An Adaptive Sparse Algorithm for Synthesizing Note Specific Atoms by Spectrum Analysis, Applied to Music Signal Separation

Mohammadali AZAMIAN[1], Ehsanollah KABIR[1], Sanaz SEYEDIN[2], Ellips MASEHIAN[3]
[1]*Department of Electrical and Computer Engineering, Tarbiat Modares University, 14115, Iran*
[2]*Department of Electrical Engineering, Amirkabir University of Technology, 15875, Iran*
[3]*Faculty of Engineering, Tarbiat Modares University, 14115, Iran*
ma.azamianjazi@modares.ac.ir

***Abstract*—In this paper, a sparse method is proposed to synthesize the note-specific atoms for musical notes of different instruments, and is applied to separate the sounds of two instruments coexisting in a monaural mixture. The main idea is to explore the inherent time structures of the musical notes by a novel adaptive method. These structures are used to synthesize some time-domain functions called note-specific atoms. The note-specific atoms of different instruments are integrated in a global dictionary. In this dictionary, there is only one note-specific atom for each note of any instrument, resulting in a sparse space for each instrument. The signal separation is done by mapping the mixture signal to the global dictionary. The signal related to each instrument is estimated by a summation of the mapped note-specific atoms tagged for that instrument. Experimental results demonstrate that the proposed method improves the quality of signal separation compared to a recently proposed method.**

***Index Terms*—acoustic signal processing, matching pursuit algorithms, signal reconstruction, source separation, spectral analysis.**

I. INTRODUCTION

Musical signals are typically mixtures of several sources, and separation of them is desired for audio processing tasks such as extraction of instrument sound samples, music transcription, and musical instrument identification. Processing the mixed signals is difficult and it is preferred to first extract the musical signal of each instrument and next, process the extracted signal [1]. Human's auditory system is able to extract meaningful structures in sounds by means of an advanced process, through sending sound information to the brain in meaningful structures that leads to decrease of information redundancy [2].

The difference between the basic information of sounds leads to discrepancy among them. This information can be described in the form of some basis functions which are called "Atoms" and a group of them called "dictionary" [3]. A dictionary can be constructed by means of orthogonal bases [4] or using a tight frame [5]. These dictionaries can be improved by previously developed dictionary learning methods [6-7]. Signal representation using sparse dictionaries are considered in several recently researches as an efficient method for different audio processing tasks such as audio quality enhancement [8], speech processing [9], sparse coding [10] and audio source separation [11].

To separate signals in a mixture, it is firstly represented into fundamental bases. There are two global methods of representation of signals: complete and over-complete representations. In complete representations such as Fourier and Wavelet, orthogonal functions are used. Sparse representation of signals cannot be done with these methods [12]. But in over-complete, a signal is decomposed into the atoms of a dictionary that are not necessarily orthogonal. Therefore, there might be some atoms in the dictionary that are not contributed in the representation of that signal; hence, called over-complete. In order to choose suitable atoms from the dictionary, the Matching Pursuit (MP) algorithm is usually used [13-14].

The main purpose of this study is to improve the method of extracting the inherent time structure of musical signal sources for synthesizing time domain functions, called note-specific atoms, for each musical note of any instrument. We extract the structural elements of a note by signal analysis in the frequency domain by proposing a new adaptive method employing the characteristics of musical notes in low and high frequencies. Then, we use these elements to synthesize the note-specific atom. Next, note-specific atoms of different instruments are integrated in a global dictionary. For the music signal separation, the mixture signal is projected onto note-specific atoms existing in the global dictionary. The proposed method is experimented for the test signals which are mixtures of Piano and Clarinet, as well as Guitar and Violin signals.

The structure of the paper is as follows: Section II provides a brief review of audio signal source separation in the time domain, Section III describes some principles, Section IV introduces the proposed method to construct a global dictionary for using in audio source separation, and experimental results are presented in Section V.

II. RELATED WORKS

The single-channel source separation can be done in time, frequency, or time-frequency domains. Whereas some approaches use *a priori* information such as the number of instruments and their type for better performance, some others solve the problem via a Blind Source Separation (BSS) scheme. We focus on approaches which employ *a priori* information.

Different approaches have been developed so far to solve source separation problems. In [15-16] a monaural mixture

Digital Object Identifier 10.4316/AECE.2017.02014

is decomposed into the product of basis spectra and time-varying gain factors using the Independent Component Analysis (ICA) algorithm. Signal separation is then performed by clustering the basis spectra into disjoint sets. The clustering process can be done through different methods using instrument-specific features [15] or statistical distance measures [16]. The phase information of the source signal is used to represent the source in the time domain. However, clustering is a challenging task and the phase has to be estimated for the signal representation [17-18].

The time structure of the sound sources has been obtained in [19] through learning a set of functions in time domain. These basic functions perfectly encode the sound signal using statistical methods. First, some particular basis functions are chosen and then their weights are determined using the Maximum Likelihood (ML) method. Finally, each sound source is estimated as a weighted linear summation of the basis functions (Fig. 1). This method results in a sparse representation of sources if each base has a high correlation with one of the sources and low correlation with others.

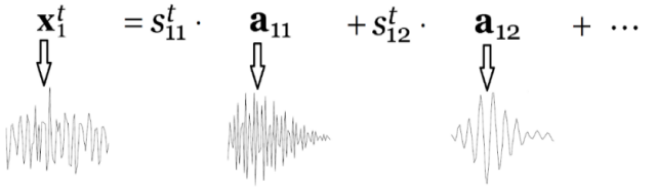

Figure 1. Learning the basis functions using maximum likelihood [19]

A main challenge in the signal separation algorithms in the time domain is signal representation. Signal separation is done well, if the mixture signal is represented to the basic elements of sources in a good manner. In [20], an auto-tagging system is developed for music that uses a "bag of systems" representation based on generative modelling. A rich dictionary of musical codewords is used, where each codeword is a generative model that captures timbral and temporal characteristics of music. Songs are represented as a BoS histogram over codewords, which allows for the use of traditional algorithms for text document retrieval to perform auto-tagging.

Signal representation is done by means of mutual information about sound sources in [21]. In this study, the sparse representation of sources is done by optimizing a criterion related to the mutual independence. Audio content representation is used as a basic component for music recommendation in [22]. They have used the traditional local features, while adding a stage of encoding with a pre-computed codebook and a stage of pooling to get compact vectorial representations.

In [23-24] an algorithm is developed to separate music signals from background signal such as speech or environmental sounds. A source-specific representation of audio signals is used in this method. For a mixture of music and speech, it is assumed that monaural mixture signal can be represented to the summation of a music signal and a speech signal, where music and speech signals are constructed from music and subspaces, respectively. The music and the speech subspaces are subsets of a universal audio space. Because the musical sound and the speech have some characteristics of harmonic elements in common, some overlap exists between their corresponding subspaces. So, the main challenge in their research is to determine a finite set of basic elements, which is highly correlated to the music signals and uncorrelated to the speech and another finite set of basic elements which is highly correlated to the speech and uncorrelated to the music signals.

An accurate mid-level representation of music signals is needed for different applications such as information retrieval and signal processing purposes. In [25], a new mid-level representation algorithm is presented which decompose a signal into a small number of sound atoms or molecules are tagged to musical instruments. Each atom is a sum of harmonic parts whose amplitudes are specific to one instrument, and each molecule is constructed from several atoms of the same instrument. Efficient algorithms are developed to extract the best correlated atoms or molecules.

An overview of dictionary-based methods (DBMs) in audio and music signal analysis is provided in [26]. DBMs provide novel ways for analyzing and visualizing audio signals, creating multi resolution descriptions of their contents, and designing sound transformations unique to a description of audio in terms of atoms.

The authors in [27] propose a subband approach to blind signal separation in time domain. The mixtures are split to some predefined subbands and then time-domain source separation is performed in each subband. The recovered sources are then reconstructed from the subbands.

A novel time-domain algorithm is proposed in [28] that is based on a complete unconstrained decomposition of the observation space. The decomposition process is performed by an independent component analysis (ICA) algorithm. The components are then combined by an appropriate reconstruction procedure. A similar research is done in [29], uses unsupervised learning from musical sounds.

Representation of a mixture of singing voice and music sound is accomplished by learned dictionaries in [30]. The magnitude spectrogram of a song is considered as two components, a sparse component and a superposition of a low-rank component, which respectively correspond to the vocal part and the instrumental part of the song. For estimation of the subspace structures of music sources, some dictionary learning algorithms have been used and a novel algorithm is proposed that employs the learned dictionaries for decomposition of the mixed signal.

The audio source separation is accomplished in [31] by sparse representation of music signals using "source-specific" dictionaries. The main purpose in that research is to separate music signals from background signals or speech. First, the signal of each note is decomposed to some elementary Gabor atoms using MP. Then, the Gabor atoms which have higher correlation with the signal are selected for synthesizing new ones. The source-specific dictionary is then produced by integrating these new atoms. The signal of each instrument can be represented using this dictionary. There is one atom for each musical note in the resulting dictionary, and therefore, a few atoms are used to represent the musical signal of each instrument. The mixed signals are decomposed into the atoms of this dictionary and thus only those parts of the signals having large correlation with the related atoms are extracted and background signals or speech are omitted.

III. Basic Concepts

In this section, details of the considered problem are explained and the Matching Pursuit algorithm is reviewed.

A. Problem definition

In audio signal source separation problem, the final goal is to extract the signal of each instrument from a mixture. Assuming that the mixture signal $X(t)$ is formed from the signals of two different instruments $S_1(t)$ and $S_2(t)$, then:

$$X(t) = S_1(t) + S_2(t) \quad (1)$$

in which $X(t)$ is available and $S_1(t)$ and $S_2(t)$ should be determined. Mathematically, this is an "underdetermined problem" and there are numerous acceptable answers for $S_1(t)$ and $S_2(t)$. Thus, *a priori* information should be used to solve this problem. In the time domain approaches, it is assumed that the signal of any instrument can be estimated as the weighted sum of some limited basis functions that have inherent features of the instrument, as follows:

$$S'_i(t) = \sum_{m=1}^{M} A_m G_{i,m}(t) \quad (2)$$

where $S'_i(t)$ is the estimated signal for source i, $G_{i,m}(t)$ is the m-th basis function (atom) for source i, A_m is the weight of m-th basis function, and M is the number of basis functions for source i. Therefore, the problem reduces to determining the weights of basis functions of each instrument for satisfaction of (1) by minimum error. However, the main challenge is defining appropriate basis functions for each instrument.

B. The MP algorithm for signal decomposition

Decomposition of signals into the atoms of a dictionary has been done using the MP algorithm as described in Algorithm 1 [9], which decomposes an input signal X into the atoms of dictionary D in N steps. The existing atoms in the dictionary D are expressed as G_i. In the beginning, the initial residual signal R_0 is set equal to X. Then decomposition is performed for N steps. At step n, the selected atom G_n is the atom which has the highest correlation to the last residual signal R_{n-1}. To update the new residual R_n, the selected atom weighted by the correlation factor is subtracted from the last existing residual signal.

Algorithm 1: MP algorithm for signal decomposition [9]

```
1) Start with the residual signal R(t) to be set
as the input signal X(t).
2) Repeat steps 3 to 5 for N stages.
3) Calculate the correlation factor between the
current residual signal and all atoms included
in the dictionary.
4) Select the atom which has the maximum
correlation with the residual signal as the
output of current stage.
5) Update the residual signal by subtracting the
selected atom in step 4 weighted by correlation
factor.
```

After decomposition, the original signal could be reconstructed as:

$$X(t) = \sum_{n=1}^{N} \langle R_{n-1}, G_n \rangle G_n(t) + R_N(t) \quad (3)$$

It is noted that the dictionary used by the MP algorithm implicitly includes all possible time-delayed functions of an atom. Generally, we assume that once an atom is included in a dictionary, all atoms resulted from its time-delayed functions also exist in that dictionary.

IV. The Proposed Method

We propose a simple method to synthesize the note-specific atoms and apply it to audio signal separation. The main idea is to explore the inherent time structures of the musical notes by employing an adaptive method. Synthesizing the note-specific atoms from a note signal is performed in two steps, as depicted in Fig. 2.

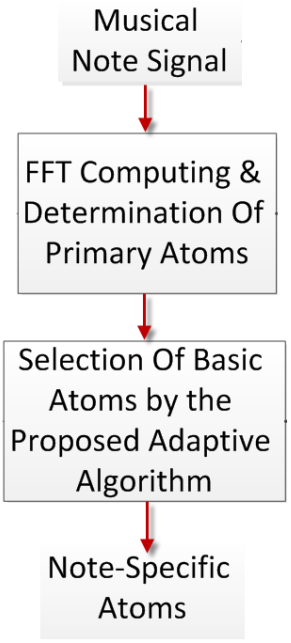

Figure 2. The steps of synthesizing note-specific atoms

Fig. 3 shows an example of the clarinet note F3 which will be followed in different stages.

If we assume the original note signal to be expressed as $S(t)$, it can be reconstructed using its FFT components as:

$$S'(t) = \frac{1}{d} \sum_{n=1}^{d} A_n \cos\left(2\pi f_n + \phi_n\right) \quad (4)$$

in which A_n, f_n and φ_n stand for amplitude, frequency, and phase of FFT components of the note signal, respectively, and d is the number of FFT components used for reconstruction. In an exact and complete representation, d is chosen as the length of spectrum and thus $S'(t)$ will be the same as $S(t)$, but a sparse representation is not achieved. A detailed look into the long term spectrum of the note signals reveals that a few components have considerable amplitude. For example, in the normalized spectrum of the clarinet note F3 in Fig. 4, the components around four frequencies have considerable amplitudes (178, 533, 888, and 1243 Hz) and other components are negligible. Thus, to achieve a sparse representation, we can employ only these components and ignore the rest.

A. Definition of the primary atoms

"Primary atoms" are defined from the FFT components of the note signal. We define a primary atom as:

$$h_{A,f,\phi}(n) = A\cos\left(2\pi fn + \phi\right); 0 \le n \le (N-1) \quad (5)$$

in which A, f and φ stand for the amplitude, frequency and

the phase of an FFT component of the note signal, respectively, and n is the sample number. The length of a primary atom, N, could be set to different numbers to achieve the best representation quality, as discussed in the next section.

In order to avoid the FFT components in reverse phase due to the symmetry, the number of primary atoms is set to half the length of the original signal.

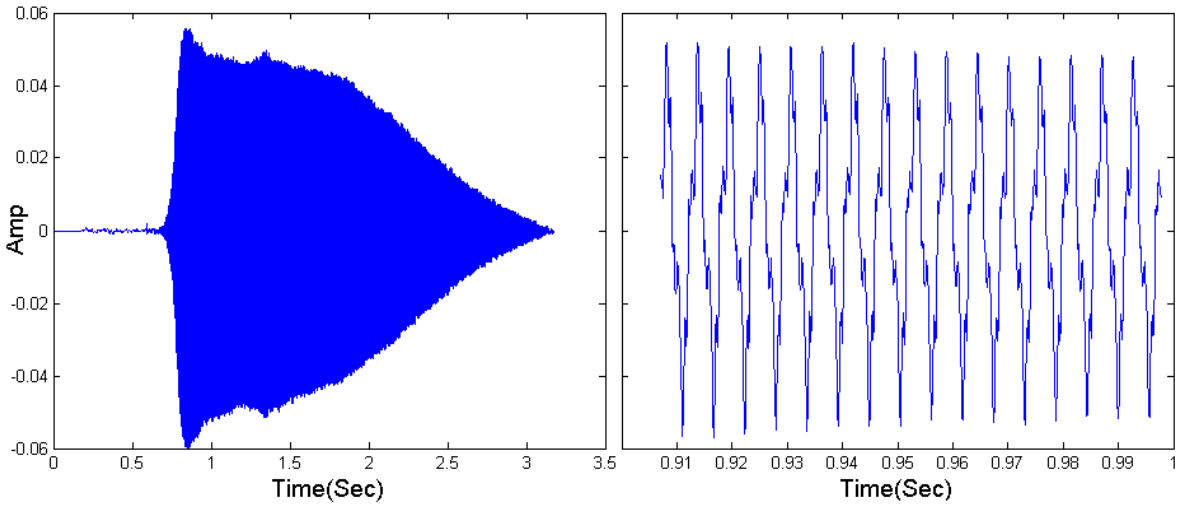

Figure 3. Clarinet note F3 signal and a zoomed view

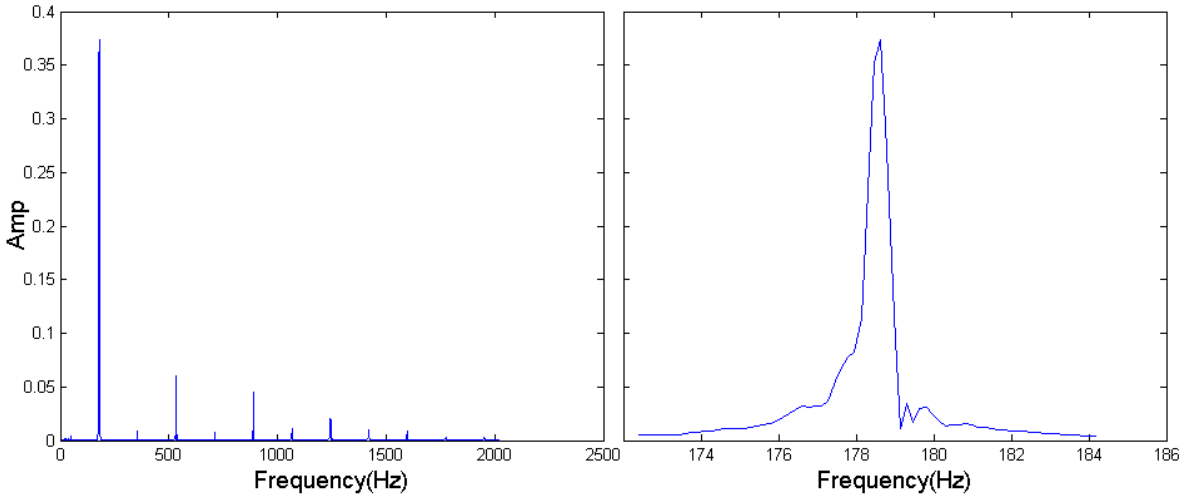

Figure 4. The spectrum of the clarinet note F3 and its more details around 178Hz

B. Selection of the basic atoms

We synthesize the note-specific atoms using only the primary atoms corresponding to the main components of the frequency spectrum, called "basic atoms", which are in the peaks of the long term spectrum. We believe that the basic atoms contain the main inherent structure of the note signal. Since there are only a few basic atoms in a note signal spectrum, the note-specific atom is sparse. For example, the number of the basic atoms for clarinet F3 note (Fig. 3) will be 3 to 5, depending on the absolute threshold we use for the amplitude.

We use an efficient and robust procedure for the computation of the main peaks in the note signal spectrum. This task is also performed in other related algorithms such as CFAR [32]. In the proposed algorithm, the primary atom which is a candidate to be a basic atom should meet three conditions to assure finding accurate atoms and avoid redundant ones: First, it must be higher than an absolute threshold. Second, it must be greater than the lateral cells within a specified neighborhood, and third, the average of its lateral cells in the specified neighborhood must be greater than a lateral threshold. These conditions can be summarized as follows:

$$A_n > TH_A \tag{6}$$

$$A_n > A_m; \quad n-L \le m \le n+L \tag{7}$$

$$\frac{1}{2L+1}\sum_{n-L}^{n+L} A_i > TH_L \tag{8}$$

where A_n is the amplitude of the primary atom, TH_A is absolute threshold, TH_L is lateral threshold, and L is neighborhood distance from each side.

The first condition is surveyed to select the primary atoms which have large amplitude. The amplitude should be greater than an absolute threshold (TH_A) as the first input parameter of the algorithm.

The purpose of the second condition is that only one atom is selected in a neighborhood. By checking this condition, we only select the atom having the highest peak in a frequency neighborhood. Surveying this condition is important to achieve sparsity. For example, consider the

spectrum details of the clarinet F3 note around 178 Hz as depicted in Fig. 4. The FFT resolution is 0.168Hz (=11025/65536). So there is more than one primary atom around the 178Hz which have large amplitude, but it is desired to select only one atom as basic atom to meet sparsity conditions. The second condition, select the primary atom whose amplitude is the highest (about 178.7 Hz in Fig. 4), and other atoms around it are omitted. The neighborhood distance around the peak atom (L), is the second input parameter of the algorithm.

Finally, the third condition should be satisfied to ensure that the signal energy is sufficient around the basic atom. So, we suggest that the integration of amplitudes around the test cell be greater than a lateral threshold (TH_L) as the third input parameter of the algorithm.

Implementation of this proposed algorithm for selecting the basic atoms is shown in Fig. 5. A sliding window sweeps the spectrum (primary atoms) and detects the cells that satisfy the conditions in Equations (6-8). Note that before applying the algorithm, the spectrum must be normalized so that the total energy of spectra is 1.

The parameters TH_A, TH_L and L should be determined as the inputs of the proposed algorithm. These parameters can be set in order to achieve the best performance in signal representation. In the following, an adaptive algorithm is proposed to set these parameters in order to obtain the best efficiency in note signal representation.

C. *Synthesizing the note-specific atoms*

The note-specific atoms are calculated as:

$$H(n) = k\,w(n)\sum_{m=1}^{M} h_m(n); \quad 0 \le n \le (N-1) \qquad (9)$$

in which $H(n)$ is a note-specific atom, $w(n)$ is a signal processing window, M is the number of basic atoms, $h_m(n)$ is the m-th basic atom chosen from primary atoms as described in the previous section, and k is normalizing factor such that the total energy of atom is 1. The length of signal processing window, $w(n)$, is the same as the length of the note-specific atom, N.

The length of a note-specific atom in the time domain, N, could be set depending on the note frequency. The best length for note-specific atom is not identical for different notes. Our experiments demonstrated that this length typically should be longer for low frequency notes and vice versa. So, for each note we synthesize note-specific atoms in three different lengths and let MP to choose the atom with the best length in the decomposition stage.

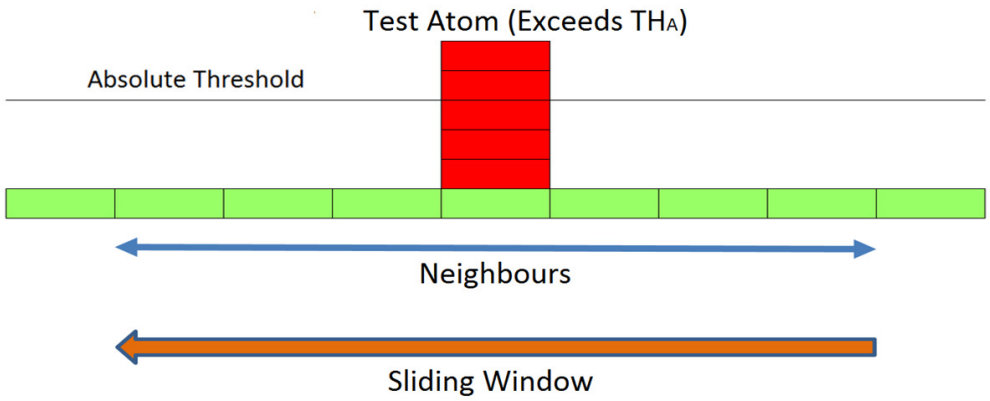

Figure 5. Peak detection by the proposed algorithm for selecting basic atoms

Common signal processing windows can be used to synthesize note-specific atom. Hamming, Gaussian and Rectangular windows are employed to synthesize three different types of note-specific atoms, and each type participates in a distinct global dictionary.

One of the challenges in the construction of dictionaries is the overlap between the instruments. The overlaps take place where the basis functions used to synthesize the atoms are source independent [31]. In the proposed algorithm, in addition to the note-specific atoms that are instrument dependent, the primary atoms which are used in synthesizing them are also note dependent. Therefore, the overlap between instruments will be minimal in the proposed method.

D. *Adaptive method for selection of the basic atoms*

To achieve the best performance, an adaptive algorithm is proposed to set the input parameters for the algorithm of basic atom selection. These parameters are optimized in order to yield the best efficiency in note signal representation.

The quality of signal representation is expressed by the Signal to Distortion Ratio (SDR), defined as follows [33]:

$$SDR = 10\log\frac{\|S\|}{\|S'-S\|} \qquad (10)$$

in which S' and S represent the extracted and original signals, respectively, and the operator $\|.\|$ denotes the sum of the squared signal samples.

For each note signal $S(t)$, firstly, the input parameters are set to some defaults. Then, the basic atoms are extracted using this parameter set and the note-specific atom is calculated as in Equation 9. A simple dictionary is constructed from this single atom and its possible time-delayed ones. In the next step, the original signal is mapped to this simple dictionary, using the MP algorithm for a predefined iteration number of N. Then, the note signal is represented by the decomposed atoms as:

$$S'(t) = \sum_{n=1}^{N} \langle R_{n-1}, G_n \rangle G_n(t) \qquad (11)$$

where $G_n(t)$ is the atom resulted from MP at step m (a possible time-delayed function of note-specific atom H_n) and $\langle R_{n-1}, G_n \rangle$ is its corresponding correlation to the residual signal at step m. The optimization criterion is the

maximization of the SDR of the representation. The procedure is repeated for all defined input parameter sets and the note-specific atom corresponding to the maximum SDR is selected as the optimum one for the current processed note.

E. Music signal separation

The global dictionary is constructed from a set of note-specific atoms synthesized for different notes of instruments. In this work, we assume that there are atoms for two instruments in the global dictionary. We synthesize note-specific atoms in three different lengths for each note. Also, more than one training data may be used in construction of global dictionary. It is notable that each atom in the dictionary is tagged with its corresponding instrument, which will be used in the separation stage. We construct global dictionaries for three types of rectangular, Hamming and Gaussian windows and use them in experiments.

Separation of music signals coexisting in a mixture is done by mapping to the atoms of a global dictionary through MP, as described in section III. We use some *a priori* information to achieve the best performance. We assume that the mixed signal is composed of only two known instruments signals and so we use a global dictionary constructed from those two Our proposed method works for the mixed music signals of the desired numbers of the instruments. However, the SDR of separation decrease when the number of instruments increases, as a result of increasing the overlap between the sources [31].

In the first step, the mixed signal is decomposed to the note-specific atoms exist in the global dictionary. Then, extracted atoms are classified into two classes according to their instrument tags. Finally, the estimated signals of two instruments will be computed by weighted summation of atoms specified to the corresponding instruments considering their time position, as follows:

$$S'_i(t) = \sum_{m=1}^{M_i} \beta_{m,i} g_{m,i}(t); \quad i = 1, 2 \tag{12}$$

where $g_{\mathrm{m,i}}$ is the m-th note-specific atom classified to instrument i and $\beta_{\mathrm{m,i}}$ is the correlation factor for this atom. The quality assessment of the separation algorithm is performed by the SDR criterion.

V. Experiments And Performance Evaluation

For the experiments, we used the RWC musical instrument sound dataset [34] and the Global Sound Bank [35] to evaluate the proposed algorithm. In the RWC, there are three variations of different musical notes, each one is made by different composer and the Global Sound Bank contains musical samples for different instruments.

Test signals were synthesized by mixing original signals and performance of separation was evaluated in SDR, by comparing original and estimated signals. We examined the separation of mixed signal of the piano and clarinet, as well as mixtures of classical guitar and violin to evaluate our proposed method.

A. Synthesizing note-specific atoms

We synthesized the note-specific atoms for all three variations of piano, clarinet, classical guitar and violin notes in the RWC dataset using our proposed method. The number of notes in the RWC is 88, 40, 64 and 78 for piano, clarinet, violin and classical guitar, respectively. The sampling frequency in the dataset is 44100 samples per second. It was reduced to 25% of the original, i.e. 11025 samples per second. The highest frequency note we used in our experiments is C8 with fundamental frequency of 4186 Hz which is less than half the sample rate, i.e. 5512.5 Hz. So the Nyquist frequency limitation is satisfied.

For each note signal, the spectrum was computed firstly. The length of the note signals we used in the experiments is lower than 65536 after down-sampling. Therefore, to calculate the long term spectrum, we equalized the lengths of all note signals to 65536 samples by zero-padding, resulting in the spectrum resolution of 0.168 Hz. After computing the spectrum, it was normalized so that the total energy is 1.

Following, the main components of the note signal spectrum was detected by the proposed algorithm. We used the proposed adaptive method to capture the best input parameter set for each note signal. The parameter set swept the range of absolute and lateral threshold value (TH_{A} and TH_{L}) from 0.001 to 0.005 in steps of 0.001 and the neighbor distance (L) from 100 to 300 in steps of 50, resulting in 125 (=5×5×5) different combinations for input parameters. There is not any efficient parameter set out of these ranges. Note-specific atoms computed for all 125 input parameter sets. The note signal, then, is represented using each of these atoms by means of MP. The iteration number of MP is set to be 1200. According to the results achieved in Section V, this iteration number is sufficient for a meaningful SDR of representation. Finally, the note-specific atom which can represent the original signal with the best SDR is selected as the final atom for the current processing note. This procedure is repeated for three types of signal processing windows.

The note-specific atom computed for the clarinet F3 note using different windows for length of 1024 samples is shown in Fig. 6. Compared to the spectrum of this note at Fig. 4, Gaussian and Hamming windows demonstrate better characteristics in the frequency domain. However, considering the experimental results, we have not observed any preference in choosing these two windows.

B. Constructing the dictionaries

We constructed global dictionaries for two instrument pairs, one for piano and clarinet and the other for classical guitar and violin. In the experiments, the lengths of the note-specific atoms are selected as N = 512, 1024, and 2048. According to our experiments, the representation performance does not increase by choosing atoms with lengths larger than 2048 and smaller than 512. Moreover, our observations showed that the best representation can be typically achieved in lengths of 2048 for lower frequencies, and in lengths of 512 for higher frequency notes. So, lengths of 512, 1024 and 2048 are used to synthesize note-specific atoms, lead to the time periods of 46.4, 92.9, and 185.8 ms, respectively, for the sample rate of 11025 Hz.

For each instrument pair, different types of dictionaries are made using different windows. We choose two training data and for each note signal we synthesized note-specific

atom in three lengths. For comparisons, we also constructed source-specific dictionaries (SSD) [31]. We used a Gabor atom dictionary to extract new atoms. To perform a fair comparison, all these new atoms were synthesized in three different lengths for two training data. The global dictionaries and the source-specific dictionaries were used for evaluation of signal separation algorithms.

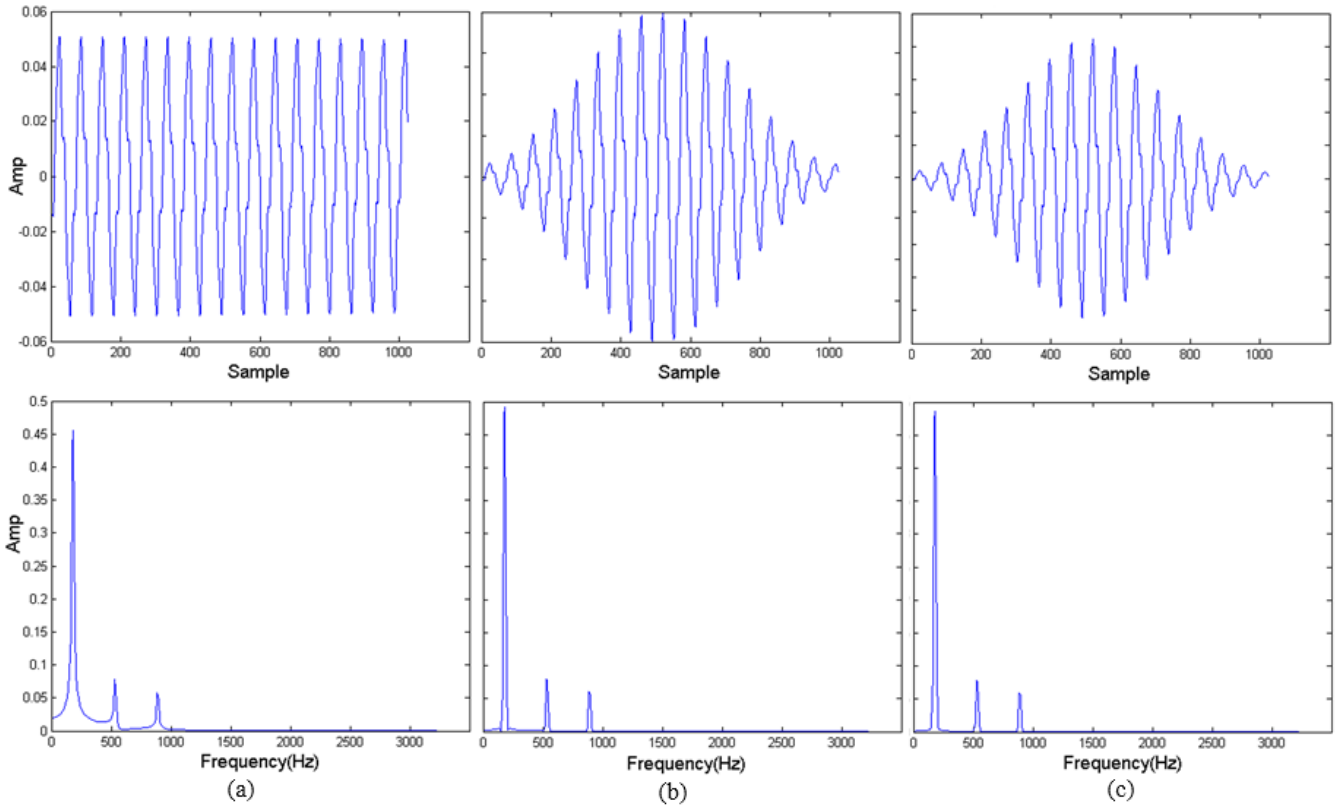

Figure 6. Atoms obtained for the clarinet note F3: a) Rectangular window, b) Hamming window, c) Gaussian window

C. Separation of the musical signal sources

We used the constructed dictionaries to separate mixed music signals of two instruments. In the first experiment, we produced mixed signals by weighted summation of note signals from two instruments. These signals were chosen from the RWC dataset. Since two variations of datasets are used in the learning process (for making the global dictionary), we used the third for making test signals. Test signal are fed to the MP algorithm with different dictionaries to evaluate different methods.

A sample of the test signals and the mixture are depicted in Fig. 7. In this sample, we synthesized the test signals for piano and clarinet and then, these signals were mixed together. The results of signal separation for this sample using global dictionary with Hamming window and source-specific dictionaries are shown in Fig. 8. In this experiment, we continued the MP algorithm for 1200 iterations.

The quality of signal separation for this sample is shown in Table I. As seen, our proposed method demonstrated better efficiency. The SDR of extracted signals versus the iteration number of MP algorithm for this test signal is shown in Fig. 9. As depicted, there in not a sensible difference in the SDR for iteration numbers above 1200.

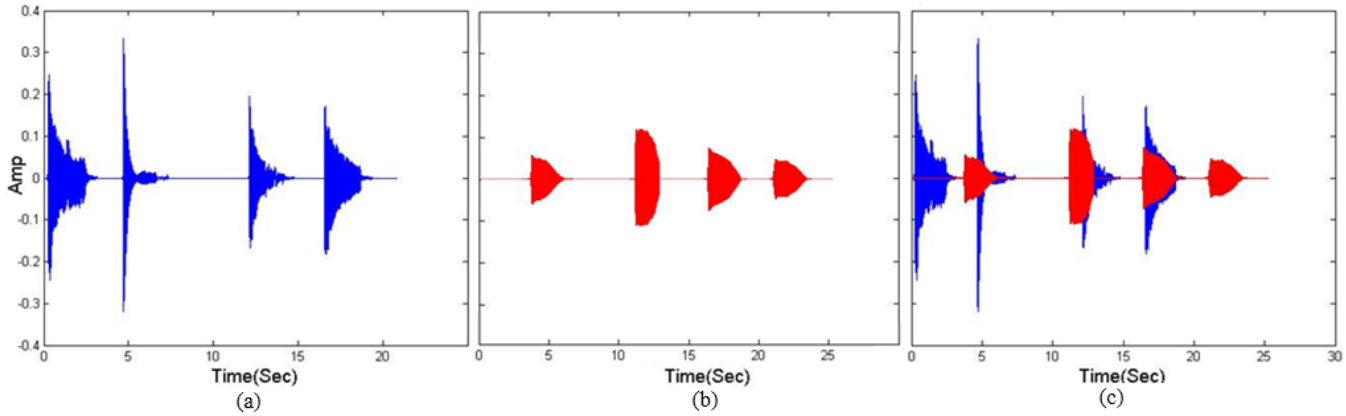

Figure 7. A sample test signal used for evaluation of the algorithm: piano signal sample (a), clarinet signal sample (b), and the mixed signal (c)

We produced different mixed signals and used them as inputs to the separation algorithm. Since we have three variations of the signals in the RWC datasets, we performed a cross validation. Each time two variations of note signals were used for making the dictionaries and the other is used for making the test signals. We compared our proposed

algorithm using the method developed in [31] which uses source-specific dictionary. We aimed to propose a new adaptive source dependent method for sound representation and source separation. Therefore, we compared the results with source-specific dictionaries as a relevant baseline, and showed that our algorithm outperforms the SSD method. Thus, comparison with source independent methods is out of the scope of this paper, since this comparison has been done previously and effectiveness of source dependent algorithms is proved in [31].

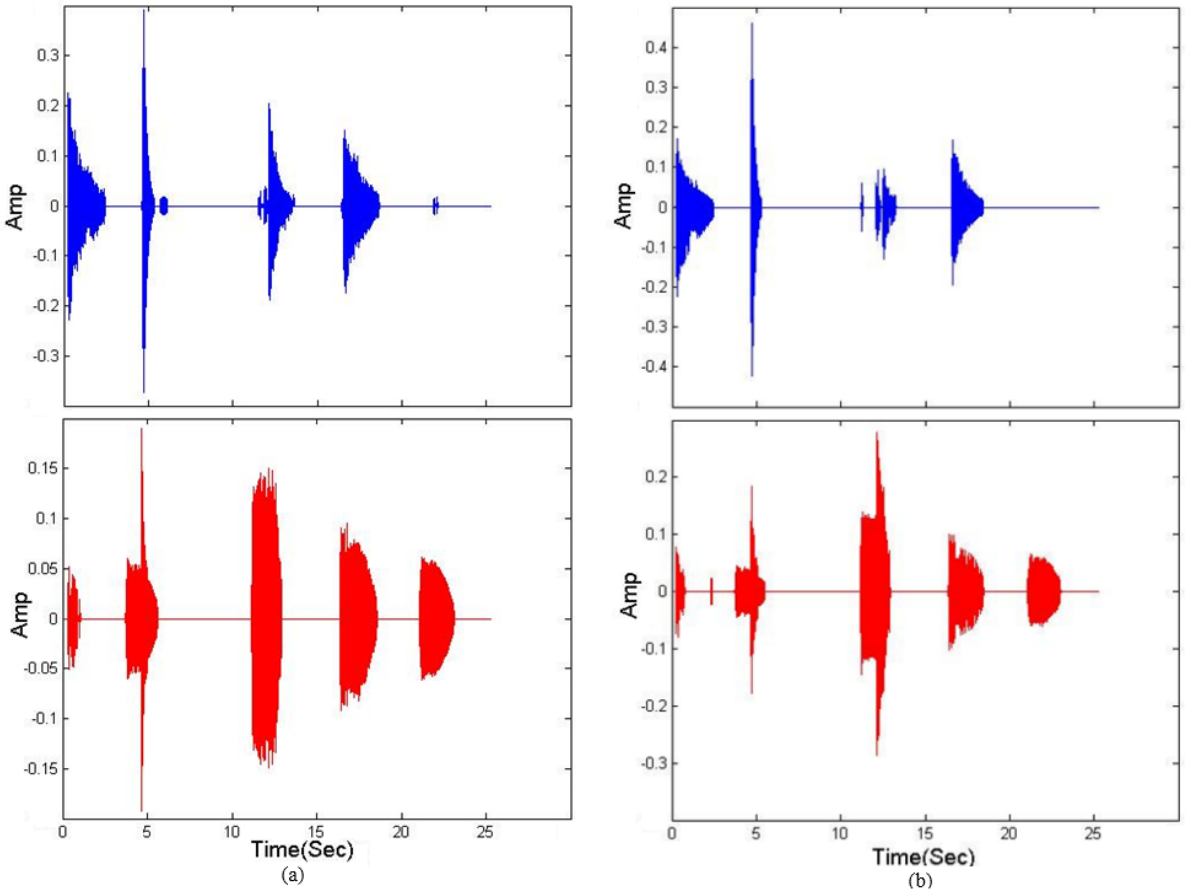

Figure 8. The Estimated signal for piano (top) and clarinet (bottom): a) proposed global dictionary, b) SSD

TABLE I. SDRS OF THE EXTRACTED SIGNALS FROM A MIXTURE OF THE PIANO AND CLARINET

Instrument	SSD[31]	Proposed method using Hamming window
Piano	2.9	5.6
Clarinet	5.7	7.9

The average SDRs achieved by different dictionaries are presented in Tables II and III for the mixtures of piano-clarinet and guitar-violin, respectively. It is clear that the proposed approach enhances the separation efficiency by 3 dB on average in the sense of SDR.

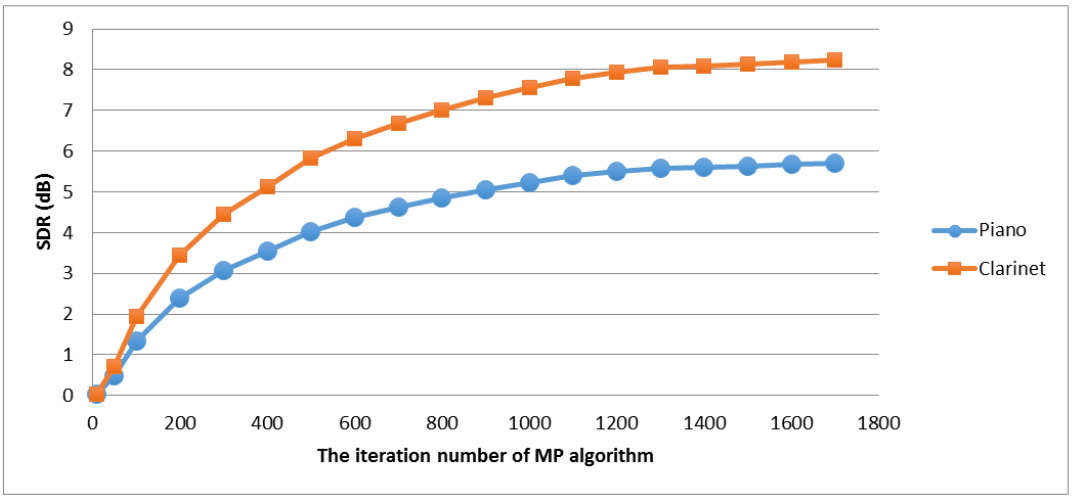

Figure 9. The SDR of the extracted signals versus the iteration number of MP algorithm for the test signal of Fig. 7

Using the signal processing windows improved the quality of the signal separation significantly. As seen in Tables II and III, the Hamming and Gaussian windows when used in synthesizing the note-specific atoms, improved the quality of signal separation, averagely 1.6 dB in the SDR criterion compared to rectangular window.

Enhancement in the quality of separation is resulted due to the following reasons. To construct the SSD, Gabor atoms are used as primary atoms for all notes of any instrument and new atoms are synthesized for each note using these global atoms. The Gabor atoms are inherently defined independently of the source. So the primary atoms are note-independent and instrument-independent, and the time-consuming MP algorithm is used to select relevant atoms for each note among global Gabor atoms. However, in our method, primary atoms are not global and are defined specifically for each note of any instrument from the FFT of note signal. Therefore, the primary atoms are more relevant to each note.

TABLE II. AVERAGE SDRS OF THE EXTRACTED SIGNALS FROM THE MIXTURES OF THE PIANO AND CLARINET

Instrument	test data	SSD	Proposed method		
			Rect.	Gauss.	Hamm.
Piano	1	3.9	4.8	6.7	6.9
	2	2.7	3.9	5.5	5.4
	3	3.4	4.6	6.0	6.3
Clarinet	1	5.9	6.9	8.8	8.8
	2	4.6	5.8	7.8	7.7
	3	5.4	6.0	8.3	8.1

TABLE III. AVERAGE SDRS OF THE EXTRACTED SIGNALS FROM THE MIXTURES OF THE CLASSICAL GUITAR AND VIOLIN

Instrument	test data	SSD	Proposed method		
			Rect.	Gauss.	Hamm.
Classical Guitar	1	1.8	2.8	4.1	4.4
	2	2.1	3.2	4.4	5.2
	3	3.3	4.2	5.3	5.4
Violin	1	1.1	3.3	3.0	3.5
	2	2.5	3.5	4.5	5.0
	3	3.0	4.0	5.7	5.7

We also select the basic atoms from the primary ones by a simple algorithm. Basic atoms are the most effective primary atoms that describe the structure of the signal note very well. In other words, for each note we find only a few basic atoms, resulting in a good sparsity. Another reason for this enhancement is due to the proposed adaptive algorithm used for synthesizing note-specific atoms. This algorithm tracks the input parameters for the appropriate selection of basic atoms which leads to the best efficiency. Thus, any inefficiency occurred due to the estimation of input parameters is eliminated or minimized.

In the next experiment, we evaluated the ability of our proposed algorithm in the separation of the musical signals from a mixture of samples recorded in more real condition. Original real samples were obtained from the Universal Sound Bank [35]. After down sampling to 11025 Hz, two types of mixtures were built, one from the piano and clarinet and the other from the guitar and violin. The global dictionary for each pair was constructed using all three variations of signal notes in the RWC dataset. The Hamming window was used to synthesize the note-specific atoms. The average SDRs for the extracted signals from the mixtures are depicted in Table IV. As seen, the separation performance decreases comparing to the results of Tables II and III. The single note signals in RWC are recorded at laboratory. So, there is not any noise or background signals in the test samples which are made by summation of single note signals and consequently, separation process for those test signals yields better results.

TABLE IV. AVERAGE SDRS FOR THE EXTRACTED SIGNALS FROM THE MIXTURES OF SAMPLES RECORDED IN MORE FACTUAL CONDITION

Mixture	SSD		Proposed Method	
	A	B	A	B
Piano (A) + Clarinet (B)	1.5	3.2	3.5	5.0
Guitar (A) + Violin (B)	1.6	0.5	3.8	2.2

Finally, we examined the extraction of the musical signals from the signals with the background sounds. Test signals are generated by mixing the musical signals and other recorded sounds taken from the Universal Sound Bank [35]. We select the animals, trains, machine and doors sound to mix to the instrument sounds. The average SDRs of the separation process for different mixtures are shown in Table V. The results demonstrate the efficiency of the proposed algorithm.

TABLE V. AVERAGE SDRS FOR EXTRACTION OF THE BACKGROUND SOUND FROM THE MUSICAL SIGNALS

Instrument Signal Mixed By Background Sound	SSD	Proposed Method
Piano	7.1	8.6
Guitar	9.0	10.8
Clarinet	10.3	11.9
Violin	8.8	9.4

The execution time of the algorithm, also, can be measured for different methods. The execution time in the separation stage is not related to the method (SSD or proposed). This is due to the fact that both of our proposed and SSD methods use the same algorithm in the separation stage, namely, the MP algorithm. Thus, the execution time is depended on the instruments existing in the mixture. Since the number of the notes of different instruments is not equal and consequently the size of their corresponding dictionary is different, the required time for projecting the mixture to the dictionaries is not the same. For example, the required time for separation of the sources in a mixture of piano and clarinet with the length of 8 minutes is 24.0 seconds when MP procedure iterates 1200 times, while this time is 30.8 seconds for a mixture of classical guitar and violin.

The execution time of different methods is different in the stage of synthesizing dictionaries. This stage is offline and is done only once for each pair of instruments and so, is not a critical and important factor. We measured this execution time for our proposed method and compared it to the SSD method in table VI. We used an Intel Core2 CPU, 2.2GHz, 32-bit PC to implement the algorithm. The operating system was windows7 and we used MATLAB 2015.

TABLE VI. CONSUMING TIME IN SECONDS FOR CONSTRUCTING DICTIONARIES

Instrument Pair	SSD	Proposed Method (Non-Adaptive)	Proposed Method (Adaptive)
Piano - Clarinet	120.0	10.3	124.6
Guitar - Violin	134.4	11.4	138.5

According to the results, the time needed for constructing dictionaries is greatly reduced if we use constant algorithm parameters. However, if we use adaptive method to obtain optimum parameters, the execution time is comparable.

VI. CONCLUSION

In this paper, a sparse method was introduced to synthesize the note-specific atoms which are used in music

signal separation. We have proposed a structural method based on the long term spectrum analysis to synthesize the note-specific atoms of different instruments, and suggested an adaptive algorithm to compute input parameters. In this algorithm, we changed the parameters of the algorithm adaptively to achieve the best separation performance in the sense of SDR. Moreover, we used shorter atoms for high-frequency and longer ones for low-frequency notes for the sake of better representation. In addition, we investigated the effects of using different signal processing windows on the algorithm efficiency.

Comparisons of the experimental results of the proposed method with a previously suggested source-specific dictionary technique indicated that our method yields a better performance in the SDR criterion for the test signals constructed from summation of single notes (See Table II and Table III). The observations also demonstrated that using the signal processing windows to synthesize the note-specific atoms, outperforms the efficiency of signal separation.

The ability of proposed method is examined in the separation of realistic music signals and also, separation of an instrument signal from background sounds. The results, demonstrated the efficiency of proposed method.

REFERENCES

[1] A. Liutkus, J. L. Durrieu, L. Daudet, and G. Richard, "An overview of informed audio source separation," WIAMIS 2013, Paris, pp. 1–4, 2013. doi:10.1109/WIAMIS.2013.6616139.

[2] H. B. Barlow, Possible Principles Underlying the Transformations of Sensory Messages, W. A. Rosenblith, Ed. The MIT Press, 2012, pp. 216–234. doi:10.7551/mitpress/9780262518420.003.0013

[3] N. Cho, C. C. J. Kuo, "Sparse representation of musical signals using source-specific dictionaries," IEEE Signal Processing Letters, vol. 17, no. 11, pp. 913-916, Nov. 2010. doi:10.1109/LSP.2010.2071864

[4] R. Gribonval, M. Nielsen, "Sparse representations in unions of bases," IEEE Transactions on Information Theory, vol. 49, no. 12, pp. 3320–3325, Dec. 2003. doi: 10.1109/TIT.2003.820031

[5] E. J. Candès, L. Demanet, "The curvelet representation of wave propagators is optimally sparse," Communications on Pure and Applied Mathematics, vol. 58, no. 11, pp. 1472–1528, Nov. 2005. doi:10.1002/cpa.20078

[6] M. Yaghoobi, T. Blumensath, M. E. Davies, "Dictionary learning for sparse approximations with the majorization method," IEEE Transactions on Signal Processing, vol. 57, no. 6, pp. 2178–2191, Jun. 2009. doi: 10.1109/TSP.2009.2016257

[7] M. Aharon, M. Elad, A. Bruckstein, "rmK-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," IEEE Transactions on Signal Processing, vol. 54, no. 11, pp. 4311–4322, Nov. 2006. doi:10.1109/TSP.2006.881199

[8] H. Huang, J. Yu, W. Sun, "Super-resolution mapping via multi-dictionary based sparse representation," Int. Conf. on Acoustics Speech and Signal Processing, IEEE, Florence, 2014, pp. 3523-3527. doi:10.1109/ICASSP.2014.6854256

[9] Y. Xu, G. Bao, X. Xu, Z.Ye, "Single-channel speech separation using sequential discriminative dictionary learning," Signal Processing, vol. 106, pp. 134–140, Jan. 2015. doi:10.1016/j.sigpro.2014.07.012

[10] M. Yaghoobi, L. Daudet, M. E. Davies, "Parametric dictionary design for sparse coding," IEEE Trans. Signal Processing, vol. 57, no. 12, pp. 4800–4810, Dec. 2009. doi:10.1109/TSP.2009.2026610

[11] M. Zibulevsky, B. A. Pearlmutter, "Blind source separation by sparse decomposition in a signal dictionary," Neural Computation, vol. 13, no. 4, pp. 863–882, Apr. 2001. doi:10.1162/089976601300014385

[12] M. M. Goodwin, M. Vetterli, "Matching pursuit and atomic signal models based on recursive filter banks," IEEE Transactions on Signal Processing, vol. 47, no. 7, pp. 1890–1902, Jul. 1999. doi:10.1109/78.771038

[13] S. G. Mallat, Z. Zhang, "Matching pursuit with time-frequency dictionaries," IEEE Transactions on Signal Processing, vol. 41, no. 12, pp. 3397–3415, Dec. 1993. doi:10.1109/78.258082

[14] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," IEEE Transactions on Information Theory, vol. 50, no. 10, pp. 2231–2242, Oct. 2004. doi:10.1109/TIT.2004.834793

[15] C. Uhle, C. Dittmar, T. Sporer, "Extraction of drum tracks from polyphonic music using independent subspace analysis," in Proc. 4th Int. Symp. Ind. Compon. Anal. Blind Signal Separation, Nara, Japan, 2003, pp. 843–848. ISBN: 4-9901531-1-1

[16] M. A. Casey, A. Westner, "Separation of mixed audio sources by independent subspace analysis," in Proc. Int. Compon. Music Conf., Berlin, Germany, 2000, pp. 154–161. Permalink: http://hdl.handle.net/2027/spo.bbp2372.2000.142

[17] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparse criteria," IEEE Trans. Audio Speech Lang. Process., vol. 15, no. 3, pp. 1066–1074, Mar. 2007. doi:10.1109/TASL.2006.885253

[18] D. FitzGerald, Automatic drum transcription and source separation, Ph.D. dissertation, Dublin Inst. Technol., Dublin, Ireland, pp. 11-62, 2004.

[19] G. J. Jang, T. W. Lee, Y. H. Oh, "Single-channel signal separation using time-domain basis functions," IEEE Signal Processing Letters, vol. 10, no.6, pp. 168-171, Jun. 2003. doi:10.1109/LSP.2003.811630

[20] K. Ellis, E. Coviello, A. B. Chan, and G. Lanckriet, "A Bag of Systems Representation for Music Auto-Tagging," IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 12, pp. 2554–2569, Dec. 2013. doi:10.1109/TASL.2013.2279318

[21] S. Choi, A. Cichocki, H. M. Park, S. Y. Lee, "Blind source separation and independent component analysis: a review," Neural Information Processing-Letters and Reviews, vol. 6, no. 1, pp. 1-57, 2005. ISBN: 89-89453-04-6

[22] Y. Vaizman, B. McFee, G. Lanckriet, "Codebook-based audio feature representation for music information retrieval," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 22, no. 10, pp. 1483-1493, 2014. doi:10.1109/TASLP.2014.2337842

[23] N. Cho, Y. Shiu, C.-C. J. Kuo, "Audio Source Separation with Matching Pursuit and Content-Adaptive Dictionaries (MP-CAD)," 2007, pp. 287–290. doi:10.1109/ASPAA.2007.4393000

[24] N. Cho, Yu Shiu, and C.-C. J. Kuo, "Efficient music representation with content adaptive dictionaries," 2008, pp. 3254–3257. doi:10.1109/ISCAS.2008.4542152

[25] P. Leveau, E. Vincent, G. Richard, and L. Daudet, "Instrument-Specific Harmonic Atoms for Mid-Level Music Representation," IEEE Transactions on Audio, Speech, and Language Processing, vol. 16, no. 1, pp. 116–128, Jan. 2008. doi:10.1109/TASL.2007.910786

[26] B. L. Sturm, C. Roads, A. McLeran, J. J. Shynk, "Analysis, visualization, and transformation of audio signals using dictionary-based methods," Journal of New Music Research, vol. 38, no. 4, pp. 325–341, Dec. 2009. doi:10.1080/09298210903171178

[27] B. S. Kirei, M. D. Topa, I. Muresan, I. Homana, N. Toma, "Blind source separation for convolutive mixtures with neural networks," Advances in Electrical and Computer Engineering, vol. 11, no. 1, pp. 63–68, 2011. doi:10.4316/AECE.2011.01010

[28] Z. Koldovsky, P. Tichavsky, "Time-domain blind separation of audio sources on the basis of a complete ICA decomposition of an observation space," IEEE Transactions on Audio, Speech, and Language Processing, vol. 19, no. 2, pp. 406–416, Feb. 2011. doi:10.1109/TASL.2010.2049411

[29] G. J. Jang and T. W. Lee, "A probabilistic approach to single channel blind signal separation," in Advances in Neural Information Processing Systems, Vancouver, BC, Canada, Dec. 2002, pp. 1178–1180.

[30] Y. Yang, "Low-rank representation of both singing voice and music accompaniment via learned dictionaries," ISMIR, 2013, pp. 427-432. ISBN: 978-0-615-90065-0

[31] N. Cho, C. C. J. Kuo, "Sparse music representation with source-specific dictionaries and its application to signal separation," IEEE Transactions on Audio Speech Lang. Process., vol. 19, no. 2, pp. 337-348, Feb. 2011. doi:10.1109/TASL.2010.2047810

[32] L. L. Scharf, C. Demeure, Statistical Signal Processing: Detection, Estimation, and Time Series Analysis, Addison-Wesley Pub. Co, 1991, pp.312-329. ISBN: 978-0201190380

[33] E. Vincent, S .Araki, P. Bofill, "The 2008 signal separation evaluation campaign: community-based approach to large-scale evaluation," in Independent Component Analysis and Signal Separation, vol. 5441, Springer Berlin Heidelberg, 2009, pp. 734-741. doi:10.1007/978-3-642-00599-2_92

[34] M. Goto, H. Hashiguchi, T. Nishimura, R. Oka, "RWC music database: musical instrument sound database," ISMIR, 2003, pp. 229-230.

[35] Universal Sound Bank [Online]. Available: http://eng.universal-soundbank.com/instruments.htm