Towards Shifted NMF for Improved Monaural Separation

Rajesh Jaiswal[†], **Derry Fitzgerald**[†], **Eugene Coyle**[†] **and Scott Rickard**[*]

[†]*Department of Electrical Engineering*
Audio Research Group
Dublin Institute of Technology

[*]*Department of Electronic Engineering*
University College Dublin

E-mail: `rajesh.enc@gmail.com` `derry.fitzgerald@dit.ie`

Abstract — **The ability of Non-negative Matrix Factorisation (NMF) to decompose magnitude spectrogram into meaningful entities has found use in many audio applications. NMF can be used to factorise audio spectrogram of a music signal into parts based frequency basis functions which typically corresponds to notes and chords in music. However, these pitched basis functions needed to be clustered to their respective sources. Many clustering algorithms have been proposed to group these basis functions. Recently, Shifted Non-negative Matrix Factorisation (SNMF) based methods have been used to reconstruct individual sound sources. Clustering of basis functions using SNMF uses a Constant Q Transform (CQT) of the frequency basis functions. Here, we argue that incorporating the CQT into the SNMF model can be used to better the separation quality of individual sources. An algorithm is presented to estimate sound sources and is an improvement to the existing techniques. Results are compared to show the improvement.**

Keywords — **NMF, Frequency basis functions, shifted NMF, Single channel Source Separation.**

I Introduction

[1]The process by which the individual music sources are estimated from a single channel music mixture is known as Monaural Sound Source Separation (SSS). A monaural mixture has a complex overlapping of audio signals from different musical instruments in frequency and time. Also, the absence of stereo space in the monaural mixture makes SSS a difficult problem. However, the availability of a method to separate sound sources would help in analysis, manipulation and re-localisation of audio data which would benefit many audio applications such as music transcription, pitch modification, remixing monophonic sound to 5.1 surround system.

Recently, Non-negative Factorisation (NMF) [1] based method have been used in separating sound sources with considerable success [2, 3]. NMF can approximately decompose the time-frequency representations such as the audio magnitude spectrogram X into parts based NMF basis functions such that the frequency basis functions that typically represent the spectral envelope of individual notes or chords present in the mixture. The reduced rank approximation of the factorised spectrogram gives factors $\mathbf{A}$ and $\mathbf{B}$.

$$|\mathbf{X}| = X \approx \hat{X} = \mathbf{AB} \tag{1}$$

where the matrix $\mathbf{A}$ is of size $n \times r$ and the matrix $\mathbf{B}$ is of size $r \times m$, with $r < n,\ m$. $\mathbf{X}$ represents the complex valued spectrogram of the input signal. In equation 1 the input magnitude spectrogram X is approximated by $\hat{X}$. Here, $\hat{X}$ is a linear combination of the columns of matrix $\mathbf{A}$, and the corresponding rows of matrix $\mathbf{B}$. Matrix $\mathbf{A}$ contains frequency basis functions and matrix $\mathbf{B}$ stores the corresponding amplitude basis functions which gives temporal information that

[1]This work is funded by ABBEST PhD Research Scholarship at Dublin Institute of Technology, Dublin, Ireland.

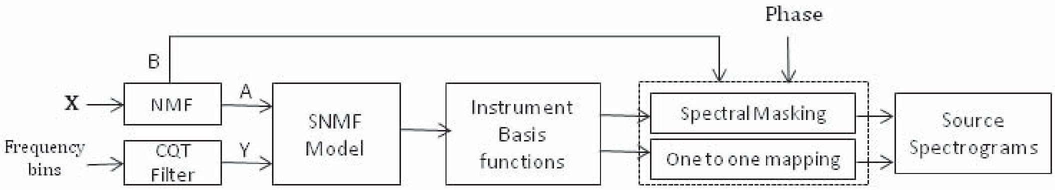

Figure 1: *Block Diagram of the System model*

describes when the frequency basis functions are active within the mixture.

The factorisation can be achieved through various techniques. However, a widely used cost function, the generalised Kullback-Leibler (kl) divergence [1] has been used for this paper. Also, all the elements in $\mathbf{A}$ and $\mathbf{B}$ are constraint to be non-negative.

$$\mathbf{D}_{kl}(X||\hat{X}) = \sum_{i,j}(X_{ij}log\frac{X_{ij}}{\hat{X}_{ij}} - X_{ij} + \hat{X}_{ij}) \qquad (2)$$

In general, there are multiple notes per instrument present in the real world audio signals. A drawback of using NMF is that the number of frequency basis functions obtained are usually greater in number than the required number of sources. Therefore, the frequency basis functions need to be clustered to their respective sources. Many clustering algorithm have been proposed. A data-adaptive method to map the separated signals into sources has been proposed in [7]. In [8], a source-filter based model was used to determine the parameters to segregate the frequency basis functions. Here, the clustering is performed in Mel frequency domain and then the respective sources are recovered from the clustered basis functions.

Recently, Shifted NMF (SNMF) was proposed in order to avoid clustering of basic functions [5]. SNMF uses the property of shift invariance in the frequency basis functions. This solves the problem of grouping of different notes played by same musical instrument. It is assumed that all notes played by a single pitched instrument can be approximated by translating a single frequency basis function corresponding to the instrument in question. This assumption ensures the shift invariant property of frequency basis functions. The translation of the basis functions are done in frequency, typically using semitone shifts in frequency or integer divisions thereof. However, to incorporate the shift invariant property, a logarithmic frequency resolution is required, such as can be obtained via a Constant Q transform. This is used to obtain a mapping from the linear frequency domain to a logarithmic frequency domain. The CQT spectrogram acts as an input to the SNMF model. Recently, two techniques were proposed [9] to improve clustering obtained by the SNMF algorithm to reconstruct the sound sources. They are one-to-one mapping method and the masking technique to separate the basis functions. We will discuss the techniques in detail in section V.

A notable shortcoming of using SNMF algorithms is that it uses a log-frequency spectrogram and has no true inverse of log-frequency spectrogram. Despite recent improvements to approximately map the log-frequency component into linear domain, there are still room for improvements. The non-availability of exact inverse of CQT transform adversely affect the separation quality. To this end, we argue that incorporating the transform from linear to log frequency domains into the SNMF algorithm can further improve the separation of the sound sources.

The outline of the paper is as follows. The outline of the paper is as follows. Section II gives an overview of the proposed SNMF clustering algorithm. The detailed description of SNMF model is discussed in section III. The simulation details are covered in section VI followed by the discussion of results.

II Overview

The system model for the proposed algorithm is shown in figure 1. The magnitude spectrogram X of the input mixture is obtained by using the short-time Fourier transform (STFT). Then, the non-negative factorisation of X results in $\mathbf{A}$ and $\mathbf{B}$. Also a transform matrix $\mathbf{Y}$ is calculated using CQT. This is done by generating a constant Q filterbank [4]. For musical applications, a constant Q filterbank contains center frequencies that are geometrically spaced according to equal tempered scale. In general, half-tone spacing is used i.e. $\sqrt[12]{2}$. Then, the frequency basis functions contained in $\mathbf{A}$ and CQT coefficients stored in $\mathbf{Y}$ are fed into SNMF model to recover the instrument basis functions. It can be noted that, the recovered instrument basis functions are in linear domain which is different from [9]. A working model of SNMF is explained in section IV. Thereafter, the source spectrograms for individual sources are recovered using two techniques. They are Spectral masking and One-to-one mapping. The reconstruction of the synthetic sound signal corresponding to sources requires the phase information of the complex valued spectrogram. The SNMF based algorithm that uses the shift-invariant property of the frequency basis function is explained in the following sections.

III SNMF Algorithm

We will first discuss the notations and parameters used in SNMF model [5]. The notations are same as described in [10]. Tensors of any given dimension can be denoted as calligraphic upper-case letters such as $(\mathcal{R})$. The contracted product of the two tensors of finite dimension results in a tensor. However, the dimensions along with the tensors are multiplied, must be pre-defined. This can be explained as follows. Let a tensor $\mathcal{R}$ be of dimension $I_1 \times \cdots \times I_S \times L_1 \times \cdots \times L_P$ and a tensor $\mathcal{D}$ be of dimension $I_1 \times \cdots \times I_S \times J_1 \times \cdots \times J_N$. Then, the contracted tensor multiplication along the first S modes of $\mathcal{R}$ and $\mathcal{D}$ can be denoted as:

$$\langle \mathcal{R}\mathcal{D} \rangle_{\{1,\ldots,S;1,\ldots,S\}} = \sum_{i_1=1}^{I_1} \cdots \sum_{i_1=1}^{I_1} \mathcal{R} \times \mathcal{D} = \mathcal{Z} \quad (3)$$

The dimensions, along which the tensors $\mathcal{R}$ and $\mathcal{D}$ are multiplied, are specified in curly brackets. The resultant tensor $\mathcal{Z}$ will be of dimension $L_1 \times \cdots \times L_P \times J_1 \times \cdots \times J_N$. A tensor is indexed using lower case letters, such as i and is denoted by $\mathcal{R}(i, j)$. Operator $\otimes$ is used to indicate an elementwise multiplication between tensors.

a) Shifted NMF

As discussed earlier in section I, the SNMF algorithm assumes that the timbre of a note produced by an instrument does not change for all the pitches present in music. Therefore, a frequency basis function i.e. the timbre of a note can be used to characterise an instrument present in a music mixture. This property holds good due to the fact that according to the even tempered chromatic scale, the fundamental frequencies of adjacent notes are geometrically spaced. Hence, a translated frequency basis function (note) played by an instrument can be used to approximate all the notes corresponding to that instrument. However, in reality the timbre of a note does change with pitch for a particular instrument. A solution for this problem was addressed in [9]. Here, all the frequency basis functions were used to approximately define all the notes present in the mixture. Then, the clustering of frequency basis functions were performed using SNMF model.

Another drawback of using SNMF model is the need of CQT for log-frequency mapping of linear frequency NMF basis functions. The use of log-frequency spectrogram results in a deterioration of the separation quality of individual sources. In order to avoid using frequency basis functions in CQT domain we have incorporated CQT in SNMF model to improve clustering. This is explained in next section.

IV Methodology

a) Linear Approximation of SNMF

The proposed SNMF uses matrix $\mathbf{A}$ and matrix $\mathbf{Y}$ as an input parameter. Here, matrix $\mathbf{A}$ contains NMF frequency basis functions in linear domain. The NMF basis functions in $\mathbf{A}$ are obtained using equation 1. Also, the factor matrix $\mathbf{A}$ is considered as a spectrogram of NMF basis functions. Another input parameter to SNMF, the transform matrix $\mathbf{Y}$ of size $f \times n$ contains the CQT coefficients, where f is the number frequency bins. The transform matrix can be calculated by taking the fourier transform of a bank of the complex exponentials, whose centre frequencies are geometrically spaced.

The spectrogram $\mathbf{A}$ is then factorised using SNMF model to approximately determine the instrument basis functions as shown in the following equation:

$$\mathcal{A} \approx \langle \langle \mathcal{P}\mathcal{D} \rangle_{\{3,1\}} \mathcal{H} \rangle_{\{[2:3],[1:2]\}} \quad (4)$$

where, $\mathcal{P}$ is constant tensor of size $n \times k \times f$ and can be obtained using equation 5.

$$\mathcal{P} = \langle \mathcal{Y}\mathcal{R} \rangle_{\{1,1\}} \quad (5)$$

The matrices $\mathbf{A}$ and $\mathbf{Y}$ are same as tensors $\mathcal{A}$ and $\mathcal{Y}$ respectively in equation 4. Here, $\mathcal{R}$ is a translation tensor of dimension $f \times k \times f$ for k possible frequency translations. $\mathcal{R}$ translates the instrument basis functions in $\mathcal{D}$ up or down in frequency by half a tone to approximately cover all the notes played by the particular instrument. The tensor $\mathcal{D}$ of size $f \times s$ contains instrument basis functions for each source, where s is the number of sources. Tensor $\mathcal{H}$ of size $k \times s \times r$ denotes a time activation function. For example, $\mathcal{H}(i\,,\, j\,,\, :)$ indicates the time envelope for the i^{th} translation of the j^{th} source. It gives the temporal information about a given note that is being played by a particular instrument. The cost function used to obtain tensors $\mathcal{D}$ and $\mathcal{H}$ is same as used for NMF. Thus, the SNMF problem using KL divergence can be defined as

$$\langle \mathcal{L}, \mathcal{H} \rangle = \min_{\mathcal{L},\mathcal{H} \geq 0} \mathbf{D}_{kl}(\mathcal{A}||\langle \mathcal{L}\mathcal{H} \rangle_{\{2:3,1:2\}}) \quad (6)$$

where $\mathcal{L}$ denotes

$$\mathcal{L} = \langle \mathcal{P}\mathcal{D} \rangle_{\{3,1\}} \quad (7)$$

In equation 7, $\mathcal{P}$ is a constant tensor. Therefore, the problem defined in equation 6 is reduced to minimising the non-negative tensors $\mathcal{D}$ and $\mathcal{H}$.

b) *Update Equations*

The update equations for tensor $\mathcal{D}$ and tensor $\mathcal{H}$ are derived using the cost function described in 2. The iterative multiplicative updates used for the translated frequency basis functions in $\mathcal{D}$ are determined in a similar manner as done in [5]. This can be formulated as follows:

$$\mathcal{D} \leftarrow \mathcal{D} \otimes \left(\frac{\langle \langle \mathcal{P}\mathcal{A} \rangle_{\{1,1\}} \mathcal{H} \rangle_{\{[1,3],[1,3]\}}}{\langle \langle \mathcal{P}\mathcal{O} \rangle_{\{1,1\}} \mathcal{H} \rangle_{\{[1,3],[1,3]\}}} \right) \quad (8)$$

where $\mathcal{O}$ of size $n \times r$ is a tensor of all ones. Similarly, the multiplicative updates for the activation functions in $\mathcal{H}$ are calculated as follows:

$$\mathcal{H} \leftarrow \mathcal{H} \otimes \left(\frac{\langle \langle \mathcal{P}\mathcal{D} \rangle_{\{3,1\}} \mathcal{A} \rangle_{\{1,1\}}}{\langle \langle \mathcal{P}\mathcal{D} \rangle_{\{3,1\}} \mathcal{O} \rangle_{\{1,1\}}} \right) \quad (9)$$

The tensors $\mathcal{D}$ and $\mathcal{H}$ are constraint to be non-negative. This is ensured by random positive initialisation and multiplicative updates. After the factorisation, the individual instrument basis functions can be roughly reconstructed using the slices of tensor, $\mathcal{D}(:,s)$ and $\mathcal{H}(:,s,:)$. This is shown in equation 10.

$$\mathcal{A}_s \approx \langle \langle \mathcal{P}\mathcal{D}(:,s) \rangle_{\{3,1\}} \mathcal{H}(:,s,:) \rangle_{\{2:3,1:2\}} \quad (10)$$

where $\mathcal{A}_s$ denotes a spectrogram containing instrument basis functions for source s.

It is important to note that, this method of grouping of frequency basis functions is different from previously proposed methods in [9] because of the following two reasons. Firstly, the SNMF model uses the linear domain NMF basis functions as an input and the CQT transform matrix is fed into the SNMF algorithm to exploit the shift-invariant property. This is done by using the CQT transform matrix to map the linear domain NMF basis functions to the CQT domain before every iteration until the convergence is achieved. Secondly, the use of the CQT inside the SNMF model avoids the need to use the inverse CQT for recovering the NMF basis functions. As a result, the separated NMF basis functions, contained in $\mathcal{A}_s$, are in the linear domain. Thus, the linear approximation of SNMF model can be used to separate frequency basis functions corresponding to their respective sources in a given music mixture.

V Signal Reconstruction

Having obtained the instrument basis functions individual source spectrogram can be reconstructed by techniques used in [9]. There are two methods used for the reconstruction of the synthetic sound sources.

a) *One-to-one mapping*

The first method of reconstruction is one to one mapping. Here, the separation of the NMF frequency basis functions is done by reconstructing the individual source spectrograms $\mathbf{A}_s$. This is done as follows. The energy of individual frame in each spectrogram $\mathbf{A}_s$ for s number of sources is calculated. Subsequently, the frequency basis functions in the original NMF matrix $\mathbf{A}$ is assigned to the source that has the highest energy at that frame. This can be formulated as follows:

$$\mathbf{E} = \sum^{n} \mathbf{A}_s \quad (11)$$

where, matrix $\mathbf{E}$ of size $r \times s$ contains the energy of each frame of each spectrogram corresponding to the individual sources. The individual basis function is indexed by δ_s corresponding to respective sources. For a particular source s, δ_s can be defined by the following equation:

$$\delta_s(r) = argmax_s\left(E(r,:)\right) \quad (12)$$

The index vector δ_s is of length r and it contains 0 and 1. The contents of δ_s are repeated along the rows and columns to match the number elements corresponding to $\mathbf{A}$ and $\mathbf{B}$. This results in two matrixes δ_{s1} of size $n \times r$ and δ_{s2} of size $r \times m$. Then, the complex valued source spectrogram, $\mathbf{X}_s$ corresponding to source s is obtained as follows:

$$\mathbf{X}_s = \mathbf{X} \cdot \left(\frac{(\mathbf{A} \cdot \delta_{s1})(\delta_{s2} \cdot \mathbf{B})}{\mathbf{AB}} \right) \quad (13)$$

Operator $\cdot$ is used to denote an elementwise multiplication between matrixes. Further, the individual sources are obtained using inverse STFT.

b) *Spectral Masking*

The second method of source reconstruction is that of spectral masking. Having obtained individual spectrogram for frequency basis functions $\mathbf{A}_s$, the source spectrogram can be reconstructed

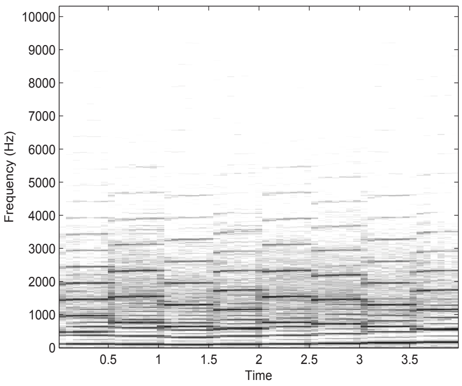

Figure 2: Spectrogram of a input mixture signal

using spectral masking as detailed in [9]. The estimated source spectrograms are used to generate individual source masking filters. The masking filter is then applied to the frequency basis functions obtained using NMF. Here, the individual source filters are created using $\mathbf{A}_s$. Then, the original matrix $\mathbf{A}$ is filtered using source filters to reconstruct the source frequency basis functions $\hat{\mathbf{A}}_s$. The calculation of $\hat{\mathbf{A}}_\mathbf{s}$ is shown in the equation:

$$\hat{\mathbf{A}}_s = \mathbf{A} \cdot \left(\frac{\mathbf{A}_s^{\cdot 2}}{\sum^s \mathbf{A}_s^{\cdot 2}} \right) \tag{14}$$

As each row vector in $\mathbf{A}$ has a corresponding column vector in $\mathbf{B}$, clustering of the time activations is handled automatically. Then, the source magnitude spectrogram is obtained as follows:

$$X_s = \hat{\mathbf{A}}_s \mathbf{B}_s \tag{15}$$

The phase information for the source spectrograms is retrieved from the original complex valued spectrogram $\mathbf{X}$ as shown in equation 16.

$$\mathbf{X}_s = \mathbf{X} \cdot \left(\frac{X_s^{\cdot 2}}{\sum^s X_s^{\cdot 2}} \right) \tag{16}$$

where $\mathbf{X}_s$ are the complex valued source spectrograms. Thereafter, the individual sound sources can be reconstructed using inverse STFT. In the following section, we will discuss the details about the test mixture and simulation setup used for the experiments detailed in this paper.

VI Simulation setup

The test mixtures consists of a dataset of 25 monaural music mixtures, which were used for all subsequent evaluations in this paper. The 25 test signals were the mixtures of 2 instruments. They were generated by mixing of a large library of orchestral samples of notes and chords produced by

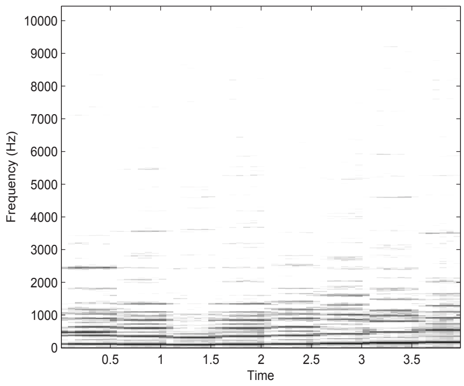

Figure 3: Spectrogram of the separated source 1.

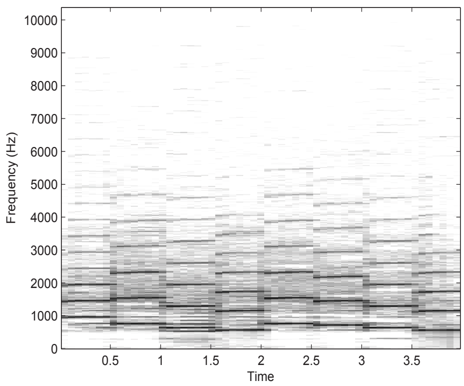

Figure 4: Spectrogram of the separated source 2

a total of 15 pitched instruments [11]. The sampling rate of the input mixtures were 44.1 kHz and were of 4 to 8 seconds in length. The notes played by different instruments in the test mixtures are in harmony and are overlapping in time and frequency. This ensures that the SNMF algorithms are tested to separate notes played simultaneously by pitched instruments. The details, of how the dataset were mixed together, are listed in [11].

The STFT was used to obtain the magnitude spectrogram of the input signal with a 75% overlapping of commonly used Hann window of 4096 samples in length. The number of basis functions for NMF basis functions were set to 13 to cover all the notes played in the mixture.

Matrices $\mathbf{A}$ and $\mathbf{B}$ and tensors $\mathcal{D}$ and $\mathcal{H}$ were randomly initialised. 24 frequency bins per octave ranging from $55Hz$ to $22.05kHz$ were used to determine the transform matrix $\mathcal{Y}$. The number of sources in the SNMF algorithm was set to 2. The NMF and SNMF algorithms ran for 300 and 50 iterations respectively. The number of allowable frequency shifts k, for translating basis functions, were ranged from 5 to 12.

Figure 2 shows the magnitude spectrogram of a test signal containing music signals of two pitched instruments. Figure 3 and figure 4 show the spectrogram of reconstructed sound source of a test mixture. Through visual inspection, it can be concluded that the linear approximation of SNMF can be used to separate frequency basis functions corresponding to sources in monaural mixture. The performance of the proposed SNMF algorithms is evaluated in the following section.

VII Results

We will compare the results of the proposed method i.e. linear approximation of SNMF algorithm (SNMF$_{lmask}$) with the recently proposed SNMF algorithms (SNMF$_{mask}$ and SNMF$_{map}$) [9]. It is important to note that SNMF$_{mask}$ used spectral masking to reconstruct the original signal and SNMF$_{map}$ makes use of one-to-one mapping to recover the source spectrogram. Both SNMF$_{mask}$ and SNMF$_{map}$ use log frequency spectrogram of NMF basis functions as an input to SNMF model.

SNMF algorithm	SDR	SIR	SAR
SNMF$_{map}$	7.69	20.61	8.83
SNMF$_{mask}$	10.25	27.15	10.87
SNMF$_{lmask}$	11.11	32.13	11.47

Table 1: Mean SDR, SIR and SAR for separated sound sources using SNMF algorithms.

A summary of the results for all the SNMF algorithms are listed in table 1. The commonly used quality measures signal-to-distortion ratio (SDR), the signal-to-interference ratio (SIR), and the signal-to-artifacts ratio (SAR) are used for the evaluation of the different SNMF algorithms. SDR measures the amount of distortion present in the output signal, SIR calculates the interference of all the sources present in the separated signal and SAR determines the artifacts present in the separated signal. The definition of the quality measures can be found in more detail in [12]. The results are calculated by averaging the quality measures over frequency shifts k and the number of sources s, present for each mixture in the test dataset.

From the table 1, we can see that SNMF$_{lmask}$ outperforms the other listed SNMF algorithms. We can see that there is a significant improvement of separation quality with the use of SNMF$_{lmask}$ over SNMF$_{map}$. It can be concluded from the SIR score that the SNMF$_{lmask}$ performs considerably better than SNMF$_{mask}$ to remove interference between the sources in a given mixture. There is approximately 1 dB improvement on the SAR and SDR scores but on listening, the separated sound sources using SNMF$_{lmask}$ were audibly better than those of SNMF$_{mask}$. This highlights the fact that the quality measures do not correlate well with human perception of separation quality.

VIII conclusion

A SNMF based algorithm has been proposed to group NMF frequency basis functions corresponding to their respective sources. We have implemented the algorithm to use the NMF frequency basis function in linear domain as an input to SNMF model. This avoids the need of log-frequency transform (CQT) and thus there is no need of inverse CQT for the reconstruction of the synthetic signal. The CQT is incorporated in SNMF model to obtain the instrument basis functions. The presented SNMF algorithm can potentially be used to separate sound sources of a monaural music signal.

References

[1] D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorisation, " *Advances in Neural Information Processing System*, 2000, pp. 556-562.

[2] S. A. Abdullah and M. D. Plumbley, "Polyphonic transcription by non-negative sparse coding of power spectra," *Proceedings of the* 5^{th} *International Conference on Music Information Retrieval (ISMIR 2004)*, Barcelona, Spain, Oct. 2004.

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

[4] J. C. Brown and M. S. Puckette,"An efficient algorithm for the calculation of a Constant Q Transform," *Journal Acoustic Society America*, vol. 92, no. 5, pp. 26982701, 1992.

[5] D. FitzGerald, M. Cranitch and E. Coyle, "Shifted Non-negative matrix factorisation for sound source separation," *IEEE Workshop of Statistical Signal Processing, Bordeaux*, France, 2005.

[6] D. FitzGerald, M. Cranitch and E. Coyle, "Extended Non-negative Tensor Factorisation Models for Musical Sound Source Separation," *Computational Intelligence and Neuroscience*, Hindawi Publishing Corp., 2008.

[7] T. Virtanen, "Sound Source Separation Using Sparse Coding with Temporal Continuity Objective," *International Computer Music Conference*, 2003.

[8] M. Spiertz and V. Gnann, "Source-Filter based clustering for monaural blind source

separation," *Proceedings of the* 12^{th} *International Conference on Digital Audio Effects*, Italy, 2009.

[9] Jaiswal, R., FitzGerald, D., Coyle E., and Rickard, S. "Clustering NMF basis functions using Shifted NMF for monaural sound source separation," *in Proc. IEEE Int. Conference on Acoustic Speech and Signal Processing ICASSP*, May, 2011.

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

[11] P. Siedlaczek, "Advanced Orchestra Library Set," 1997.

[12] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech and Language Processing*, 2006.