SINGLE CHANNEL SPEECH-MUSIC SEPARATION USING MATCHING PURSUIT 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 matching pursuit (MP) with multiple dictionaries and spectral masks is proposed in this work. A training data for speech and music signals is used to build two sets of magnitude spectral vectors of each source signal. These vectors' sets are called dictionaries, and the vectors are called atoms. Matching pursuit is used to sparsely decompose the magnitude spectrum of the observed mixed signal as a nonnegative weighted linear combination of the best atoms in the two dictionaries that match the mixed signal structure. The weighted sum of the resulting decomposition terms that include atoms from the speech dictionary is used as an initial estimate of the speech signal contribution in the mixed signal, and the weighted sum of the remaining terms for the music signal contribution. The initial estimate of each source is used to build a spectral mask that is used to reconstruct the source signals. Experimental results show that integrating MP with spectral mask gives good separation results.

Index Terms— Source separation, single channel source separation, speech music separation, speech processing, and Matching pursuit.

1. INTRODUCTION

Single channel audio source separation is an interesting and hard research problem. There are various studies focusing on separating source signals from a mixture of multiple speakers [1], multiple musical instruments [2], speech and music [3], or speech and noise [4]. In this paper, we focus on separating speech signals from background music signals.

Single channel source separation aims to separate the original source signals from only one observed mixture of these source signals. Since only single mixed signal of these source signals is available, the separation techniques usually rely on prior knowledge which is training data of each original source signal that are in the mixture. These training data are used to build a representative model for each source that can capture the characteristics of these source signals. These models can be probabilistic distributions like Gaussian mixture models (GMM) [5], or HMMs [1]. It can be also a model of a set of trained basis vectors [4]. The idea of decomposing the mixed signal with a set of trained bases for each source signal was used in previous works, and the estimate of each source signal is found by grouping the decomposion results that are related to each source bases. In [4], Non-negative matrix factorization (NMF) was used to train a set of basis vectors from a training data of each source signal, then NMF is used with these trained bases to decompose the mixed signals. In [6, 3], learned codebooks or dictionaries that consist of a number of code vectors for the source signals were used. Different optimization algorithms were used to find the best combination of codebooks' vectors that can explain the observed mixed signal. Usually the codebooks are learned using any clustering technique like k-means or using dictionary learning algorithms to optimally represent each source.

In this paper, matching pursuit and spectral masks are used to separate a mixture of speech and music signals. Given sufficient training data from each source signal we propose that, the best model of these data is the data themselves as claimed in [7]. These training data for each source are used to build two dictionaries of speech and music magnitude spectral (MS) vectors or "atoms". After observing the mixed signal, matching pursuit is used to sparsely decompose the MS of the mixed signal as a weighted linear combination of the dictionaries' atoms for each source. The weighted sum of the terms that include atoms from speech dictionary in the decomposition result is used as an initial estimate of the MS of the speech signal. The weighted sum of the remaining decomposition terms that include atoms from music dictionary is used as an initial estimate of the MS of the music signal. These initial estimates of the sources are used to build different spectral masks. These masks are used to find a new estimate of each source signal from the mixed signal. The novelty in this work is in many aspects. First, we build source specific dictionaries instead of using dictionaries of Gabor atoms that represent every source as in [8]. Second, our separation algorithm works in the frequency domain rather than in the time domain as in [8], because spectral power or magnitude is considered to be a more powerful representation of audio signals than the time domain representation. Third, we consider the linkage and smoothness between the consequent frames in our decomposition algorithm, that is instead of decomposing one MS frame every time we stack a number of consequent frames in one super vector, then every super vector is shifted by one frame at a time. This gives us a chance to decompose every frame multiple times with different neighbor frames, and then average the results. Fourth, combining the spectral mask with MP to get a better separation of the mixed signal.

The rest of this paper is organized as follows: In section 2, a mathematical description of the problem is given. In section 3, a brief explanation about matching pursuit and how we built our dictionaries is described. In section 4, an explanation of applying matching pursuit with different spectral masks in source separation is given. The remaining sections are for the results of our experiments and our observations.

2. PROBLEM FORMULATION

Single channel speech-music separation problem can be formulated as follows: Assume we have a single observed signal $y(t)$, which is the mixture of two sources speech $x(t)$ and music $m(t)$. The source separation problem aims to find estimates for $x(t)$ and $m(t)$ from $y(t)$. The framework in this paper is in the short time Fourier transform (STFT) domain. Let $Y(t,f)$ be the STFT of $y(t)$, where t represents the frame index and f is the frequency-index. Due to linearity of the STFT, we have:

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

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

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

$$M_y(t,f) = M_x(t,f) + M_m(t,f). \tag{3}$$

Here $M_x(t,f)$ and $M_m(t,f)$ are unknown magnitude spectra, and need to be estimated using observed mixed signal and training speech and music signals. The magnitude spectrum for the observed signal $M_y(t,f)$ is obtained by taking the magnitude of the DFT of the windowed signal.

3. SIGNAL DECOMPOSITION USING MATCHING PURSUIT

Assume a set of training data for speech and music signal is available. A speech dictionary $\boldsymbol{D}_s$[1] which contains normalized MS vectors or atoms $\boldsymbol{g}_s$ of the training speech signal is built, and a music dictionary $\boldsymbol{D}_m$ with MS atoms $\boldsymbol{g}_m$ is also built using the music training data. The main idea in this work is to sparsely decompose the MS of the mixed signal $\boldsymbol{y}$ as a weighted linear combination of the speech atoms $\boldsymbol{g}_s$ and the music atoms $\boldsymbol{g}_m$ according to the contribution of every source in the mixed signal.

$$\boldsymbol{y} \approx \underbrace{\sum_{\boldsymbol{D}_s} c_s \boldsymbol{g}_s}_{speech\ part} + \underbrace{\sum_{\boldsymbol{D}_m} c_m \boldsymbol{g}_m}_{music\ part}, \tag{4}$$

in matrix form

$$\boldsymbol{y} \approx \boldsymbol{D}_s \boldsymbol{c}_s + \boldsymbol{D}_m \boldsymbol{c}_m. \tag{5}$$

The coefficients' vectors $\boldsymbol{c}_s$ and $\boldsymbol{c}_m$ of the speech and music parts respectively are unknowns and need to be calculated and enforced to be sparse. Therefore, we need to solve the following L_0 norm problem:

$$\boldsymbol{c}_s, \boldsymbol{c}_m = \arg\min_{\widehat{\boldsymbol{c}}_s, \widehat{\boldsymbol{c}}_m} \left(\|\widehat{\boldsymbol{c}}_s\|_0 + \|\widehat{\boldsymbol{c}}_m\|_0 \right). \tag{6}$$

Subject to

$$\boldsymbol{y} = \boldsymbol{D}_s \boldsymbol{c}_s + \boldsymbol{D}_m \boldsymbol{c}_m,$$
$$\boldsymbol{D}_s \boldsymbol{c}_s \geq 0,$$
$$\boldsymbol{D}_m \boldsymbol{c}_m \geq 0.$$

Where $\|\boldsymbol{c}\|_0$ counts the number of non-zero entries in vector $\boldsymbol{c}$. The exact solution to (6) is not easy to find. However, a simple iterative and greedy algorithm like matching pursuit can find good approximations.

[1]The notations here are as follows: bold capital letters are for matrices, bold small letters are for vectors others are for scalars.

3.1. Matching pursuit

Matching pursuit [9] is an algorithm that approximates any signal or vector by decomposing it into a weighted linear combination of a set of basis elements called atoms. MP iteratively picks out the best atoms that can match the structure of the signal from a dictionary which contains a huge number of atoms in a greedy fashion. Let the dictionary $\boldsymbol{D} = [\boldsymbol{D}_s, \boldsymbol{D}_m]$, an atom $\boldsymbol{g}_n \in \boldsymbol{D}$, and given a signal $\boldsymbol{y}$ in $\Re^d$ space, MP tries to find a good approximation $\widehat{\boldsymbol{y}}$ as a linear combination of N atoms $\boldsymbol{g}_n$ selected from dictionary $\boldsymbol{D}$.

$$\widehat{\boldsymbol{y}} = \sum_{n=0}^{N-1} c_n \boldsymbol{g}_n \qquad \boldsymbol{g}_n \in \boldsymbol{D}, \tag{7}$$

such that

$$\left\| \boldsymbol{y} - \sum_{n=0}^{N-1} c_n \boldsymbol{g}_n \right\|_2 < \epsilon.$$

For sparse solution $N \ll d$. Let $\boldsymbol{R}^0\boldsymbol{y} = \boldsymbol{y}$, the vector $\boldsymbol{R}^0\boldsymbol{y}$ can be decomposed in the first iteration into

$$\boldsymbol{R}^0\boldsymbol{y} = < \boldsymbol{g}_0, \boldsymbol{R}^0\boldsymbol{y} > \boldsymbol{g}_0 + \boldsymbol{R}^1\boldsymbol{y}. \tag{8}$$

Where $\boldsymbol{R}^1\boldsymbol{y}$ is the residual vector after approximating $\boldsymbol{y}$ in $\boldsymbol{g}_0$'s direction, $< \boldsymbol{g}_0, \boldsymbol{R}^0\boldsymbol{y} >$ is the dot product, and $\boldsymbol{g}_0$ is atom chosen to maximize the correlation with $\boldsymbol{R}^0\boldsymbol{y}$ as

$$\boldsymbol{g}_0 = \arg\max_{\boldsymbol{g} \in \boldsymbol{D}} \left| < \boldsymbol{g}, \boldsymbol{R}^0\boldsymbol{y} > \right|.$$

The residue $\boldsymbol{R}^1\boldsymbol{y}$ in the second iteration is decomposed by projecting it on the atom that best matches it from $\boldsymbol{D}$, and a new residue $\boldsymbol{R}^2\boldsymbol{y}$ is obtained. This procedure is repeated until the norm of the residual is less than a predefined threshold or when the maximum number of iterations is achieved. After N iterations, the following approximation is constructed:

$$\boldsymbol{y} = \sum_{n=0}^{N-1} < \boldsymbol{g}_n, \boldsymbol{R}^n\boldsymbol{y} > \boldsymbol{g}_n + \boldsymbol{R}^N\boldsymbol{y}, \tag{9}$$

and the norm of the final residual is

$$\left\| \boldsymbol{R}^N\boldsymbol{y} \right\|^2 = \|\boldsymbol{y}\|^2 - \sum_{n=0}^{N-1} < \boldsymbol{g}_n, \boldsymbol{R}^n\boldsymbol{y} >^2 .$$

Since we are working with atoms and mixed signals vectors in the MS domain, we should make sure that the residual vector's components after every iteration $n \in [0, N-1]$ are nonnegative.

$$\boldsymbol{R}^n\boldsymbol{y} = \left[\boldsymbol{R}^n\boldsymbol{y}\right]_+ .$$

Where $\left[\boldsymbol{R}^n\boldsymbol{y}\right]_+ = \max\left[\boldsymbol{R}^n\boldsymbol{y}, 0\right]$. One of the advantages of matching pursuit is the complete freedom in designing the dictionary, but the performance highly depends on the structure of the dictionary.

3.2. Building the dictionaries

The magnitude spectrograms of the whole available training data for speech and music signals are found. We build two different dictionary matrices one for each source. Every column or atom $\boldsymbol{g}_s$ in the speech dictionary matrix $\boldsymbol{D}_s$ is formed by stacking its corresponding frame with its "$2L$" surrounding frames from the speech spectrogram in one super vector. Therefore, every column vector in D_s is

built by stacking "$2L+1$" neighbor frames from the speech spectrogram in one vector. For example, the column number l in the speech dictionary matrix $\boldsymbol{D}_s$ is

$$\boldsymbol{g}_s(l) = \left[\boldsymbol{m}_s^T(l-L), ..., \boldsymbol{m}_s^T(l), ..., \boldsymbol{m}_s^T(l+L)\right]^T.$$

Where $\boldsymbol{m}_s(l)$ is the magnitude spectrum column vector corresponding to frame number l of the training speech signal spectrogram. A mirror imaging at the edges of the spectrogram is performed. Any training frame which has energy less than a threshold is removed. The music dictionary matrix $\boldsymbol{D}_m$ is built from the music spectrogram in the same way. Then all atoms in both dictionaries are normalized. The two dictionaries matrices $\boldsymbol{D}_s$ and $\boldsymbol{D}_m$ are used to build one big dictionary matrix $\boldsymbol{D} = [\boldsymbol{D}_s, \boldsymbol{D}_m]$.

4. SIGNAL SEPARATION

In this section, we explain the proposed method for separating the mixed signal. We use matching pursuit and multi-dictionaries to decompose the mixed signal. Then the initial estimates of the underlying source signals in the mixed signal are found by grouping every source's atoms from the decomposition result, and the summation for every group is taken separately. These initial estimates are used to build different masks that define at each time-frequency component the ratio of every source signal in the mixed signal. The mask is used to scale the mixed signal in the STFT domain to find a smooth estimate of every source signal in the mixture.

4.1. Signal decomposition using matching pursuit

After observing the mixed signal, the spectrogram of the mixed signal is computed. A matrix $\boldsymbol{Y}$ of the observed mixed MS's frames is built. Every column of this matrix is formed by stacking its corresponding frame with its surrounding "$2L$" frames from the mixed signal spectrogram in one super vector. This is analogous to the way we constructed the dictionaries' matrices in section 3.2. For example, the column number l of the matrix $\boldsymbol{Y}$ is

$$\boldsymbol{y}(l) = \left[\boldsymbol{m}_y^T(l-L), ..., \boldsymbol{m}_y^T(l), ..., \boldsymbol{m}_y^T(l+L)\right]^T,$$

where $\boldsymbol{m}_y(l)$ is the frame number l of the observed mixed signal spectrogram. We need to decompose each column vector $\boldsymbol{y}$ in the matrix $\boldsymbol{Y}$ with the best atoms that match its structure from the dictionary matrix $\boldsymbol{D}$. We build a matrix $\boldsymbol{X}$ for the initial estimated separated speech MSs and a matrix $\boldsymbol{M}$ for the initial estimated separated music MSs.

$$\boldsymbol{Y} \approx \boldsymbol{X} + \boldsymbol{M}.$$

For every column $\boldsymbol{y}$, $\boldsymbol{x}$, and $\boldsymbol{m}$ in the matrices $\boldsymbol{Y}$, $\boldsymbol{X}$, and $\boldsymbol{M}$ the separation algorithm works as follows:

1. $n = 0$, $\boldsymbol{x} = \boldsymbol{m} = \boldsymbol{0}$, and let $\boldsymbol{R}^0\boldsymbol{y} = \boldsymbol{y}$.
2. Project $\boldsymbol{R}^n\boldsymbol{y}$ at iteration n into the dictionary $\boldsymbol{D} = [\boldsymbol{D}_s, \boldsymbol{D}_m]$.
3. Find the atom $\boldsymbol{g}_n \in \boldsymbol{D}$ that gives maximum dot product

$$\boldsymbol{g}_n^* = \arg\max_{\boldsymbol{g}\in\boldsymbol{D}} \left(\boldsymbol{R}^n\boldsymbol{y}\right)^T \boldsymbol{g},$$

$$c_n^* = \left(\boldsymbol{R}^n\boldsymbol{y}\right)^T \boldsymbol{g}_n^*.$$

4. If $\boldsymbol{g}_n^* \in \boldsymbol{D}_s$

$$\boldsymbol{x} = \boldsymbol{x} + c_n^*\boldsymbol{g}_n^*.$$

Elseif $\boldsymbol{g}_n^* \in \boldsymbol{D}_m$

$$\boldsymbol{m} = \boldsymbol{m} + c_n^*\boldsymbol{g}_n^*.$$

5. Remove $\boldsymbol{g}_n^*$ from the dictionary $\boldsymbol{D}$.
6. $\boldsymbol{R}^{n+1}\boldsymbol{y} = [\boldsymbol{R}^n\boldsymbol{y} - c_n^*\boldsymbol{g}_n^*]_+$.
7. If $\frac{\left\|\boldsymbol{R}^{n+1}\boldsymbol{y}\right\|_2^2}{\left\|\boldsymbol{R}^0\boldsymbol{y}\right\|_2^2} > \epsilon$ and $n < N-1$
   $n = n+1$; go to step 2.
   else stop.

Where N is the maximum allowed number of iterations.

As shown in step 4 in the previous algorithm, the initial estimates for every speech and music MS vectors are found by finding the weighted sum of the decomposition terms that include atoms from speech dictionary and music dictionary respectively as follows:

For speech part, we got

$$\boldsymbol{x} \approx \sum_k c_s(k)\boldsymbol{g}_s(k) \quad \boldsymbol{g}_s \in \boldsymbol{D}_s, \tag{10}$$

and for music part

$$\boldsymbol{m} \approx \sum_j c_m(j)\boldsymbol{g}_m(j) \quad \boldsymbol{g}_m \in \boldsymbol{D}_m, \tag{11}$$

where $\boldsymbol{g}_s(k)$ is the best atom from the speech dictionary $\boldsymbol{D}_s$ that match the structure of the mixed signal vector $\boldsymbol{y}$ at iteration k and $c_s(k) =< \boldsymbol{g}_s(k), \boldsymbol{R}^k\boldsymbol{y} >$ is its weight, and $\boldsymbol{g}_m(j)$ is the best atom from the music dictionary $\boldsymbol{D}_m$ that match $\boldsymbol{y}$ at iteration $j \neq k$ and $c_m(j)$ is its weight. Applying the previous procedures on all vectors in matrix $\boldsymbol{Y}$ we get the matrix $\boldsymbol{X}$ for the initial estimated separated speech MSs with $\boldsymbol{x}$ in its columns, and the matrix $\boldsymbol{M}$ for initial estimated separated music MSs with $\boldsymbol{m}$ in its columns. For example, the column number l in the matrix $\boldsymbol{X}$ is

$$\boldsymbol{x}(l) = \left[\widetilde{\boldsymbol{m}}_x^T(l-L), ..., \widetilde{\boldsymbol{m}}_x^T(l), ..., \widetilde{\boldsymbol{m}}_x^T(l+L)\right]^T,$$

where $\widetilde{\boldsymbol{m}}_x(l)$ is the frame number l of the initial estimated spectrogram of the separated speech signal from the frame $\boldsymbol{m}_y(l)$ of the mixed signal spectrogram. Notice that every frame $\widetilde{\boldsymbol{m}}_x(l)$ is differently estimated $2L+1$ times in $2L+1$ different columns $\boldsymbol{x}$ in matrix $\boldsymbol{X}$. We find the overall initial estimated spectrograms' frames $\widehat{\boldsymbol{m}}_x$ and $\widehat{\boldsymbol{m}}_m$ for each speech and music frame by averaging their corresponding frames in the $2L+1$ neighbor columns in the matrices $\boldsymbol{X}$ and $\boldsymbol{M}$ respectively.

4.2. Source signals reconstruction and masks

We can directly use the initial estimate spectrograms of the speech and music signals that are found in section 4.1 as the final estimate of every source, but the two estimated spectra $\widehat{M}_x(t,f)$ and $\widehat{M}_m(t,f)$ may not sum up to the mixture $M_y(t,f)$. We usually get nonzero decomposion error. Thus, MP gives us an approximation:

$$M_y(t,f) \approx \widehat{M}_x(t,f) + \widehat{M}_m(t,f).$$

Assuming noise is negligible in our mixed signal, the spectrogram of the source signals' sum should be directly equal to the spectrogram of the mixed signal. To make the error zero, we use the initial estimate $\widehat{M}_x(t,f)$ and $\widehat{M}_m(t,f)$ to build a mask as follows:

$$H(t,f) = \frac{\widehat{M}_x^p(t,f)}{\widehat{M}_x^p(t,f) + \widehat{M}_m^p(t,f)}, \tag{12}$$

where $p > 0$ is a parameter. Notice that elements of $\boldsymbol{H} \in (0, 1)$. Using different p values leads to different kinds of masks. When $p = 2$ the mask $H(t, f)$ is a Wiener filter. The value of p controls the saturation level of the ratio in (12). When $p > 1$, the larger component will dominate more in the mixture. At $p = \infty$, we achieve a binary mask (hard mask) which will choose the larger source component as the only component. 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

$$\hat{X}(t, f) = H(t, f)Y(t, f), \tag{13}$$

where $\hat{X}(t, f)$ is the final STFT estimate of the speech signal. After finding the contribution of the speech signal in the mixed signal, the estimated speech signal $\hat{x}(t)$ can be found by using inverse STFT of $\hat{X}(t, f)$.

5. EXPERIMENTS AND DISCUSSION

The proposed algorithm is applied on simulated mixtures of speech and music data at 16kHz sampling rate. For training speech data, 540 short utterances from a single speaker was used. We left out 20 utterances for testing. For piano music data, piano music from piano society web site [10] was downloaded. We used 38 pieces from different composers but from a single artist for training and left out one piece for the testing stage. The MS dictionaries for speech and music data were trained using the STFT, a Hamming window was used, and the FFT was taken at 512 points. We form the dictionaries' atoms as we mentioned before, we find MS from every FFT frame by using the first 257 points only since the remaining points are the conjugate of the first 257 points. Then we concatenated every five ($L = 2$) MS frames in one column vector with size (5*257) as we have mentioned in section 3.2. Each vector in the speech and music dictionaries is in 1285 dimensions (5*257). 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 STL software suite [11]. For each SMR value, we obtained 20 test utterances this way.

The reason for working with training and testing vectors of five MS frames at a time is that we got remarkable improvement by working that way rather than working with a single frame. It is obvious that this will slow down the separation algorithm. We tried to work with concatenating ten frames but the improvement was not noticeable compared to concatenating five frames, so we worked with five frames at a time for memory capacity and speed.

Performance measurement of the separation algorithms was done using metrics introduced in [12]. Projection of the predicted signal onto the original speech signal is termed as the target signal. Source distortion ratio (SDR) is defined as the ratio of the target energy to all errors in the reconstruction.

Table 1 shows the performance results for the estimated speech signal by using MP without masks and the performance of using MP with different kinds of masks. In the case of using MP without masks, we directly use the initial spectrogram estimate of speech signal that was found in section 4.1 as the final estimate of the speech source. We get better results in the case of using MP with spectral masks. However, we should not use the hard mask in the case of low speech to music ratio. The table also shows the fact that, when the speech signal dominates more in the mixed signal, then it is better to use the mask with larger p.

6. CONCLUSION

In this work, we studied single channel speech-music separation using matching pursuit with multiple dictionaries and spectral masks. We built dictionaries from training speech and music "piano" signals. The dictionaries, matching pursuit, and spectral masks were used to separate the mixed signal. To speed up the proposed algorithm and get the ability to use bigger dictionaries, our next work will focus on combining matching pursuit with kd-tree or working with tree based matching pursuit [13].

Table 1. Source/Distortion Ratio SDR in dB for speech signal of using matching pursuit(MP) with different spectral masks.

SMR dB	No mask	p=1	p=2	p=3	Hard mask
-5	2.86	**3.38**	3.23	2.90	2.39
0	7.13	**7.92**	7.80	7.56	6.97
5	9.83	**10.99**	10.90	10.68	10.17
10	13.59	15.74	**16.00**	15.87	15.45
15	14.72	17.53	**17.89**	17.75	17.30
20	16.32	20.77	21.99	**22.05**	21.79

7. REFERENCES

[1] J. R. Hershey, S. J. Rennie, P. A. Olsen, and T. T. Kristjansson, "Super-human multi-talker speech recognition: A graphical modeling approach," *Computer Speech and Language*, vol. 24, no. 1, pp. 45–66, Jan. 2010.

[2] 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.

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

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

[5] Aarthi M. Reddy and Bhiksha. Raj, "Soft Mask Methods for single-channel speaker separation," *IEEE Trans. of Audio, Speech, and Language Processing*, vol. 15, Aug. 2007.

[6] L. Benaroya, L. M. Donagh, F. Bimbot, and R. Gribonval, "Non negative sparse representation for Wiener based source separation with a single sensor," in *Proc. IEEE Conf. Acoust. Speech Sig. Proc.*, 2003.

[7] Smaragdis P., M. Shashanka, and B. Raj, "A sparse non-parametric approach for single channel separation of known sounds," in *Neural Information Processing Systems, Vancouver, BC, Canada*, Dec. 2009.

[8] N. Cho, Yu. Shiu, and Jay. Kuo, "Audio source separation with matching pursuit and content-adaptive dictionaries (mp-cad)," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 287–290, Oct. 2007.

[9] S.G. Mallat and Z. Zhang, "Matching pursuit with time frequency dictionaries," *IEEE Trans. Signal Process*, vol. 41, pp. 3397–3415, Dec. 1993.

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

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

[12] 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.

[13] P. Jost, P. Vandergheynst, and P. Frossard, "Tree-Based pursuit: Algorithm and properties," *IEEE Trans. Signal Process*, vol. 54, pp. 4685–4697, Dec. 2006.