SINGLE CHANNEL SPEECH MUSIC SEPARATION USING NONNEGATIVE MATRIX FACTORIZATION AND SPECTRAL MASKS

Emad M. Grais and Hakan Erdogan

Faculty of Engineering and Natural Sciences,
Sabanci University, Orhanli Tuzla, 34956, Istanbul.
Email: grais@su.sabanciuniv.edu, haerdogan@sabanciuniv.edu

ABSTRACT

A single channel speech-music separation algorithm based on nonnegative matrix factorization (NMF) with spectral masks is proposed in this work. The proposed algorithm uses training data of speech and music signals with nonnegative matrix factorization followed by masking to separate the mixed signal. In the training stage, NMF uses the training data to train a set of basis vectors for each source. These bases are trained using NMF in the magnitude spectrum domain. After observing the mixed signal, NMF is used to decompose its magnitude spectra into a linear combination of the trained bases for both sources. The decomposition results are used to build a mask, which explains the contribution of each source in the mixed signal. Experimental results show that using masks after NMF improves the separation process even when calculating NMF with fewer iterations, which yields a faster separation process.

Index Terms— Source separation, single channel source separation, semi-blind source separation, speech music separation, speech processing, nonnegative matrix factorization, and Wiener filter.

1. INTRODUCTION

The performance of any speech recognition system is very sensitive to the added music or any other signals to the speech signal. It is preferable to remove the music from the background of the speech to improve the recognition accuracy.

Single channel source separation (SCSS) is a very challenging problem because only one measurement of the mixed signal is available. Recently, there are many ideas proposed to solve this problem. Most of these ideas rely on the prior knowledge, that is "training data" of the signals to be separated. NMF has been found to be an interesting approach to be used in source separation problems, especially when the nonnegativity constraint is necessary. In [1], sparse NMF was used with trained basis vectors to separate the mixture of two speech signals. In [2], NMF with trained basis vectors and a prior model for the weights' matrix from the training data was proposed to denoise the speech signal. In [3], different NMF decompositions were done for both training and testing data and SVM classifiers were used to decide on the correspondence of the basis vectors to different source signals. An unsupervised NMF with clustering was used in [4], to separate the mixed signal without any training data or any prior information about the underlying mixed signals. In [5], NMF was used to decompose the mixed data by fixed trained basis vectors for each source in one method, and in another method the NMF was used without trained basis vectors to decompose the mixing data, but it requires human interaction for clustering the resulting basis vectors. The idea of using Wiener filter as a soft mask for SCSS problem has been proposed in many studies before. In [6, 7], a short-time power spectral density dictionary of each source is developed and the mixed signal spectrum is represented as a linear combination of these dictionary entries, then the Wiener filter was used to estimate the source signals. In [8], the training data was modeled in power spectral density domain by a Gaussian mixture model (GMM) for each source, then every model was adapted to better represent the source signals in the mixed signal, and finally the adaptive Wiener filter was used with the adapted models to estimate the source signals. Various types of spectral masks were used with matching pursuit in [9] to separate speech signals from background music signals.

This paper proposes a supervised speech music separation algorithm, which combines NMF with different masks. There are two main stages of this work, a training stage and a separation stage. In the training stage, the NMF is used to decompose the training data of each source in the magnitude spectrum domain into a basis vectors matrix and a weights matrix. In the separation stage, the NMF decomposes the mixed signals as a linear combination of the trained basis vectors from each source. The initial estimate for each source is found by combining its corresponding components from the decomposed matrices. Then these initial estimates are used to build various masks, which are used to find the contribution of every source in the mixed signal.

Our main contribution in this paper is using NMF with different types of masks to improve the separation process, which leads to a better estimate for each source from the

mixed signal. It also gives us the facility of making the separation process faster by working with NMF with fewer iterations.

The remainder of this paper is organized as follows: In section 2, a mathematical description of the problem is given. We give a brief explanation about NMF and how we use it to train the basis vectors for each source in section 3. In section 4, the separation process is represented. In the remaining sections, we represent our observations and the results of our experiments.

2. PROBLEM FORMULATION

Single channel speech-music separation problem is defined as follows: Assume we observe a signal $x(t)$, which is the mixture of two sources speech $s(t)$ and music $m(t)$. The source separation problem aims to find estimates for $s(t)$ and $m(t)$ from $x(t)$. Algorithms presented in this paper are applied in the short time Fourier transform (STFT) domain. Let $X(t,f)$ be the STFT of $x(t)$, where t represents the frame index and f is the frequency-index. Due to linearity of the STFT, we have:

$$X(t,f) = S(t,f) + M(t,f), \tag{1}$$

$$|X(t,f)|\, e^{j\phi_X(t,f)} = |S(t,f)|\, e^{j\phi_S(t,f)} + |M(t,f)|\, e^{j\phi_M(t,f)}. \tag{2}$$

In this work, we assume the sources have the same phase angle as the mixed signal for every frame, that is $\phi_S(t,f) = \phi_M(t,f) = \phi_X(t,f)$. This assumption was shown to yield good results in earlier work. Thus, we can write the magnitude spectrogram of the measured signal as the sum of source signals' magnitude spectrograms.

$$\boldsymbol{X} = \boldsymbol{S} + \boldsymbol{M}. \tag{3}$$

Here $\boldsymbol{S}$ and $\boldsymbol{M}$ are unknown magnitude spectrograms, and need to be estimated using observed data and training speech and music spectra. The magnitude spectrogram for the observed signal $x(t)$ is obtained by taking the magnitude of the DFT of the windowed signal.

3. NON-NEGATIVE MATRIX FACTORIZATION

Non-negative matrix factorization is an algorithm that is used to decompose any nonnegative matrix $\boldsymbol{V}$ into a nonnegative basis vectors matrix $\boldsymbol{B}$ and a nonnegative weights matrix $\boldsymbol{W}$.

$$\boldsymbol{V} \approx \boldsymbol{BW}. \tag{4}$$

Every column vector in the matrix $\boldsymbol{V}$ is approximated by a weighted linear combination of the basis vectors in the columns of $\boldsymbol{B}$, the weights for basis vectors appear in the corresponding column of the matrix $\boldsymbol{W}$. The matrix $\boldsymbol{B}$ contains nonnegative basis vectors that are optimized to allow the data in $\boldsymbol{V}$ to be approximated as a nonnegative linear combination of its constituent vectors. $\boldsymbol{B}$ and $\boldsymbol{W}$ can be found by solving the following minimization problem:

$$\min_{\boldsymbol{B},\boldsymbol{W}} C\left(\boldsymbol{V}, \boldsymbol{BW}\right), \tag{5}$$

subject to elements of $\boldsymbol{B}, \boldsymbol{W} \geq 0$.

Different cost functions C lead to different kinds of NMF. In [10], two different cost functions were analyzed. The first cost function is the Euclidean distance between $\boldsymbol{V}$ and $\boldsymbol{BW}$ given as follows:

$$\min_{\boldsymbol{B},\boldsymbol{W}} \left(\|\boldsymbol{V} - \boldsymbol{BW}\|_2^2 \right), \tag{6}$$

where

$$\|\boldsymbol{V} - \boldsymbol{BW}\|_2^2 = \sum_{i,j} \left(\boldsymbol{V}_{i,j} - (\boldsymbol{BW})_{i,j} \right)^2.$$

The second cost function is the divergence of $\boldsymbol{V}$ from $\boldsymbol{BW}$, which yields the following optimization problem:

$$\min_{\boldsymbol{B},\boldsymbol{W}} D\left(\boldsymbol{V} \,||\, \boldsymbol{BW}\right), \tag{7}$$

where

$$D\left(\boldsymbol{V} \,||\, \boldsymbol{BW}\right) = \sum_{i,j} \left(\boldsymbol{V}_{i,j} \log \frac{\boldsymbol{V}_{i,j}}{(\boldsymbol{BW})_{i,j}} - \boldsymbol{V}_{i,j} + (\boldsymbol{BW})_{i,j} \right).$$

The second cost function was found to work well in audio source separation [2], so we only use it in this paper. The NMF solution for equation (7) can be computed by alternating updates of $\boldsymbol{B}$ and $\boldsymbol{W}$ as follows:

$$\boldsymbol{B} \leftarrow \boldsymbol{B} \otimes \frac{\frac{\boldsymbol{V}}{\boldsymbol{BW}} \boldsymbol{W}^T}{\mathbf{1}\boldsymbol{W}^T}, \tag{8}$$

$$\boldsymbol{W} \leftarrow \boldsymbol{W} \otimes \frac{\boldsymbol{B}^T \frac{\boldsymbol{V}}{\boldsymbol{BW}}}{\boldsymbol{B}^T \mathbf{1}}, \tag{9}$$

where $\mathbf{1}$ is a matrix of ones with the same size of $\boldsymbol{V}$, the operations $\otimes$ and all divisions are element wise multiplication and division respectively.

3.1. Training the bases

Given a set of training data for speech and music signals, the STFT is computed for each signal, and the magnitude spectrogram $\boldsymbol{S}_{\text{train}}$ and $\boldsymbol{M}_{\text{train}}$ of speech and music signals are calculated respectively. The goal now is to use NMF to decompose these spectrograms into bases and weights matrices as follows:

$$\boldsymbol{S}_{\text{train}} \approx \boldsymbol{B}_{\text{speech}} \boldsymbol{W}_{\text{speech}}. \tag{10}$$

$$\boldsymbol{M}_{\text{train}} \approx \boldsymbol{B}_{\text{music}} \boldsymbol{W}_{\text{music}}. \tag{11}$$

We use the update rules in equations (8) and (9) to solve equations (10) and (11). $\boldsymbol{S}_{\text{train}}$ and $\boldsymbol{M}_{\text{train}}$ have normalized columns, and after each iteration, we normalize the columns of $\boldsymbol{B}_{\text{speech}}$ and $\boldsymbol{B}_{\text{music}}$. All the matrices $\boldsymbol{B}$ and $\boldsymbol{W}$ are initialized by positive random noise. The best number of basis vectors depends on the application, the signal type and dimension. Hence, it is a design choice: Larger number of basis vectors may result in lower approximation error, but may result in overtraining and/or a redundant set of basis and require more computation time as well. Thus, there is a desirable number of bases to be chosen for each source .

4. SIGNAL SEPARATION AND MASKING

After observing the mixed signal $x(t)$, the magnitude spectrogram $\boldsymbol{X}$ of the mixed signal is computed using STFT. The goal now is to decompose the magnitude spectrogram $\boldsymbol{X}$ of the mixed signal as a linear combination with the trained basis vectors in $\boldsymbol{B}_{\text{speech}}$ and $\boldsymbol{B}_{\text{music}}$ that were found from solving equations (10) and (11). The initial estimates of the underlying sources in the mixed signal are then found as shown in section 4.1. We use the decomposition results to build different masks. The mask is applied on the mixed signal to find the underlying source signals as shown in section 4.2.

4.1. Decomposition of the mixed signal

The NMF is used again here to decompose the magnitude spectrogram matrix $\boldsymbol{X}$, but with a fixed concatenated bases matrix as follows:

$$\boldsymbol{X} \approx \left[\boldsymbol{B}_{\text{speech}} \;\; \boldsymbol{B}_{\text{music}}\right] \boldsymbol{W}, \tag{12}$$

where $\boldsymbol{B}_{\text{speech}}$ and $\boldsymbol{B}_{\text{music}}$ are obtained from solving equations (10) and (11). Here only the update rule in equation (9) is used to solve (12), and the bases matrix is fixed. $\boldsymbol{W}$ is initialized by positive random noise. The spectrogram of the estimated speech signal is found by multiplying the bases matrix $\boldsymbol{B}_{\text{speech}}$ with its corresponding weights in matrix $\boldsymbol{W}$ in equation (12). Also the estimated spectrogram of the music signal is found by multiplying the bases matrix $\boldsymbol{B}_{\text{music}}$ with its corresponding weights in matrix $\boldsymbol{W}$ in (12). The initial spectrograms estimates for speech and music signals are respectively calculated as follows:

$$\tilde{\boldsymbol{S}} = \boldsymbol{B}_{\text{speech}} \boldsymbol{W}_S. \tag{13}$$

$$\tilde{\boldsymbol{M}} = \boldsymbol{B}_{\text{music}} \boldsymbol{W}_M. \tag{14}$$

Where $\boldsymbol{W}_S$ and $\boldsymbol{W}_M$ are submatrices in matrix $\boldsymbol{W}$ that correspond to speech and music components respectively in equation (12).

4.2. Source signals reconstruction and masks

Typically, in the literature $\tilde{\boldsymbol{S}}$ and $\tilde{\boldsymbol{M}}$ are directly used as final estimates of the source signal spectrograms. However, the two estimated spectrograms $\tilde{\boldsymbol{S}}$ and $\tilde{\boldsymbol{M}}$ may not sum up to the mixed spectrogram $\boldsymbol{X}$. Especially since we enforce the NMF algorithm to deal with fixed bases, we usually get nonzero decomposion error. So, NMF gives us an approximation:

$$\boldsymbol{X} \approx \tilde{\boldsymbol{S}} + \tilde{\boldsymbol{M}}.$$

Assuming noise is negligible in our mixed signal, the component signals' sum should be directly equal to the mixed spectrogram. To make the error zero, we use the initial estimated spectrograms $\tilde{\boldsymbol{S}}$ and $\tilde{\boldsymbol{M}}$ to build a mask as follows:

$$\boldsymbol{H} = \frac{\tilde{\boldsymbol{S}}^p}{\tilde{\boldsymbol{S}}^p + \tilde{\boldsymbol{M}}^p}, \tag{15}$$

where $p > 0$ is a parameter, $(.)^p$, and the division are element wise operations. Notice that, elements of $\boldsymbol{H} \in (0, 1)$, and using different p values leads to different kinds of masks. These masks will scale every frequency component in the observed mixed signal with a ratio that explains how much each source contributes in the mixed signal such that

$$\hat{\boldsymbol{S}} = \boldsymbol{H} \otimes \boldsymbol{X}, \tag{16}$$

$$\hat{\boldsymbol{M}} = (\boldsymbol{1} - \boldsymbol{H}) \otimes \boldsymbol{X}, \tag{17}$$

where $\hat{\boldsymbol{S}}$ and $\hat{\boldsymbol{M}}$ are the final estimates of speech and music spectrograms, $\boldsymbol{1}$ is a matrix of ones, and $\otimes$ is element-wise multiplication. By using this idea we will make the approximation error zero, and we can make sure that the two estimated signals will add up to the mixed signal. The value of p controls the saturation level of the ratio which can be seen in Figure 1. The case of $p = 1$ is the linear ratio which is in the x-axis of the plot. When $p > 1$, the larger component will dominate more in the mixture, as can be seen in the figure. At $p = \infty$, we achieve a binary mask (hard mask), which will choose the larger source component for the spectrogram bin as the only component in that bin.

Two specific values of p correspond to special masks as we elaborate in the following.

4.2.1. Wiener filter

Wiener filter, which is optimal in the mean-squared sense, can be found by:

$$\boldsymbol{H}_{\text{Wiener}} = \frac{\tilde{\boldsymbol{S}}^2}{\tilde{\boldsymbol{S}}^2 + \tilde{\boldsymbol{M}}^2}, \tag{18}$$

where $(.)^2$ means taking square of every element in the matrix, also division here is element-wise. Here we use the square of the magnitude spectrum as an estimate of the power

spectral density which is required in the Wiener filter. The contribution of the speech signal in the mixed signal is

$$\hat{\boldsymbol{S}} = \boldsymbol{H}_{\text{Wiener}} \otimes \boldsymbol{X}.$$

Wiener filter works here as a soft mask for the observed mixed signal, which scales the magnitude of the mixed signal at every frequency component with values between 0 and 1 to find their corresponding frequency component values in the estimated speech signal.

4.2.2. *Hard mask*

A hard mask is obtained when $p = \infty$. It rounds the values in $\boldsymbol{H}_{\text{Wiener}}$ to ones or zeros, so we can see it as a binary mask.

$$\boldsymbol{H}_{\text{hard}} = \text{round}\left(\frac{\tilde{\boldsymbol{S}}^2}{\tilde{\boldsymbol{S}}^2 + \tilde{\boldsymbol{M}}^2}\right). \tag{19}$$

We also experimented with the linear ratio mask, that is $p = 1$ and higher order masks corresponding to $p = 3$ and $p = 4$.

After finding the contribution of the speech signal in the mixed signal, the estimated speech signal $\hat{s}(t)$ can be found by using inverse STFT to the estimated speech spectrogram $\hat{\boldsymbol{S}}$ with the phase angle of the mixed signal.

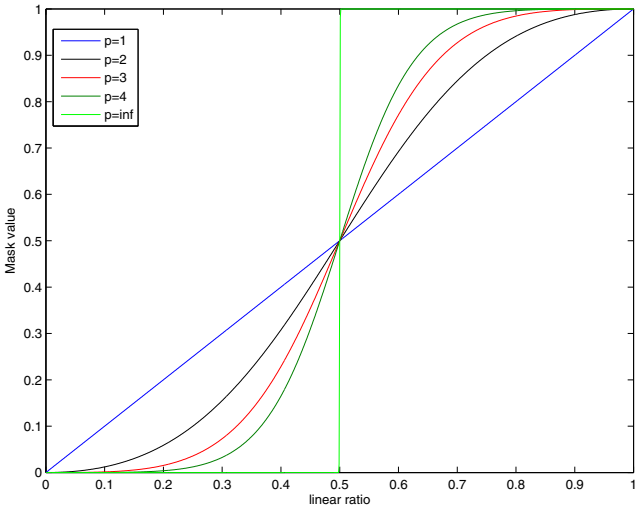
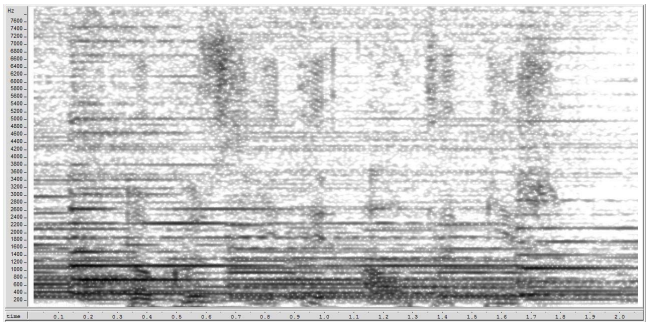


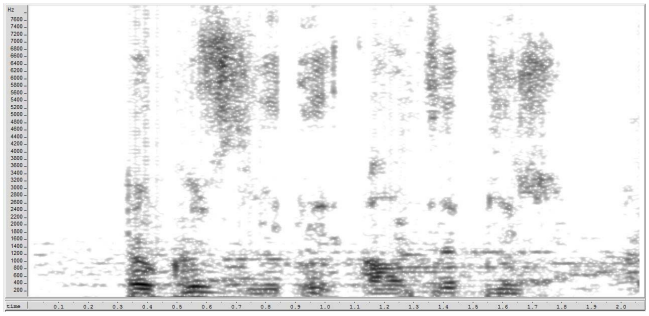
Fig. 1. The value of the mask versus the linear ratio for different values of p.

5. EXPERIMENTS AND DISCUSSION

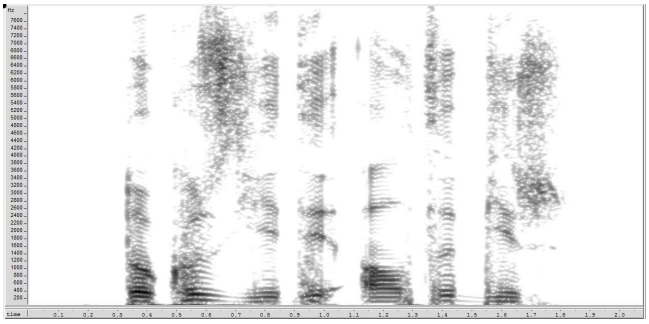
We simulated the proposed algorithms on a collection of speech and piano music data at 16kHz sampling rate. For training speech data, we used 540 short utterances from a single speaker, we used other 20 utterances for testing. For music data, we downloaded piano music from piano society



(a) The spectrogram of the mixed signal.



(b) The spectrogram of the estimated speech signal.



(c) The spectrogram of the original speech signal.

Fig. 2. The spectrograms of the mixed signal, the estimated speech signal, and the original speech signal.

web site [11]. We used 38 pieces from different composers but from a single artist for training and left out one piece for testing. The magnitude spectrograms for the training speech and music data were calculated by using the STFT: A Hamming window was used and the FFT was taken at 512 points, the first 257 FFT points only were used since the remaining points are the conjugate of the first 257 points. We trained a different number of bases N_s for the training speech signal and N_m for the training music signal. N_s and N_m take the values 32, 64, 128, and 256. The test data was formed by adding random portions of the test music file to the 20 speech utterance files at different speech-to-music ratio (SMR) values in dB. The audio power levels of each file were found using the "audio voltmeter" program from the G.191 ITU-T

Table 1. Source distortion ratio (SDR) in dB for the speech signal using NMF with Wiener filter for different numbers of bases.

SMR dB	$N_s = 256$ $N_m = 256$	$N_s = 128$ $N_m = 128$	$N_s = 256$ $N_m = 64$	$N_s = 128$ $N_m = 64$	$N_s = 64$ $N_m = 64$	$N_s = 256$ $N_m = 32$	$N_s = 128$ $N_m = 32$	$N_s = 64$ $N_m = 32$
-5	5.13	**5.34**	2.93	4.07	4.59	0.52	1.5	1.89
0	8.85	**9.68**	8.14	8.9	8.98	6.04	6.73	7.02
5	9.96	**11.15**	10.09	10.73	10.41	8.39	9.22	9.16
10	12.89	15.33	15.9	**15.99**	14.24	14.71	15.24	14.49
15	14.32	17.21	**19.56**	18.84	16.04	18.99	18.74	17
20	14.82	18.15	**22.12**	20.68	16.94	22.07	21.32	18.54

STL software suite [12]. For each SMR value, we obtained 20 test utterances this way.

Performance measurement of the separation algorithms was done using the source distortion ratio metric that is introduced in [13]. Source distortion ratio (SDR) is defined as the ratio of the target energy to all errors in the reconstruction. The target signal is defined as the projection of the predicted signal onto the original speech signal.

Table 1 shows the separation performance of using NMF with different numbers of bases N_s and N_m. We got good results at low SMR with $N_s = 128$ and $N_m = 128$. We got these results by using Wiener filter as a mask, the maximum number of iterations in NMF was 1000. The NMF iterations were stopped when the rate of change in the cost function value to the initial cost function value was less than 10^{-4}.

Table 2 shows the performance of using NMF without masks and the performance of using NMF with different kinds of masks; which shows that, we got better results when $p \geq 2$ in equation (15). These results indicate that NMF under-estimates the stronger source signal, so that boosting the stronger source component yields better performance as can be seen in Figure 1. However, one should not use hard mask since it makes binary decisions, which does not result in good performance as compared to using p with values between two to four.

In Table 3, we argue that using Wiener filter after NMF with half the number of iterations as compared to the regular NMF gives similar or better results in some cases. In other words, using Wiener with NMF with a small number of iterations can lead to the same or even better results than using NMF only with a large number of iterations. This leads to a speed up in the separation process.

Figure 2(a) shows the spectrogram of a mixture of speech and music signals, which are mixed at SMR=0dB. Figure 2(c) shows the spectrogram of the original speech signal. Figure 2(b) shows the spectrogram of the estimated speech signal from the mixture with $N_s = 128, N_m = 128$, and $p = 4$. As we can see from Figure 2(b), the proposed algorithm successfully suppresses the background music signal from the mixed signal even when the music level is high, and yields a good approximation of the speech signal with some distortions, especially at low frequencies. Audio demonstrations of our experiments are available at
http://students.sabanciuniv.edu/grais/speech/scsmsunmfasm/

Table 2. Source distortion ratio (SDR) in dB for the speech signal in case of using NMF with different masks, with $N_s = N_m = 128$.

SMR dB	No mask	Wiener filter	Hard mask	$p = 1$	$p = 3$	$p = 4$
-5	4.1	5.34	4.69	4.11	**5.41**	5.35
0	8.79	9.68	9.05	8.81	**9.72**	9.66
5	10.29	11.15	10.59	10.31	**11.22**	11.17
10	14.45	15.33	14.93	14.5	**15.52**	**15.52**
15	16.33	17.21	16.84	16.4	17.45	**17.48**
20	17.1	18.15	18.08	17.19	18.49	**18.56**

Table 3. Source distortion ratio (SDR) in dB for the speech signal in case of using NMF without a mask and with Wiener filter for a different number of iterations R, and with $N_s = N_m = 128$.

SMR dB	NMF without mask $R = 200$	NMF with Wiener filter $R = 100$
-5	2.32	3.55
0	7.18	7.33
5	8.66	8.44
10	11.55	11.32
15	14.26	12.60
20	14.79	12.93

6. CONCLUSION

In this work, we studied single channel speech-music separation using nonnegative matrix factorization and spectral masks. After using NMF to decompose the mixed signal, we built a mask from the decomposition results to find the contribution of each source signal in the mixed signal. We proposed a family of masks, which have a parameter to control the saturation level. The proposed algorithm gives better results and facilitates to speed up the separation process.

7. REFERENCES

[1] Mikkel N. Schmidt and Rasmus K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *International Conference on Spoken Language Processing (INTERSPEECH)*, 2006.

[2] K. W. Wilson, B. Raj, P. Smaragdis, and A. Divakaran, "Speech denoising using nonnegative matrix factorization with priors," in *in Proc. of ICASSP*, 2008.

[3] M. Heln and T. Virtanen, "Separation of drums from polyphonic music using nonnegtive matrix factorization and support vector machine," in *in Proc. Eur. Signal Process. Conf., Istanbul, Turkey*, 2005.

[4] T. Virtanen, "Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 15, pp. 1066–1074, Mar. 2007.

[5] B. Wang and M. D. Plumbley, "Investigating single-channel audio source separation methods based on non-negative matrix factorization," in *Proceedings of the ICA Research Network International Workshop*, 2006.

[6] L. Benaroya, F. Bimbot, G. Gravier, and G. Gribonval, "Experiments in audio source separation with one sensor for robust speech recognition," *Speech Communication*, vol. 48, no. 7, pp. 848–54, July 2006.

[7] Hakan Erdogan and Emad M. Grais, "Semi-blind speech-music separation using sparsity and continuity priors," in *International Conference on pattern recognition (ICPR)*, 2010.

[8] Ozerov A., Philippe P., Bimbot F., and Gribonval R., "Adaptation of Bayesian models for single-channel source separation and its application to voice/music separation in popular songs," *IEEE Trans. of Audio, Speech, and Language Processing*, vol. 15, 2007.

[9] Emad M. Grais and Hakan Erdogan, "Single channel speech-music separation using matching pursuit and spectral masks," in *19th IEEE Conference on Signal Processing and Communications Applications (SIU)*, 2011.

[10] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Advances in Neural Information Processing Systems*, vol. 13, pp. 556–562, 2001.

[11] URL, "http://pianosociety.com," 2009.

[12] URL, "http://www.itu.int/rec/T-REC-G.191/en," 2009.

[13] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Tr. Acoust. Sp. Sig. Proc.*, vol. 14, no. 4, pp. 1462–69, July 2006.