BLIND RHYTHMIC SOURCE SEPARATION: NONNEGATIVITY AND REPEATABILITY

Minje Kim[1], *Jiho Yoo*[2], *Kyeongok Kang*[3] *and Seungjin Choi*[4]

Electronics and Telecommunications Research Institute (ETRI), Korea[1,3]
Department of Computer Science, POSTECH, Korea[2,4]
{mkim[1], kokang[3]}@etri.re.kr, {zentasis[2], seungjin[4]}@postech.ac.kr

ABSTRACT

An unsupervised method is proposed aiming at extracting rhythmic sources from commercial polyphonic music whose number of channels is limited to one. Commercial music signals are not usually provided with more than two channels while they often contain multiple instruments including singing voice. Therefore, instead of using conventional ways, such as modeling mixing environments or statistical characteristics, we should introduce other source-specific characteristics for separating or extracting the sources. In this paper, we concentrate on extracting rhythmic sources from the mixture with the other harmonic sources. An extension of nonnegative matrix factorization (NMF) is used to analyze multiple relationships between spectral and temporal properties in the given input matrices. Moreover, temporal repeatability of the rhythmic sound sources is implicated as common rhythmic property among segments of an input mixture signal. The proposed method shows acceptable, but not superior separation quality to the referred drum source separation systems. However, it has better applicability due to its blind manner in separation.

Index Terms— Nonnegative matrix factorization, rhythmic source separation, musical information research

1. INTRODUCTION

Separating sources from a monaural mixture or binaural ones has been concerned seriously in the musical information research area because it can help enhance the performance of various musical information research (MIR) including automatic music transcription, musical similarity analysis, query by humming, and music classification. Furthermore, object-based audio services, such as high-quality Karaoke service, can broaden their market if we can automatically get instrumental source signals from a mixed music. Musical source separation (MSS) tasks can be regarded as underdetermined blind source separation (BSS) tasks, where less number of mixtures are available than that of sources. Usually, MSS needs different kinds of assumptions about sources from conventional BSS, which models mixing environments or statistical characteristics of the sources.

NMF, which factorizes an input nonnegative matrix into two nonnegative matrices [6, 7] has shown good performance in a variety of research areas. NMF uses parts-based representation of the input data, which is also known as an important symptom when the human brain processes information. The parts-based, or sparse, representation of NMF has led a meaningful progress in the musical information research. It was used as a method for automatic transcription of percussive or polyphonic music [9], a feature extraction tool for audio classification [1], and so on. We are interested in the usage of NMF in the MSS area. NMF has shown promising separation results in simple MSS tasks [2, 3, 5, 10], and the attempt was also made for separating drum sources from commercial music using NMF and support vector machine [4] or using an extension of NMF without classification process [12].

Although NMF has been used as an effective tool for separating musical sources from their monaural mixture, the parts-based representation does not always guarantee satisfying separation for various kinds of instrumental sources. For example, NMF needs to know the number of notes to rebuild the play of a multi-pitched instrument. Furthermore, NMF seeks a linear decomposition which cannot easily follow continuously changing pitches of certain instruments, for example, singing voice, bending or sliding of strings, and so on. In the case of rhythmic instruments, however, we can assume that they do not change their spectral characteristics during their repeated playing in the temporal domain, so the linear decomposition is eligible for analyzing them.

In this paper, we follow the general concept of drum source separation, which allocates some components of NMF results to the rhythmic sources, and the others to the harmonic sources. However, we assumed the more difficult situation where no prior knowledge about the sources, drum-solo playing for the training process in this case, is available. To tackle this problem, we segmented the input mixture signal into shorter excerpts, and then factorized them into the common part and the individual parts which represent rhythmic and harmonic sources, respectively. We used an extension of NMF which was proposed in [8] as the name of fixed-effects analysis NMF (FFX-NMF).

The rest of this paper is organized as follows. Section §2 introduces the standard NMF technique as a factorization tool for spectrogram, which is a short-time Fourier transformed time-frequency representation of a given music signal. A detailed review of FFX-NMF is given along with its relationship to nonnegative partial co-factorization (NMPCF) [12] in Section §3, followed by our definition and corresponding model of blind rhythmic source separation in Section §4. Section §5 provides some comparative experimental results using 10 real-world commercial music signals, and Section §6 concludes the work.

2. NONNEGATIVE MATRIX FACTORIZATION FOR SPECTROGRAM FACTORIZATION

For a given nonnegative matrix $\boldsymbol{X}$, NMF seeks to find two nonnegative factor matrices $\boldsymbol{A}$ and $\boldsymbol{S}$ by minimizing the Euclidean error or I-divergence between the original input matrix and the reconstructed one. For example, the Euclidean error function is given by,

$$\underset{\boldsymbol{A},\boldsymbol{S}\geq 0}{\arg\min}\, \mathcal{J} = \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{S}\|_F^2, \tag{1}$$

where $\|\cdot\|_F^2$ represents the Frobenius norm (square root of the sum of the absolute squares of matrix elements). In [7], multiplicative update rules to learn the factor matrices were driven by carefully choosing the step size of the gradient descent method, which minimizes the objective function (1). Since the factor matrices are updated only by multiplicative operations, they can remain in nonnegative during the iterative updates if they have been initiated with nonnegative (random) numbers. The multiplicative update rules are given by

$$\boldsymbol{A} \leftarrow \boldsymbol{A} \odot \frac{\boldsymbol{X}\boldsymbol{S}^\top}{\boldsymbol{A}\boldsymbol{S}\boldsymbol{S}^\top}, \qquad \boldsymbol{S} \leftarrow \boldsymbol{S} \odot \frac{\boldsymbol{A}^\top \boldsymbol{X}}{\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{S}}, \tag{2}$$

where $\odot$ is an element-wise product (Hadamard product) and divisions are carried out in the element-wise way as well.

The multiplicative update rules can be derived from another way which builds the multiplication factor by taking positive terms of the partial derivative as its numerator, while taking negative ones as its denominator. For instance, the partial derivative of the objective function $\mathcal{J}$ in (1) with respect to the basis factor matrix $\boldsymbol{A}$ produces following positive and negative terms:

$$\begin{aligned} \frac{\partial \mathcal{J}}{\partial \boldsymbol{A}} &= \left[\frac{\partial \mathcal{J}}{\partial \boldsymbol{A}}\right]^+ \quad - \quad \left[\frac{\partial \mathcal{J}}{\partial \boldsymbol{A}}\right]^- \\ &= \boldsymbol{A}\boldsymbol{S}\boldsymbol{S}^\top \quad - \quad \boldsymbol{X}\boldsymbol{S}^\top. \end{aligned}$$

We can generalize it to construct the multiplicative update rule for an arbitrary factor matrix $\boldsymbol{\Theta}$ like this:

$$\boldsymbol{\Theta} \leftarrow \boldsymbol{\Theta} \odot \left(\frac{[\partial \mathcal{J} / \partial \boldsymbol{\Theta}]^-}{[\partial \mathcal{J} / \partial \boldsymbol{\Theta}]^+} \right)^{.\eta}, \tag{3}$$

where η denotes an element-wise power operation to control learning the rate by limiting it to $0 < \eta \leq 1$. The following update rules which we propose are derived in this way.

In this paper, we commonly used short-time Fourier transformed spectrogram for building the input matrices of NMPCF. We used a Hamming window of length 2048 for the input signal with 44.1kHz sampling rate. The 2048 samples were transformed into frequency domain signals, while $7/8$ of them were overlapped with the next frame to be transformed. With a pre-defined number of components, the number of column vectors of $\boldsymbol{A}$, NMF decomposes the input magnitude spectrogram into $\boldsymbol{A}$ and $\boldsymbol{S}$ where spectral basis vectors and their temporal encodings are contained, respectively. Moreover, with proper guessing of the number of components, NMF is known to separately learn those basis vectors so that some of them carry the drum-like property, while the others contain the harmonic property. To restore the rhythmic sources, one should classify the permuted basis vectors (and corresponding encodings) into the target source group and the others. Aside from the argument that the sparse representation of NMF really separates drums well, reordering those permuted basis vectors can be another burden since it needs to know about the property of sources priorly.

3. FFX-NMF AS A SPECIAL CASE OF NONNEGATIVE MATRIX PARTIAL CO-FACTORIZATION

FFX-NMF, which was proposed for classifying electroencephalogram (EEG) [8], assumes that there are common basis vectors across multiple subjects, while allowing individual basis vectors per each subject to represent subject-specific components. For l-th input matrix, FFX-NMF decomposes it with the form,

$$\boldsymbol{X}^{(l)} = \boldsymbol{A}_C \boldsymbol{S}_C^{(l)} + \boldsymbol{A}_I^{(l)} \boldsymbol{S}_I^{(l)}, \tag{4}$$

where $\boldsymbol{A}_C$ consists of the common bases which is shared by all the input matrices. On the other hands, $\boldsymbol{A}_I^{(l)}$ contains basis vectors representing the components which l-th input matrix carries individually. $\boldsymbol{S}_C^{(l)}$ and $\boldsymbol{S}_I^{(l)}$ are corresponding encoding matrices.

At the same time, NMPCF was developed to enhance the nonnegative matrix co-factorization (NMCF) [11] concept by allowing partial-sharing of basis vectors with the other input matrices in order to separate drum sources [12]. The partial-sharing concept of NMPCF can set aside some of its resulting basis vectors not to be shared with other input matrices in a similar way that FFX-NMF allows individual basis vectors. This concept is also a main improvement of NMPCF on NMCF. Therefore, FFX-NMF can be considered as a special case of NMPCF, which assumes that every input matrix contains both the common and individual basis vectors while general NMPCF does not. Likewise, NMPCF covers (4) including the case when $\boldsymbol{A}_I^{(l)}$ is an empty matrix to handle the special kind of input matrices which can be reconstructed solely by common bases. Fig. 1 (a) illustrates this case when there exists an input matrix which is compounded of common basis vectors only. A detailed explanation of fig. 1 will be found in Section §4.

This unified concept of NMPCF for L given input matrices can be expressed with the objective function given by,

$$\begin{aligned} \mathcal{J}_{\text{NMPCF}} \quad &= \quad \sum_{l=1}^{L} \lambda_l \left\| \boldsymbol{X}^{(l)} - \boldsymbol{A}_C \boldsymbol{S}_C^{(l)} - \boldsymbol{A}_I^{(l)} \boldsymbol{S}_I^{(l)} \right\|_F^2 \\ &+ \quad \gamma \left\{ \sum_{l=1}^{L} \|\boldsymbol{A}^{(l)}\|_F^2 \right\}, \end{aligned} \tag{5}$$

where the regularization term is defined by $\sum_{l=1}^L \|\boldsymbol{A}^{(l)}\|_F^2 = L\|\boldsymbol{A}_C\|_F^2 + \sum_{l=1}^L \|\boldsymbol{A}_I^{(l)}\|_F^2$. Multiplicative update rules to learn $\boldsymbol{S}^{(l)}$, $\boldsymbol{A}_C^{(l)}$, and $\boldsymbol{A}_I^{(l)}$,

$$\begin{aligned} \boldsymbol{S}^{(l)} \quad &\leftarrow \quad \boldsymbol{S}^{(l)} \odot \left(\frac{\boldsymbol{A}^{(l)\top} \boldsymbol{X}^{(l)}}{\boldsymbol{A}^{(l)\top} \boldsymbol{A}^{(l)} \boldsymbol{S}^{(l)}} \right)^{.\eta}, \\ \boldsymbol{A}_C \quad &\leftarrow \quad \boldsymbol{A}_C \odot \left(\frac{\sum_l \lambda_l \boldsymbol{X}^{(l)} \boldsymbol{S}_C^{(l)\top}}{\sum_l \lambda_l \boldsymbol{A}^{(l)} \boldsymbol{S}^{(l)} \boldsymbol{S}_C^{(l)\top} + \gamma L \boldsymbol{A}_C} \right)^{.\eta}, \\ \boldsymbol{A}_I^{(l)} \quad &\leftarrow \quad \boldsymbol{A}_I^{(l)} \odot \left(\frac{\lambda_l \boldsymbol{X}^{(l)} \boldsymbol{S}_I^{(l)\top}}{\lambda_l \boldsymbol{A}^{(l)} \boldsymbol{S}^{(l)} \boldsymbol{S}_I^{(l)\top} + \gamma \boldsymbol{A}_I^{(l)}} \right)^{.\eta}, \end{aligned} \tag{6}$$

can be derived using (3) after partially differentiating (5) with respect to each factor matrix. The weighting factor λ_ls control our concentrations across the input matrices.

4. BLIND RHYTHMIC SOURCE SEPARATION AND NMPCF

Our key assumption about blind rhythmic source separation is that some basis vectors can be shared during the inter-segment analysis due to the temporal repeatability of rhythmic sources. If we divide the input mixture spectrogram $\boldsymbol{X}$ into the smaller segments $\boldsymbol{X}^{(1)}, \boldsymbol{X}^{(2)}, \cdots, \boldsymbol{X}^{(L)}$, we can get multiple excerpts from the given mixture song. Regarding those segments as the input matrices of NMPCF, and by the above assumption about rhythmic sources, we can expect that $\boldsymbol{A}_C$ will contain basis vectors from the rhythmic sources only. Besides, the individual basis vectors $\boldsymbol{A}_I^{(l)}$ will describe the harmonic sources in each segment which cannot be shared

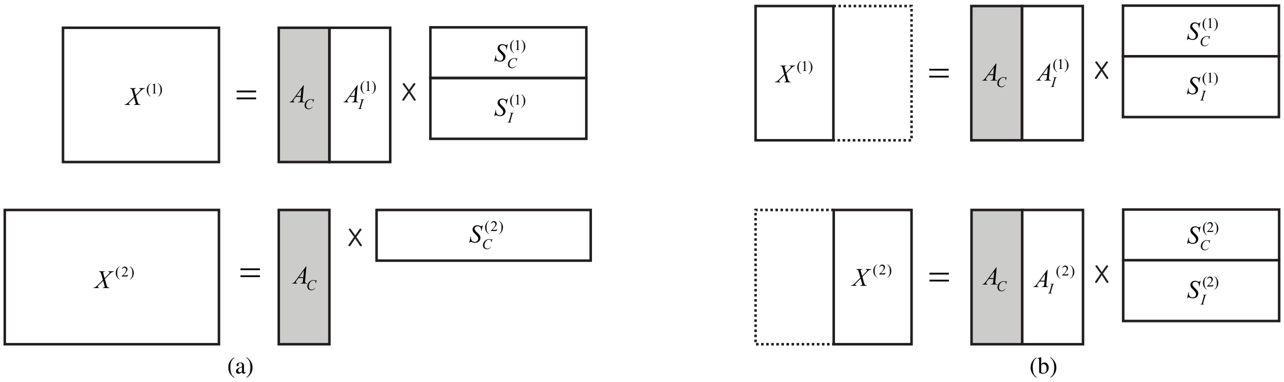

Fig. 1. (a) Drum source separation model using NMPCF. The drum-only training signal $\boldsymbol{X}^{(2)}$ is used to learn drum parts from the mixed signal $\boldsymbol{X}^{(1)}$ as prior knowledge. (b) Rhythmic source separation model using NMPCF. There is no training signal, but the input matrices $\boldsymbol{X}^{(1)}$ and $\boldsymbol{X}^{(2)}$ are the segmented mixture signal. Our proposed methods seek to find the common part $\boldsymbol{A}_C$ which reconstructs rhythmic sources repeatedly playing across all segments.

among the segments because of its changing frequency characteristics and relatively lower repeatability in their temporal appearances. Fig. 1 (b) gives us a pictorial example assuming that the input mixture spectrogram is divided into two segments. On the contrary to the case in fig. 1 (a), where the input matrix $\boldsymbol{X}^{(2)}$ is a collection of drum-only signals, fig. 1 (b) represents a blind situation that there is no pure training data priorly known. Our proposed system relies on the assumption about rhythmic sources with no help of prior knowledge about the sources, so we can regard it as a blind method.

Meanwhile, we prefer to call the proposed method as a *rhythmic* source separation method than *drum*. The reason why we hesitate to limit our method just for separating drums is that sometimes bass guitar sounds are extracted with our method as well. Therefore, we define the word, rhythmic, to cover drums, bass guitars, and all the repeatedly playing instruments. Although bass guitars, for example, can play multiple notes, they are also rhythmic. Because they are apt to be coincidently played with kicks of bass drum in popular music. With deeper consideration about the definition of rhythmic sound, we can find that some instruments are not easy to be clearly classified whether they are rhythmic or harmonic. There are many criteria which can help classify instruments, for example, noise-likeness, periodicity, spectral roll-off point, and so on. So the decision of instrument classification depends on what kind of criteria we are using. One of our key criteria is that rhythmic sources are repeating, and it groups bass guitars in the same class with drums. We believe that this definition about rhythmic instruments is plausible to the people widely.

5. NUMERICAL EXPERIMENTS

For the numerical experiments, we used 100 seconds excerpts of 10 real-world commercial music signals with 44.1kHz sampling rate and 16 bit encoding. Furthermore, to examine the separation quality more quantitatively, we also secured the corresponding source signals. All of them were publicly released famous Korean pop numbers which were collected from various artists. We examined our method using them instead of well-known database for BSS in order to check the usability in the market. We examined our method by dividing those mixture music signals into the successive pre-defined length of segments. There are several parameters to be set like these:

- Parameters for time-frequency transform: $7/8$ of 2048 windowed samples were overlapped with the next one.
- η, γ, λ_l: set to 1 since they do not influence the separation results much.
- Number of basis vectors: 30 for $\boldsymbol{A}_C$ and 15 for all $\boldsymbol{A}_I^{(l)}$. A little more or less numbers does not affect much, but we allocated enough number of basis vectors for commonly sharing in order to cover whole segments.
- Segment length: it does lead fluctuations of separation results (SNR) around 1 to 2 dB, but we have no way to judge its goodness in advance. We heuristically picked up 4 seconds for all segments, but the fluctuation is acceptable if the length of the segments is long enough to contain a lot of rhythmic sources.
- Optimal iteration number: the objective function does converge within several hundreds of iterations, but a lower value of the objective function does not always guarantee better separation results. It is because the SNR values of rhythmic sources make a peak roughly between 5 to 20 and then decrease. The optimal iteration number with respect to SNR also varies with the input mixture signals, so we decided to stop updating at the 15-th iteration even though this is not the best for every input.

Therefore, the L segments of the transformed input mixture are fed to the NMPCF optimization process (6). After the decomposition, for a given segment $\boldsymbol{X}^{(l)}$, we reconstructed the magnitude spectrum of the rhythmic source $\boldsymbol{Y}_R^{(l)}$ by a simple matrix inner-production between the common basis vectors $\boldsymbol{A}_C$ and their segment-specific encodings $\boldsymbol{A}_C^{(l)}$. The reconstructed rhythmic source segments $\widetilde{\boldsymbol{Y}_R^{(l)}}$ are concatenated to make the full-length rhythmic parts of the song, $\widetilde{\boldsymbol{Y}_R}$. The harmonic sources $\boldsymbol{Y}_H^{(l)}$ can be restored in a similar way, except the individual basis vectors $\boldsymbol{A}_I^{(l)}$ and their corresponding encodings $\boldsymbol{S}_I^{(l)}$ are used instead. For a practical purpose, one can get less noisy harmonic sources by subtracting the restoration of rhythmic sources from the mixture rather than multiplying the analyzed individual factor matrices.

We measured signal-to-noise ratio (SNR) given by,

$$\text{SNR} = 10\log_{10}\frac{\sum s(t)^2}{(\sum s(t) - \tilde{s}(t))^2},$$

Table 1. SNR of separation measured for the 10 popular music songs using NMF+SVM, S-NMPCF, and U-NMPCF.

Song	SNR (Drums)			SNR (Harmonic)		
	NMF +SVM	S-NMPCF	U-NMPCF	NMF +SVM	S-NMPCF	U-NMPCF
1	8.02	**8.84**	6.21	**8.43**	7.95	4.71
2	4.58	**5.48**	4.91	3.49	**4.66**	3.56
3	4.29	**5.04**	4.23	4.69	**5.98**	4.28
4	**3.62**	3.01	1.64	**5.14**	4.21	2.34
5	**5.56**	5.20	1.32	6.17	**6.47**	4.52
6	4.82	**6.90**	0.27	1.35	**5.40**	4.26
7	3.87	**3.94**	3.92	**7.08**	6.68	4.49
8	-0.68	**2.76**	1.80	3.91	**6.36**	3.20
9	4.19	**4.32**	4.16	**7.30**	7.04	4.41
10	**7.90**	7.81	5.08	**8.41**	8.08	5.29
mean	4.62	**5.33**	3.35	5.60	**6.28**	4.11

where $s(t)$ and $\tilde{s}(t)$ are the original source and the reconstructed one, respectively. Table 1 shows comparative separation results among three systems including NMF+SVM [4], supervised NMPCF (S-NMPCF) [12], and the proposed unsupervised NMPCF (U-NMPCF). We set the parameters of those systems as same as possible with the ones when they were originally proposed. We commonly used the mixture signals above-mentioned, but, for NMF+SVM and S-NMPCF, 130 seconds of additional drum-only signals were used for training.

We concede that the supervised methods, NMF+SVM and S-NMPCF, usually work better than the proposed unsupervised method, U-NMPCF. However, considering that the proposed method did not use any prior information about the rhythmic sources, the quality of U-NMPCF is quite promising. Furthermore, in the case of some low performance results, such as song 6, there is actually another optimal iteration number where the separation quality is higher. However, with the assumption of blindness, we set the iteration number as 15 for all test songs. Another important reason of the degradation is that the reconstructed rhythmic sources contain not only drum sources, but also bass guitar sounds, which were compared with drum-only original source signals. We suggest to the potential users that they listen carefully to the separated sounds at every iteration and manually decide where to stop to get the best result.

6. CONCLUSION

This paper presented a novel method of separating rhythmic sources from monaural music mixtures. We used a branch of NMF, named NMPCF, to learn temporally repeating sources across the successive segments of the input mixture. According to our definition of rhythmic sources, the proposed method yielded the meaningful separation results recovering most of drum sounds and rhythmically playing bass guitar sounds as well. Our method has some aspects to improve including the decision of optimal number of iterations and length of segments. However, the authors believe that this blind method can help deal with anomalous rhythmic sources, for example, electronic sounds, which cannot be separated well with the supervised methods using ordinary drum database.

Acknowledgments: This research is supported by Ministry of Culture, Sports and Tourism(MCST) and Korea Creative Content Agency(KOCCA) in the Culture Technology(CT) Research & Development Program 2009.

7. REFERENCES

[1] Y. C. Cho and S. Choi, "Nonnegative features of spectro-temporal sounds for classfication," *Pattern Recognition Letters*, vol. 26, no. 9, pp. 1327–1336, 2005.

[2] D. FitzGerald, M. Cranitch, and E. Coyle, "Shifted nonnegative matrix factorisation for sound source separation," in *IEEE Workshop on Statistical Signal Processing*, Bordeaux, France, 2005.

[3] ——, "Sound source separation using shifted non-negative tensor factorisation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toulouse, France, 2006.

[4] M. Helen and T. Virtanen, "Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine," in *European Signal Processing Conference*, 2005.

[5] M. Kim and S. Choi, "On spectral basis selection for single channel polyphonic music separation," in *Proceedings of the International Conference on Artificial Neural Networks (ICANN)*, vol. 2. Warsaw, Poland: Springer, 2005, pp. 157–162.

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

[7] ——, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 13. MIT Press, 2001.

[8] H. Lee and S. Choi, "Group nonnegative matrix factorization for EEG classification," in *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, Clearwater Beach, Florida, 2009.

[9] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, 2003, pp. 177–180.

[10] T. O. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[11] J. Yoo and S. Choi, "Weighted nonnegative matrix co-tri-factorization for collaborative prediction," in *Proceedings of 1st Asian Conference on Machine Learning*, Nanjing, China, 2009.

[12] J. Yoo, M. Kim, K. Kang, and S. Choi, "Nonnegative matrix partial co-factorization for drum source separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Austin, Texas, USA, 2010.