# NONNEGATIVE MATRIX PARTIAL CO-FACTORIZATION FOR DRUM SOURCE SEPARATION

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

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

## ABSTRACT

We address a problem of separating drums from polyphonic music containing various pitched instruments as well as drums. Nonnegative matrix factorization (NMF) was successfully applied to spectrograms of music to learn basis vectors, followed by support vector machine (SVM) to classify basis vectors into ones associated with drums (rhythmic source) only and pitched instruments (harmonic sources). Basis vectors associated with pitched instruments are used to reconstruct drum-eliminated music. However, it is cumbersome to construct a training set for pitched instruments since various instruments are involved. In this paper, we propose a method which only incorporates prior knowledge on drums, not requiring such training sets of pitched instruments. To this end, we present nonnegative matrix partial co-factorization (NMPCF) where the target matrix (spectrograms of music) and drum-only-matrix (collected from various drums *a priori*) are simultaneously decomposed, sharing some factor matrix partially, to force some portion of basis vectors to be associated with drums only. We develop a simple multiplicative algorithm for NMPCF and show its usefulness empirically, with numerical experiments on real-world music signals.

***Index Terms***— Drum source separation, matrix co-factorization, music information processing, nonnegative matrix factorization

## 1. INTRODUCTION

Nonnegative matrix factorization (NMF) is a low-rank approximation method where a nonnegative input data matrix (target matrix) is approximated as a product of two nonnegative factor matrices [7]. NMF has been used in various applications, including image processing, brain computer interface, document clustering, collaborative predictions, and so on. Recent advances in matrix factorization methods suggest *collective matrix factorization* or *matrix co-factorization* to incorporate side information, where several matrices (target and side information matrices) are simultaneously decomposed, sharing some factor matrices. Matrix co-factorization methods have been developed to incorporate label information [15], link information [16], and inter-subject variations [8]. Collective matrix factorization [9] was studied to analyze multiple relational data matrices. Co-factorization methods were extended to a 3-factor decomposition. For instance, nonnegative matrix co-tri-factorization [14] was proposed and successfully applied to solve a cold start problem in collaborative prediction.

One of promising applications of NMF, which is addressed here, is musical signal analysis such as music source separation [6, 13] and music transcriptions [10, 1, 12]. We consider a problem of *drum source separation*, the goal of which is to extract a rhythmic source from polyphonic music consisting of multiple instruments as well as vocals [17, 11, 5, 4]. One of main approaches to drum source separation is to decompose the spectrograms (time-frequency representation) of music signals into a product of two factor matrices (one for basis matrix and the other for encoding matrix) using NMF, in order to identify components related to drum only, which are used to reconstruct drum-eliminated music signals. In such a case, some prior knowledge on drum signals is required to identify which basis vectors in NMF contribute to drum or other pitched instruments. Basis vectors learned using drum-only signals were used for initialization in decomposing the target matrix by NMF [3]. Some heuristics were used to distinguish basis vectors involving drum from the ones related to pitched instruments [2]. Support vector machine (SVM) was used to classify basis vectors into drum-related ones and others [5]. All these methods make use of prior knowledge on drum implicitly in decomposing the target matrix using NMF.

Matrix co-factorizations can be served as a useful tool when side information matrices (constructed from solo playing of musical source of interest) are available, in addition to the target matrix to be factorized. In this paper we present *nonnegative matrix partial co-factorization* (NMPCF) where we jointly decompose the target matrix and a drum-only matrix[1], forcing basis vectors involving the drum-only matrix to be shared with the target matrix factorization (see Fig 1). NMPCF makes use of drum-prior knowledge directly in decomposing the target matrix, in contrast to most of existing methods where prior knowledge was implicitly used. Co-factorizing the target matrix and the drum-only matrix leads shared basis vectors to be associated with drum characteristics, so that learned factor matrix is comprised of drum and and pitched instruments without any post-processing by pre-trained classifiers. Shared basis vectors correspond to drum and non-shared basis vectors are associated with pitched instruments. We develop simple multiplicative updates for NMPCF. Numerical experiments on real-world music confirm the validity and high performance of NMPCF, compared to a state of the arts (NMF+SVM).

## 2. NONNEGATIVE MATRIX FACTORIZATION FOR DRUM SOURCE SEPARATION

NMF can be used to analyze the spectral and temporal characteristics of sounds contained in a given spectrogram. If we denote the magnitude spectrogram matrix as $\boldsymbol{X}$, then each element $\boldsymbol{X}_{ft}$ represents the magnitude of the spectrum of the $f$-th frequency bin at the

[1]The drum-only matrix is constructed by collecting various drum sounds which are not used in the polyphonic music associated with the target matrix.

$t$-th time frame, which can be calculated by

$$\boldsymbol{X}_{ft} = \left| \sum_{n=0}^{N-1} s_t(n) \exp^{-j\frac{2\pi}{N}fn} \right|,$$

where $s_t$ is the $t$-th windowed time domain signal. By applying NMF on this nonnegative magnitude spectrogram matrix $\boldsymbol{X}$ as

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{V}^\top,$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are also constrained to be nonnegative, the resulting matrix $\boldsymbol{U}$ represents the frequency bases of the sources contained in the signal, and the corresponding columns of $\boldsymbol{V}$ represent the activations of the frequency bases across the time. As a result, some of the bases in $\boldsymbol{U}$ represent the drum sources, and the others represent the remaining harmonic sources. If we can collect the frequency basis vectors $\boldsymbol{U}_D$ representing the drum sources and corresponding activation patterns $\boldsymbol{V}_D$, we can reconstruct the magnitude spectrogram of the separated drum source $\boldsymbol{X}_D$ as,

$$\boldsymbol{X}_D = \boldsymbol{U}_D\boldsymbol{V}_D^\top.$$

However, the components representing drums are placed in arbitrary locations in $\boldsymbol{U}$, so we have to distinguish which components are representing drum sources. Prior knowledge about the drum sources, such as non-harmonic frequency spectrums and rapid decaying time domain activations, can be used. Several methods have been introduced to identify the drum components from the extracted components. [2] used a set of heuristics to identify drum components. [3] used the known frequency characteristics of the drum sources to initialize a part of factor matrix, to explicitly set the position of the drum components $\boldsymbol{U}_D$ in the factor matrix $\boldsymbol{U}$. [5] used an SVM classifier, which is trained by using a variety of spectral and temporal features of the extracted components from two drums and harmonic sources, to distinguish frequency bases of drums.

In the existing methods, the prior knowledge is only implicitly used in the separation process. If we use the prior knowledge in the initializations [3], there is no guarantee that the initialized part remains as the drum part after the decomposition process using NMF. In the case of using heuristics [2], the prior knowledge does not involve in the decomposition process, so we cannot sure that NMF actually separate the drum part and the harmonic part. NMF+SVM [5] also does not use the knowledge in the decompositions, and moreover, it requires additional burden to build the classifier, with expensive preparation of training data consists of various kinds of harmonic sources, as well as the prior drum sources. Also it is possible that the performance of separation is degraded by both the classification error of SVM and separation error of NMF. As a remedy for these problems, we propose a co-factorization based method which uses the prior knowledge explicitly in the decomposition process, and requires no additional heuristics or classifiers to distinguish drum bases.

## 3. NONNEGATIVE MATRIX PARTIAL CO-FACTORIZATION

Nonnegative Matrix Partial Co-Factorization (NMPCF) is developed for the separation of drum sounds from input music signal. NMPCF uses a decomposition model which explicitly distinguish the drum part and the harmonic part from a mixture as follows,

$$\boldsymbol{X} = \boldsymbol{U}_D\boldsymbol{V}_D^\top + \boldsymbol{U}_H\boldsymbol{V}_H^\top, \quad (1)$$

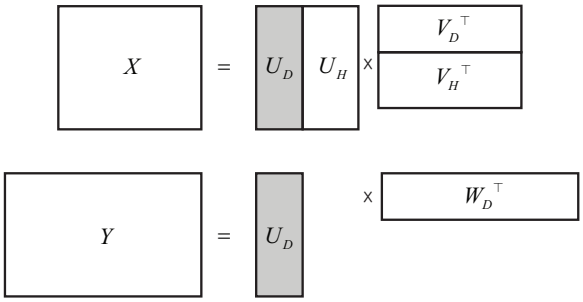


**Fig. 1**. A pictorial illustration of NMPCF model. A part of factor matrix $\boldsymbol{U}_D$ in the factorization of the target signal spectrogram is shared in the factorizations of the prior signal spectrograms.

where $\boldsymbol{U}_D$ and $\boldsymbol{V}_D$ represent the frequency and time characteristics of the drum sources, respectively, and $\boldsymbol{U}_H$ and $\boldsymbol{V}_H$ represent the frequency and time characteristics of the harmonic sources. All the matrices $\boldsymbol{U}_D$, $\boldsymbol{V}_D$, $\boldsymbol{U}_H$ and $\boldsymbol{V}_H$ are constrained to be nonnegative. Although we explicitly designate the drum bases $\boldsymbol{U}_D$ and harmonic bases $\boldsymbol{U}_H$ in this model, we do not have a way to automatically discriminate them. Therefore we exploit an additional prior knowledge which consists of the spectrogram of the drum-only signal $\boldsymbol{Y}$. Since the additional knowledge contains only the sound of drums, it can be decomposed into

$$\boldsymbol{Y} = \boldsymbol{U}_D\boldsymbol{W}_D^\top, \quad (2)$$

where $\boldsymbol{U}_D$ can be treated as the same matrix in the model (1), and $\boldsymbol{W}_D$ represents the time domain activations of each column of $\boldsymbol{U}_D$ in $\boldsymbol{Y}$. The most straight-forward way to use this additional knowledge is to decompose $\boldsymbol{Y}$ to obtain $\boldsymbol{U}_D$, then use it in the model (1) as initial values of $\boldsymbol{U}_D$ [3]. However, this kind of initialization approach does not guarantee that the initialized part remains to represent drum part after the decomposition process.

Instead of the simple initialization method, we propose a partial co-factorization method which shares the frequency basis matrix $\boldsymbol{U}_D$ to factorize the input signal $\boldsymbol{X}$ and the prior signal $\boldsymbol{Y}$ (Fig. 1). If we jointly factorize the matrix $\boldsymbol{X}$ with $\boldsymbol{Y}$, the frequency characteristics of drums are collected in the matrix $\boldsymbol{U}_D$, while $\boldsymbol{U}_H$ is trained to represent the remaining part of the input music source signal $\boldsymbol{X}$, which is supposed to be the harmonic part of the music. The objective function of NMPCF can be constructed to minimize the residuals of the models (1) and (2), which becomes

$$\mathcal{L} = \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{U}_D\boldsymbol{V}_D^\top - \boldsymbol{U}_H\boldsymbol{V}_H^\top\|_F^2 + \frac{\lambda}{2}\|\boldsymbol{Y} - \boldsymbol{U}_D\boldsymbol{W}_D^\top\|_F^2, \quad (3)$$

where $\lambda$ is the parameter adjusting the relative importance between the input data and drum prior.

If we can decompose the gradient of above objective function $\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}}$ as follows,

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}} = \left[\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}}\right]^+ - \left[\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}}\right]^-,$$

where $\left[\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}}\right]^+ > 0$ and $\left[\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}}\right]^- > 0$, then we can build the multiplicative update rules as

$$\boldsymbol{U} \leftarrow \boldsymbol{U} \odot \frac{\left[\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}}\right]^-}{\left[\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}}\right]^+}.$$

This multiplicative update rule has a stationary point at the local minimum, and does not break the nonnegativity constraint for the matrix $\boldsymbol{U}$.

From the objective function (3), we can calculate the gradient for each matrix to be updated as follows,

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \boldsymbol{U}_D} &= -\boldsymbol{X}\boldsymbol{V}_D + \boldsymbol{U}_D\boldsymbol{V}_D^\top\boldsymbol{V}_D + \boldsymbol{U}_H\boldsymbol{V}_H^\top\boldsymbol{V}_D \\
&\quad -\lambda\boldsymbol{Y}\boldsymbol{W}_D + \lambda\boldsymbol{U}_D\boldsymbol{W}_D^\top\boldsymbol{W}_D, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{V}_D} &= -\boldsymbol{X}^\top\boldsymbol{U}_D + \boldsymbol{V}_D\boldsymbol{U}_D^\top\boldsymbol{U}_D + \boldsymbol{V}_H\boldsymbol{U}_H^\top\boldsymbol{U}_D, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{U}_H} &= -\boldsymbol{X}\boldsymbol{V}_H + \boldsymbol{U}_H\boldsymbol{V}_H^\top\boldsymbol{V}_H + \boldsymbol{U}_D\boldsymbol{V}_D^\top\boldsymbol{V}_H, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{V}_H} &= -\boldsymbol{X}^\top\boldsymbol{U}_H + \boldsymbol{V}_H\boldsymbol{U}_H^\top\boldsymbol{U}_H + \boldsymbol{V}_D\boldsymbol{U}_D^\top\boldsymbol{U}_H, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{W}_D} &= -\boldsymbol{Y}^\top\boldsymbol{U}_D + \boldsymbol{W}_D\boldsymbol{U}_D^\top\boldsymbol{U}_D,
\end{aligned}$$

and the multiplicative update rule becomes

$$\boldsymbol{U}_D \leftarrow \boldsymbol{U}_D \odot \frac{\boldsymbol{X}\boldsymbol{V}_D + \lambda\boldsymbol{Y}\boldsymbol{W}_D}{\boldsymbol{U}_D\boldsymbol{V}_D^\top\boldsymbol{V}_D + \boldsymbol{U}_H\boldsymbol{V}_H^\top\boldsymbol{V}_D + \lambda\boldsymbol{U}_D\boldsymbol{W}_D^\top\boldsymbol{W}_D}, \tag{4}$$

$$\boldsymbol{V}_D \leftarrow \boldsymbol{V}_D \odot \frac{\boldsymbol{X}^\top\boldsymbol{U}_D}{\boldsymbol{V}_D\boldsymbol{U}_D^\top\boldsymbol{U}_D + \boldsymbol{V}_H\boldsymbol{U}_H^\top\boldsymbol{U}_D}, \tag{5}$$

$$\boldsymbol{U}_H \leftarrow \boldsymbol{U}_H \odot \frac{\boldsymbol{X}\boldsymbol{V}_H}{\boldsymbol{U}_H\boldsymbol{V}_H^\top\boldsymbol{V}_H + \boldsymbol{U}_D\boldsymbol{V}_D^\top\boldsymbol{V}_H}, \tag{6}$$

$$\boldsymbol{V}_H \leftarrow \boldsymbol{V}_H \odot \frac{\boldsymbol{X}^\top\boldsymbol{U}_H}{\boldsymbol{V}_H\boldsymbol{U}_H^\top\boldsymbol{U}_H + \boldsymbol{V}_D\boldsymbol{U}_D^\top\boldsymbol{U}_H}, \tag{7}$$

$$\boldsymbol{W}_D \leftarrow \boldsymbol{W}_D \odot \frac{\boldsymbol{Y}^\top\boldsymbol{U}_D}{\boldsymbol{W}_D\boldsymbol{U}_D^\top\boldsymbol{U}_D}. \tag{8}$$

The convergence of the above multiplicative update rules can be shown by using the auxiliary function method similar to the method used in [7]. In fact, the update algorithm for $\boldsymbol{V}_D$, $\boldsymbol{U}_H$, $\boldsymbol{V}_H$ and $\boldsymbol{W}_D$ is essentially equal to the algorithm for the standard NMF algorithm, because these factor matrices are not shared in co-factorization model. For the update rule of $\boldsymbol{U}_D$, we can generate an auxiliary function as the sum of the auxiliary functions used in [7], then the new auxiliary function also satisfies the positive semi-definiteness condition and the convergence can be proved in the same way.

By iteratively updating the matrices, we can find the basis matrix for the drum sound $\boldsymbol{U}_D$ and the corresponding time activations $\boldsymbol{V}_D$ without further processing. The overall process of the drum source separation using NMPCF algorithm is summarized below.

**Algorithm outline: Drum source separation using NMPCF**

1. Prepare the spectrogram of the target music signal $\boldsymbol{X}$ and the spectrogram of the prior drum signal $\boldsymbol{Y}$ Initialize factor matrices with random positive values.
2. Iterate
   (a) Update each factor matrix using (4) .. (8)
3. Reconstruct the separated signals
   (a) Compute the inverse of the spectrogram $\boldsymbol{X}_D = \boldsymbol{U}_D\boldsymbol{V}_D^\top$ to obtain drum signal
   (b) Compute the inverse of the spectrogram $\boldsymbol{X}_H = \boldsymbol{U}_H\boldsymbol{V}_H^\top$ to obtain harmonic signal

## 4. NUMERICAL EXPERIMENTS

We tested the proposed NMPCF algorithm for drum source separation problems. We used 10 commercial popular music songs, each of which is 100 seconds-long, as targets for the separation tasks. A drum track and a harmony track compound each of these songs by instantaneous addition. We segmented the song into 10-second excerpts, and used those 10 excerpts for the experiments. On the other hand, we built the prior drum signals from another 13 popular songs which consist of only drum sounds. We picked a 10 seconds excerpt from each prior drum signal, and concatenated them to make a prior drum signal with the length of 130 seconds. All the songs have sampling rate of 44,100 Hz with 16 bit encoding, and we used the window which has the length of 2048 samples (approximately 50 ms) sliding by the length of 256 samples.

For the measure of separation quality, we used the signal-to-noise ratio (SNR) which compares the original signals with the residual of original and separated signals, like,

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

where $s(n)$ is the original signal and $\tilde{s}(n)$ is the separated signal. Using drum track and harmony track of the test songs as original signals, we measured the SNRs for the separated drum and harmonic sound signals.

The parameters of the NMPCF algorithm are the number of components used for the drums, the number of components used for the harmonic sounds, and the weight parameter $\lambda$. In addition to this, the maximum number of iterations can be specified to control the length of the learning time.

For the number of components, we tested several sets of the numbers in the separation tasks to decide the appropriate numbers. Usually, the number of components required for the drums are less than that for the harmonic signals. The number 70 for the drum sources and 100 for the harmonic sources showed good performance, so we used the numbers for all test songs.

To determine the value of parameter $\lambda$ which decides the relative importance between input signal and prior signal, we ran the separation with several different numbers of $\lambda$. The value should balance the difference of the length of target signal and drum prior signal. In our case, the ratio between the target signal and prior signal is $R = 10/130$, so the numbers around these values were tested. The optimal value depends on the target songs, but we chose $R \times 0.1 \approx 0.0077$ as $\lambda$ value which usually worked well.

To decide the number of iterations, we checked the SNR value of the separation at each iteration step. Although the value of the objective function converges relatively slowly, the SNR of separation reaches the maximum value in a few iterations. The number 20 of iterations are enough to obtain the good separation results.

We ran the NMPCF algorithm to separate the target songs with the parameter values above. As a baseline of the separation performance, we also implemented an NMF+SVM method and ran it for the same dataset. Table 1 shows the SNR result of the separated drum part and the SNR result of the separated harmonic part. The separation performance of the excerpts from the same song is usually quite similar, so we averaged SNR results over the excerpts. The performance depended on the target song, but NMPCF usually worked better than NMF+SVM. Some of the results of the NMF+SVM were very low, possibly because of the failure in the decomposition in NMF or in classification process. However, NMPCF showed better or almost comparable results to NMF+SVM.

**Table 1**. SNR of separation results measured for the 10 popular music songs using NMF+SVM and NMPCF. Each SNR value is the average over 10 segments of the corresponding song. For each segment, we ran the algorithm 5 times and take the mean of the SNR results.

| | SNR (Drums) | | SNR (Harmonic) | |
|---|---|---|---|---|
| Song | NMF+SVM | NMPCF | NMF+SVM | NMPCF |
| 1 | 8.02 | **8.84** | 8.43 | 7.95 |
| 2 | 4.58 | **5.48** | 3.49 | **4.66** |
| 3 | 4.29 | **5.04** | 4.69 | **5.98** |
| 4 | 3.62 | 3.01 | 5.14 | 4.21 |
| 5 | 5.56 | 5.20 | 6.17 | **6.47** |
| 6 | 4.82 | **6.90** | 1.35 | **5.40** |
| 7 | 3.87 | **3.94** | 7.08 | 6.68 |
| 8 | -0.68 | **2.76** | 3.91 | **6.36** |
| 9 | 4.19 | **4.32** | 7.30 | 7.04 |
| 10 | 7.90 | 7.81 | 8.41 | 8.08 |
| mean | 4.62 | **5.33** | 5.60 | **6.28** |

## 5. CONCLUSIONS

We proposed the NMPCF model which shares some part of the factor matrix with the factor matrix of the prior knowledge. Simple multiplicative update algorithm was derived for the model. The resulting shared factor matrix automatically contains the frequency characteristics of drum source signals, so we could separate the drum parts and harmonic parts without further manipulations. Numerical experiments on the real world music signals showed that the proposed algorithm worked better than the existing NMF+SVM method on average.

**Acknowledgments:** This research was supported by Ministry of Culture, Sports and Tourism (MCST) and Korea Creative Content Agency (KOCCA) in the Culture Technology Research & Development Program 2009, Korea Research Foundation Grant (KRF-2008-313-D00939), and NRF WCU Program (Project No. R31-2008-000-10100-0).

## 6. REFERENCES

[1] N. Bertin, R. Badeau, and G. Richard, "Blind signal decompositions for automatic transcription of polyphonic music: NMF and K-SVD on the benchmark," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Honolulu, Hawaii, 2007.

[2] C. Dittmar and C. Uhle, "Further steps towards drum transcription of polyphonic music," in *Proceedings of the Audio Engineering Society Convention*, Berlin, Germany, 2004.

[3] D. FitzGerald, B. Lawlor, and E. Coyle, "Prior subspace analysis for drum transcription," in *Proceedings of the Audio Engineering Society Convention*, Amsterdam, The Netherlands, 2003.

[4] O. Gillet and G. Richard, "Transcription and separation of drum signals from polyphonic music," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, pp. 529–540, 2008.

[5] M. Helen and T. Virtanen, "Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine," in *Proceedings of EUSIPCO*, 2005.

[6] M. Kim and S. Choi, "Monaural music source separation: Nonnegativity, sparseness, and shift-invariance," in *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation (ICA)*, Charleston, South Carolina, 2006.

[7] D. D. Lee and H. S. Seung, "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] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, Las Vegas, Nevada, 2008.

[10] 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.

[11] C. Uhle, C. Dittmar, and T. Sporer, "Extraction of drum tracks from polyphonic music using independent subspace analysis," in *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation (ICA)*, Nara, Japan, 2003.

[12] E. Vincent, N. Berlin, and R. Badeau, "Harmonic and inharmonic nonnegative matrix factorization for polyphonic pitch transcription," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, NV, 2008.

[13] T. O. Virtanen, "Monaural sound source separation by perceptually weighted non-negative matrix factorization," Tampere University of Technology, Tech. Rep., 2007.

[14] J. Yoo and S. Choi, "Weighted nonnegative matrix co-trifactorization for collaborative prediction," in *Proceedings of the 1st Asian Conference on Machine Learning (ACML)*, Nanjing, China, 2009.

[15] K. Yu, S. Yu, and V. Tresp, "Multi-label informed latent semantic indexing," in *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Salvador, Brazil, 2005.

[16] S. Zhu, K. Yu, Y. Chi, and Y. Gong, "Combining content and link for classification using matrix factorization," in *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Amsterdam, The Netherlands, 2007.

[17] A. Zils, F. Pachet, O. Delerue, and F. Gouyon, "Automatic extraction of drum tracks from polyphonic music signals," in *Proceedings of the 2nd International Conference on Web Delivering of Music*, Darmstadt, Germany, 2002.