Figure 1. The contour plot of the magnitude spectrum matrix **X** *generated from the artificial audio data*



with repeating patterns are successfully separated by the ConvNMF-ED algorithm, with $\mathbf{W}^o(p)$ being the time-frequency representation of the repeating patterns, and $\mathbf{H}^o$ containing the temporal structure of these patterns, i.e., the occurrence time of individual patterns. We should note that the instantaneous NMF described by the learning rules (3) and (4) (as well as (5) and (6)), however, totally fails for separating the audio objects in these tests. ConvNMF-KL offers similar results to ConvNMF-ED, see e.g. (Wang, 2007, Wang et al 2009) for comparisons. We have extensively tested the algorithm for different set-ups of the parameters, including other randomly initialized matrices **W** and **H**, and found similar separation performance.

APPLICATIONS TO MUSIC ONSET DETECTION

Onset detection is an important issue for machine perception of music audio signal. It aims to detect the starting point of a noticeable change in intensity, pitch and timbre of musical sound. It usually involves several steps including preprocessing, construction of detection function and peak picking. We have shown in (Wang et al, 2008) that the linear temporal bases obtained by an NMF algorithm (e.g. Lee & Seung 2001) can be used to construct a detection function. An advantage of constructing the detection function using the NMF bases is that no prior knowledge or statistical information is required. We have demonstrated in (Wang et al, 2006, Wang et al, 2008) that different types of detection functions can be constructed from $\mathbf{H}^o$, including the first-order difference function, the psychoacoustically motivated relative difference function, and the constant-balanced relative difference function. Recently, we have also shown application examples of convolutive NMF algorithms for onset detection, see e.g. (Wang et al, 2009), where we have compared the performance of applying different convolutive NMF algorithms to onset detection. Details of these results can be found in (Wang et al, 2009) and will not be exhaustively repeated in this chapter.

Figure 2. Visualisation of the two rows of matrix $\mathbf{H}^o$ *with each row in one sub-plot*

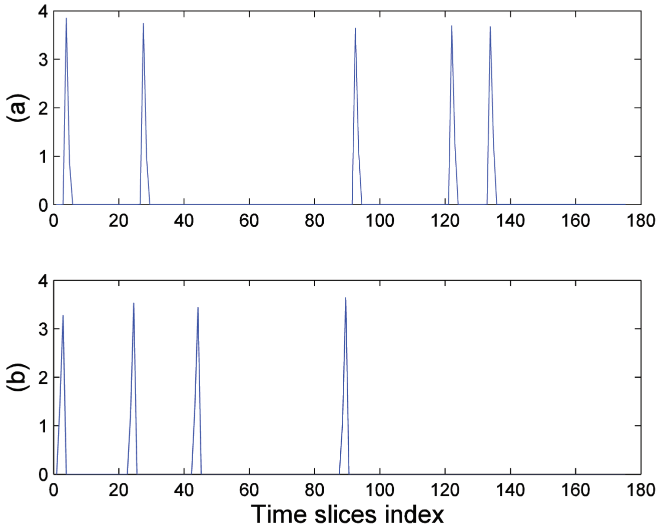
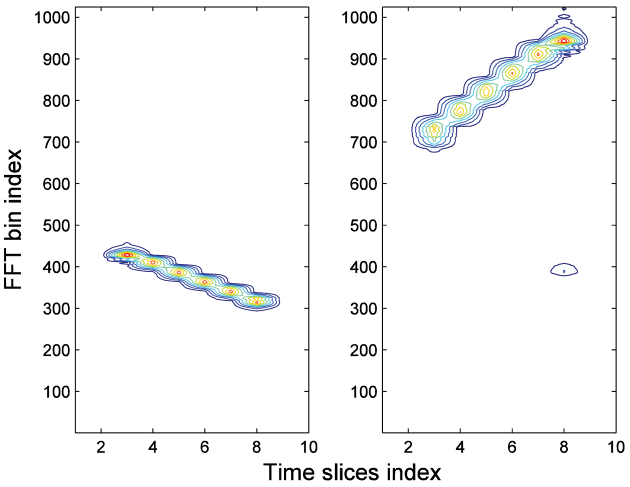


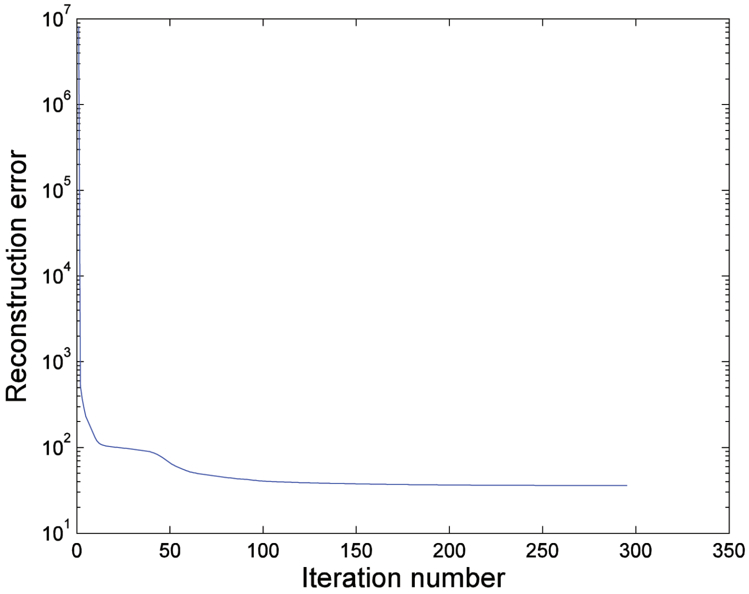
Figure 3. Visualisation of the two columns of all matrices $\mathbf{W}^o(p)$*,* $p = 0, \ldots, 5$ *as a collection, with each column visualized in one sub-plot*



EVALUATIONS OF THE CONVOLUTIVE NMF ALGORITHMS

In this section, we evaluate the performance of the three convolutive NMF algorithms, i.e. ConvNMF-ED, ConvNMF-KL and SNMF2D-LS. In (Wang et al, 2009), we have already evaluated the three algorithms from several aspects including convergence performance, computational efficiency, and note onset detection performance. Here, we intend to provide more evaluation results, some of which are complementary to those given in (Wang et al, 2009). For example, Figure 4 shows a typical convergence curve of the ConvNMF-ED

Figure 4. A typical convergence curve of the ConvNMF-ED algorithm measured by the reconstruction error versus the iteration number, where the reconstruction error is the absolute estimation error of $\hat{\mathbf{X}}$



algorithm obtained by a single run of the algorithm with a random initialization, while a comparison of the average convergence performance between the three algorithms is given in (Wang et al, 2009).

Now, we study two more aspects of the three algorithms using the following performance indices. One is the rejection ratio (RR). Let us represent $\hat{\mathbf{X}}$ as the combination of R factorized components, i.e.

$$\hat{\mathbf{X}} = \sum_{i=1}^{R} \hat{\mathbf{X}}(i) \tag{33}$$

Then, we can define the *RR* as follows

$$RR(\mathrm{dB}) = 10\log_{10}\left(\sum_{\forall j \neq i} cor\left(\hat{\mathbf{X}}(i), \hat{\mathbf{X}}(j)\right)\right) \tag{34}$$

where *cor* denotes the correlation. This performance index can measure approximately the accuracy of the separation performance for which a lower value represents a better performance. The other is the relative estimation error (REE),

$$REE(\mathrm{dB}) = 10\log_{10}\frac{\left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_F}{\left\|\mathbf{X}\right\|_F} \tag{35}$$

This performance index is less sensitive to the signal dynamics as compared with the absolute estimation error due to the adopted normalization. It measures approximately the accuracy of the factorization and a lower value represents a better performance. We ran ConvNMF-ED, ConvNMF-KL, and SNMF2D-LS for five random tests for each *T*, where *T* is the FFT frame size, and was set to be 256, 512, 1024, 2048 and 4096 respectively. Note that the results shown in (Wang et al, 2009) were based on 50 (instead of 5) random tests. The results of these five tests, together with their average are shown in Figure 5 and Figure 6, respectively. Several interesting points can be

Figure 5. The RR comparison between the algorithms ConvNMF-ED (a), ConvNMF-KL (b) and SNMF2D-LS (c). The FFF frame length T was chosen to be 256, 512, 1024, 2048, and 4096 respectively. For each T, five random tests were performed, and the RR was plotted as the average of the five tests, with individual test results plotted on the error bars.

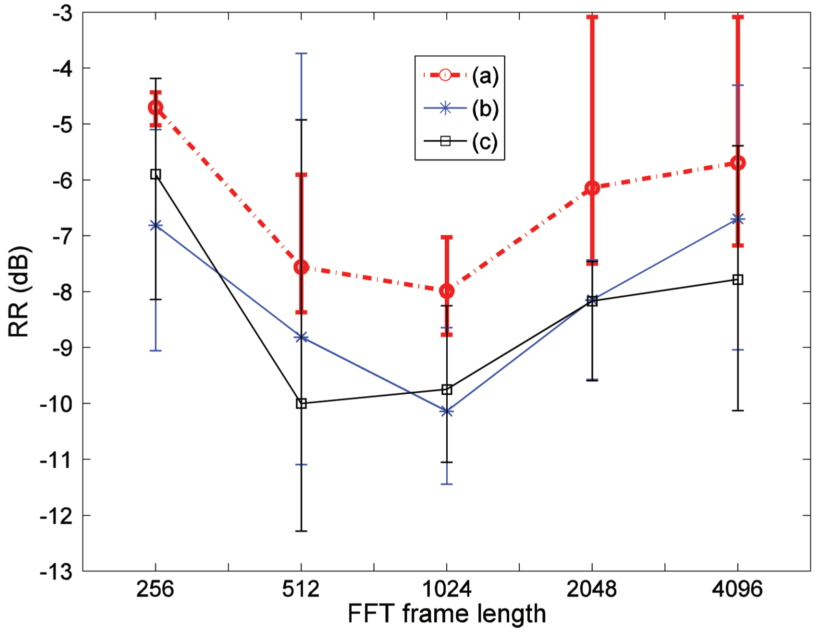
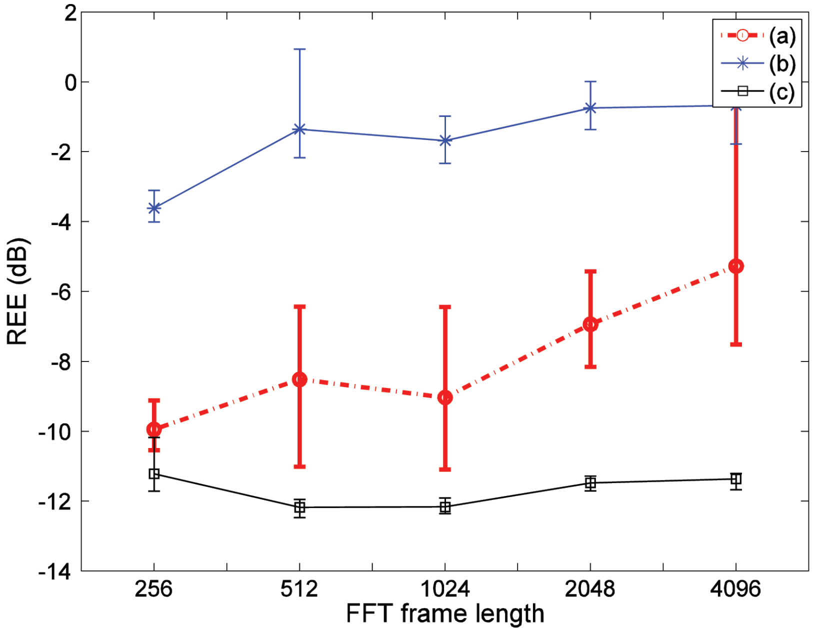


Figure 6. The REE comparison between the algorithms ConvNMF-ED (a), ConvNMF-KL (b) and SNMF2D-LS (c). The FFF frame length T was chosen to be 256, 512, 1024, 2048, and 4096 respectively. For each T, five random tests were performed, and the RR was plotted as the average of the five tests, with individual test results plotted on the error bars.



observed from these figures. First, both ConvNMF-ED and ConvNMF-KL are relatively sensitive to different initializations, which is a common issue for many NMF algorithms and how to find a performance independent initialization method remains an open problem. Second, from the test results, we notice that the algorithm ConvNMF-ED performs approximately equally well as ConvNMF-KL, although it is less accurate in terms of *RR* measurement. This suggests that the KL divergence may be advantageous for the separation of signals in the convolutive case. However, according to the *REE* measurement, ConvNMF-ED performs much better for reconstructing the original data. These observations somehow coincide with the findings for instantaneous NMF algorithms. One thing to note is that *RR* can be informative for the performance evaluation of signal separation. Therefore, SNMF2D-LS performs best in this experiment from the viewpoint of signal separation. However, it is clear from the results shown in our previous work in (Wang et al, 2009) that note events represented by $\mathbf{W}^o(p)$ and $\mathbf{H}^o$ (optimal values of $\mathbf{W}(p)$ and $\mathbf{H}$) obtained by the SNMF2D-LS algorithm are actually far from similar to the original events (e.g. the onset locations and the time-frequency signatures). This is because $\hat{\mathbf{X}}$ is a convolution of $\mathbf{W}^o(p)$ and $\mathbf{H}^o$, and consequently $\hat{\mathbf{X}}$ remains unchanged if both $\mathbf{W}^o(p)$ and $\mathbf{H}^o$ are over-shifted to the same extent. This implies that even though an algorithm reconstructs $\hat{\mathbf{X}}$ perfectly close to the original data $\mathbf{X}$, the obtained decomposition $\mathbf{W}^o(p)$ and $\mathbf{H}^o$ may not provide a meaningful interpretation to the original data. As a consequence, the *REE* and *RR* reveal only a part of the picture of the behavior of the algorithms.

FUTURE RESEARCH DIRECTIONS

Although NMF has shown to be useful for audio pattern separation (more broadly machine audio perception), there are still many open issues that require more research efforts. One of them is automatic rank selection. The decomposition rank is an important parameter for the application of an NMF algorithm. Its selection affects the results that can be achieved by the NMF algorithm and how the results might be interpreted. The convolutive NMF model involves the multiplications and additions of the multiple delayed components, current algorithms seem to be unsuitable for real-time applications, and more computationally efficient algorithms are required for such an application scenario. Most existing algorithms process the signal as a whole block. This may be a problem for long audio signals, as the generated non-negative matrix from the long signal can be of a high dimension. It is therefore desirable if we could develop adaptive or sequential algorithms to process the signals in shorter blocks and then apply the NMF algorithms for each of these blocks.

REFERENCES

Albright, R. Cox, J., Duling, D., Langville, A. & Meyer, C. (2006). Algorithms, initializations, and convergence for the nonnegative matrix factorization. *NCSU Technical Report Math 81706*.

Berry, M., Browne, M., Langville, M., Pauca, P., & Plemmons, R. (2007). *Algorithms and applications for approximate nonnegative matrix factorization*. Computational Statistics and Data Analysis.

Cichocki, A., Zdunek, R., & Amari, S. (2006a). Csiszar's divergences for non-negative matrix factorization: family of new algorithms. *Spinger Lecture Notes in Computer Science*, *3889*, 32–39. doi:10.1007/11679363_5

Cichocki, A., Zdunek, R., & Amari, S. (2006b). New algorithms for non-negative matrix factorization in applications to blind source separation. In *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Process.* (Vol. 4, pp. 621-624). Toulouse, France.

Cichocki, A., Zdunek, R., Choi, S., Plemmons, R., & Amari, S. (2007). Non-negative tensor factorization using alfa and beta divergences. *Proc. Int. Conf. on Acoustics, Speech, and Signal Process.* (pp. 1393-1396), Honolulu, Hawaii, USA.

Dhillon, I. S., & Sra, S. (2006). Generalized non-negative matrix approximations with Bregman divergences. In Y. Weiss, B. Schölkopf, and J. Platt, (Eds). *Advances in Neural Information Processing 18* (in Proc. NIPS 2006). Cambridge, MA: MIT Press.

FitzGerald, D., Cranitch, M., & Coyle, E. (2005). Shifted non-negative matrix factorization for sound source separation. In *Proc. IEEE Int. Workshop on Statistical Signal Process.* (pp.1132-1137), Bordeaux, France.

FitzGerald, D., Cranitch, M., & Coyle, E. (2006). Sound source separation using shifted non-negative tensor factorization. In *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process.* (Vol. 4, pp. 653-656).

Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, *5*, 1457–1469.

Kim, D., Sra, S., & Dhillon, I. S. (2007). Fast Newton-type methods for the least squares non-negative matrix approximation Problem. In *Proc. of the 6th SIAM Int. Conf. on Data Mining* (pp. 343-354).

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*, 788–791. doi:10.1038/44565

Lee, D. D., & Seung, H. S. (2001). Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing 13* (in Proc. NIPS 2000).Cambridge, MA: MIT Press.

Lin, C.-J. (2007). Projected gradient methods for non-negative matrix factorization. *Neural Computation*, *19*, 2756–2779. doi:10.1162/neco.2007.19.10.2756

Morup, M., Madsen, K. H., & Hansen, L. K. (2007). Shifted non-negative matrix factorization. In *Proc. IEEE Int. Workshop on Machine Learning for Signal Process* (pp. 427-432). Maynooth, Ireland.

Morup, M., & Schmidt, M. N. (2006). *Sparse non-negative matrix factor 2D deconvolution. Technical Report*. Technical University of Denmark.

O'Grady, P. D., & Pearlmutter, B. A. (2006). Convolutive non-negative matrix factorisation with a sparseness constraint. In *Proc. IEEE Int. Workshop on Machine Learning for Signal Process* (pp. 427-432), Maynooth, Ireland.

Paatero, P. (1997). Least squares formulation of robust non-negative factor analysis. *Chemometrics and Intelligent Laboratory Systems*, *37*, 23–35. doi:10.1016/S0169-7439(96)00044-5

Parry, R. M., & Essa, I. (2007). Incorporating phase information for source separation via spectrogram factorization. In *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process.* (Vol. 2, pp. 661-664). Honolulu, Hawaii, USA.

Pauca, V. P., Piper, J., & Plemmons, R. (2006). Non-negative matrix factorization for spectral data analysis. *Linear Algebra and Its Applications*, *416*(1), 29–47. doi:10.1016/j.laa.2005.06.025

Schmidt, M. N., & Morup, M. (2006). Nonnegative matrix factor 2D deconvolution for blind single channel source separation. In *Proc. 6th Int. Conf. on Independent Component Analysis and Blind Signal Separation* (pp. 700-707), Charleston, SC, USA.

Smaragdis, P. (2004). Non-negative matrix factor deconvolution, extraction of multiple sound sources from monophonic inputs. In *Proc. 5th Int. Conf. on Independent Component Analysis and Blind Signal Separation* (LNCS 3195, pp.494-499), Granada, Spain.).

Smaragdis, P. (2007). Convolutive speech bases and their application to supervised speech separation. *IEEE Trans. Audio Speech and Language Processing*, *15*(1), 1–12. doi:10.1109/TASL.2006.876726

Smaragdis, P., & Brown, J. C. (2003). Nonnegative matrix factorization for polyphonic music transcription. In *IEEE Int. Workshop on Applications of Signal Process. to Audio and Acoustics* (pp. 177-180). New Paltz, NY.

Soltuz, S., Wang, W., & Jackson, P. (2009). A hybrid iterative algorithm for non-negative matrix factorization. In *Proc. IEEE Int. Workshop on Statistical Signal Processing* (pp. 409-412).

Virtanen, T. (2003). Sound source separation using sparse coding with temporal continuity objective. In *Proc. Int. Comput. Music Conf.* (pp. 231-234), Singapore.

Virtanen, T. (2007). Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criterion. *IEEE Trans. Audio, Speech, and Language Processing*, *15*(3), 1066–1074. doi:10.1109/TASL.2006.885253

Wang, B., & Plumbley, M. D. (2005). Musical audio stream separation by non-negative matrix factorization. In *Proc. DMRN Summer Conf.* Glasgow, UK

Wang, W. (2007). Squared Euclidean distance based convolutive non-negative matrix factorization with multiplicative learning rules for audio pattern separation. In *Proc. IEEE Int. Symp. on Signal Proces. and Info. Tech.*, Cairo, Egypt.

Wang, W. (2008). Convolutive non-negative sparse coding. In *Proc. International Joint Conference on Neural Networks* (pp. 3681-3684). Hong Kong, China.

Wang, W., Cichocki, A., & Chambers, J. A. (2009). A multiplicative algorithm for convolutive non-negative matrix factorization based on squared Euclidean distance. In *IEEE Trans. on Signal Processing*, *57*(7), 2858-2864.

Wang, W., Luo, Y., Sanei, S., & Chambers, J. A. (2006). Non-negative matrix factorization for note onset detection of audio signals. In *Proc. IEEE Int. Workshop on Machine Learning for Signal Process* (pp. 447-452). Maynooth, Ireland.

Wang, W., Luo, Y., Sanei, S., & Chambers, J. A. (2008). Note onset detection via non-negative factorization of magnitude spectrum. In *EURASIP Journal on Advances in Signal Processing* (pp. 447-452).

Wang, W., & Zou, X. (2008). Non-negative matrix factorization based on projected conjugate gradient algorithm. In *Proc. ICA Research Network International Workshop* (pp. 5-8). Liverpool, UK.

Zdenuk, R., & Cichocki, A. (2007). Nonnegative matrix factorization with quadratic programming. *Neurocomputing*, *71*, 2309–2320. doi:10.1016/j.neucom.2007.01.013

Zou, X., Wang, W., & Kittler, J. (2008). Non-negative matrix factorization for face illumination analysis. In *Proc. ICA Research Network International Workshop* (pp. 52-55), Liverpool, UK.

APPENDIX A

Derivation of Equation (22)

$$\begin{aligned}\frac{\partial \hat{\mathbf{X}}_{i,j}}{\partial \mathbf{W}_{m,n}(p)} &= \frac{\partial \sum_{p}\sum_{d} \mathbf{W}_{i,d}(p)\overset{p\rightarrow}{\mathbf{H}}_{d,j}}{\partial \mathbf{W}_{m,n}(p)} \\ &= 0 + \cdots + \frac{\partial \sum_{d} \mathbf{W}_{i,d}(p)\overset{p\rightarrow}{\mathbf{H}}_{d,j}}{\partial \mathbf{W}_{m,n}(p)} + \cdots + 0 \\ &= \delta_{i,m}\overset{p\rightarrow}{\mathbf{H}}_{n,j}\end{aligned} \tag{36}$$

Derivation of Equation (24)

$$\begin{aligned}\frac{\partial \Im}{\partial \mathbf{W}_{m,n}(p)} &= \frac{\partial \sum_{i}\sum_{j}(\mathbf{X}_{i,j} - \hat{\mathbf{X}}_{i,j})^2}{\partial \mathbf{W}_{m,n}(p)} \\ &= \sum_{i}\sum_{j}(\hat{\mathbf{X}}_{i,j} - \mathbf{X}_{i,j})\frac{\partial \hat{\mathbf{X}}_{i,j}}{\partial \mathbf{W}_{m,n}(p)} \\ &= \sum_{i}\sum_{j}(\hat{\mathbf{X}}_{i,j} - \mathbf{X}_{i,j})\delta_{i,m}\overset{p\rightarrow}{\mathbf{H}}_{n,j} \\ &= \sum_{j}(\hat{\mathbf{X}}_{m,j} - \mathbf{X}_{m,j})\overset{p\rightarrow}{\mathbf{H}}_{n,j}\end{aligned} \tag{37}$$

Derivation of Equation (25)

$$\begin{aligned}\Pi_{m,n,m',n'}(p) &= \frac{\partial^2 \Im}{\partial \mathbf{W}_{m,n}(p)\partial \mathbf{W}_{m',n'}(p)} \\ &= \frac{\partial \sum_{j}(\hat{\mathbf{X}}_{m,j} - \mathbf{X}_{m,j})\overset{p\rightarrow}{\mathbf{H}}_{n,j}}{\partial \mathbf{W}_{m',n'}(p)} \\ &= \frac{\partial \sum_{j}\hat{\mathbf{X}}_{m,j}\overset{p\rightarrow}{\mathbf{H}}_{n,j}}{\partial \mathbf{W}_{m',n'}(p)} \\ &= \sum_{j}\overset{p\rightarrow}{\mathbf{H}}_{n,j}\frac{\partial \hat{\mathbf{X}}_{m,j}}{\partial \mathbf{W}_{m',n'}(p)} \\ &= \sum_{j}\delta_{m,m'}\overset{p\rightarrow}{\mathbf{H}}_{n,j}\overset{p\rightarrow}{\mathbf{H}}_{n',j}\end{aligned} \tag{38}$$

Derivation of Equation (31)

$$\begin{aligned}\left(\Pi\mathbf{w}\right)_{m,n} &= \sum_{m'}\sum_{n'}\sum_{p}\left(\sum_{j}\delta_{m,m'}\overset{p\rightarrow}{\mathbf{H}}_{n,j}\overset{p\rightarrow}{\mathbf{H}}_{n',j}\right)\mathbf{W}_{m',n'}(p)\\ &= \sum_{j}\overset{p\rightarrow}{\mathbf{H}}_{n,j}\sum_{m'}\sum_{n'}\sum_{p}\delta_{m,m'}\mathbf{W}_{m',n'}(p)\overset{p\rightarrow}{\mathbf{H}}_{n',j}\\ &= \sum_{j}\overset{p\rightarrow}{\mathbf{H}}_{n,j}\hat{\mathbf{X}}_{m,j}\\ &= \left(\hat{\mathbf{X}}(\overset{p\rightarrow}{\mathbf{H}})^{T}\right)_{m,n}\end{aligned}\tag{39}$$