Single Channel Music Sound Separation Based on Spectrogram Decomposition and Note Classification

Hafiz Mustafa and Wenwu Wang*

Centre for Vision, Speech and Signal Processing (CVSSP)
University of Surrey, GU2 7XH, UK
{w.wang,hm00045}@surrey.ac.uk
http://www.surrey.ac.uk/cvssp

Abstract. *A challenging problem in music sound separation is to separate multiple sources from a single channel mixture. In this paper, we propose a new approach for this problem based on non-negative matrix factorization (NMF) and note classification, assuming that the instruments used to play the sound signals are known a priori. The spectrogram of the mixture signal is first decomposed into building components (musical notes) using an NMF algorithm. The Mel frequency cepstrum coefficients (MFCCs) of both the decomposed components and the signals in the training dataset are extracted. The notes are then labelled to the corresponding type of instruments by the K nearest neighbors (K-NN) classfication algorithm based on the MFCCs feature vector. Finally, the source signals are reconstructed from the classified notes and the weighting matrices obtained from the NMF algorithm. Simulations are provided to show the performance of the proposed system.*

Key words: Non-negative matrix factorization, single-channel sound separation, Mel frequency cepstrum coefficients, instrument classification, K nearest neighbors, unsupervised learning

1 Introduction

Single-channel sound source separation addresses the issue of recovering multiple unknown sources from a one-microphone signal that is an observed mixture of these sources. The single-channel problem is an extreme case of underdetermined separation problems, which are inherently ill-posed, i.e., more unknown variables than the number of equations. To solve the problem, additional assumptions (or constraints) about the sources or the propagating channels are necessary. For an underdetermined system with two microphone recordings, it is possible to separate the sources based on spatial diversity using determined independent component analysis (ICA) algorithms and an iterative procedure [11]. However, unlike the techniques in e.g. ADRess [2] and DUET [12] that require at least two mixtures, the cues resulting from the sensor diversity are not

* The work of W. Wang was partly sponsored by the EPSRC of the UK.

available in the single channel case, and thus separation is difficult to achieve based on ICA algorithms.

In this paper, a new algorithm is proposed for the problem of single-channel music source separation. The algorithm is based mainly on the combination of note decomposition with note classification. The note decomposition is achieved by a non-negative matrix factorization (NMF) algorithm. NMF has been previously used for music sound separation and transcription, see e.g. [7], [1], [4], [13], [18], [19]. In this work, we first use the NMF algorithm in [17] to decompose the spectrogram of the music mixture into building components (musical notes). Then, 13-dimensional Mel Frequency Cepstrum Coefficients (MFCCs) feature vectors are extracted from the segmented frames of each decomposed note. To divide the separated notes into their corresponding instrument categories, the K nearest neighbor (NN) classifier [6] is used. The K-NN classifier is an algorithm that is simple to implement and also provides good classification performance. The source signals are reconstructed by combining the notes having same class labels. The remainder of the paper is organized as follows. The proposed separation system is described in Section 2 in detail. Some preliminary experimental results are shown in Section 3. Finally, Section 4 summarises the paper.

2 The Proposed Separation System

This section describes the details of the processes in our proposed sound source separation system. First, the single-channel mixture of music sources is decomposed into basic building blocks (musical notes) by applying the NMF algorithm. The NMF algorithm describes the mixture in the form of basis functions and their corresponding weights (coefficients) which represent the strength of each basis function in the mixture. The next step is to extract the feature vectors of the musical notes and then classify the notes into different source streams. Finally, the source signals are reconstructed by combining the notes with the same class labels. In this work, we assume that the instruments used to generate the music sources are known a priori. In particular, two kinds of instruments, i.e. piano and violin, were used in our study. The block diagram of our proposed system is depicted in Figure 1.

2.1 Music Decomposition by NMF

To find a suitable representation of the data is a fundamental problem in many data analysis tasks. NMF is a data-adaptive linear representation technique for 2-D matrices. Given a non-negative data matrix $\mathbf{X}$, the objective of NMF is to find two non-negative matrices $\mathbf{W}$ and $\mathbf{H}$, such that $\mathbf{X} \approx \mathbf{WH}$ [8]. In this work, $\mathbf{X}$ is an $S \times T$ matrix representing the spectrogram of the mixture signal, $\mathbf{W}$ is the basis matrix of dimension $S \times R$, and $\mathbf{H}$ is the weighting coefficient matrix of dimension $R \times T$. The number of basis used to represent the original matrix is described by R, i.e. the decomposition rank. Due to non-negativity constraints, this representation is purely additive. Many algorithms can be used

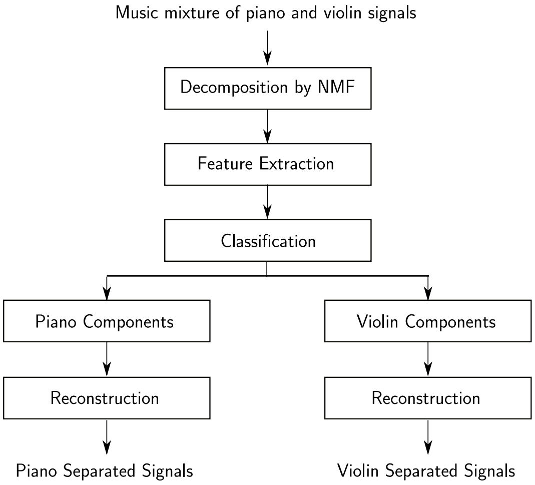

Fig. 1. Block diagram of the proposed system

to find the suitable pair of $\mathbf{W}$ and $\mathbf{H}$ such that the error of the approximation can be minimised, see e.g. [8], [9], [4], [13] and [19]. In this work, we use the algorithm proposed in [17] for the note decomposition. In comparison to the classical algorithm in [8], this algorithm considers additional constraints from the structure of the signal.

To apply the algorithm, the time-domain signal (with negative values) needs to be transform into the frequency domain using, e.g. the short-time Fourier transform (STFT). The matrix $\mathbf{X}$ is generated as the spectrogram of the signal, and in our study, the frame size of each segment equals to 40 ms, and 50 percents overlaps between the neighboring frames are used. The idea of decomposing the mixture signal is based on the observation that a music signal may be represented by a set of basic building blocks such as musical notes or other general harmonic structures. The basic building blocks are also known as basis vectors and the decomposition of the single-channel mixture into basis vectors is the first step towards the separation of multiple source signals from the single-channel mixture. If different sources in the mixture represent different basis vectors, then the separation problem can be regarded as a problem of classification of basis vectors into different categories. The source signals can be obtained by combining the basis vectors in each category.

The above mixture (or NMF) model can be equally written as

$$\mathbf{X} = \sum_{r=1}^{R} \mathbf{w}_r \mathbf{h}_r \tag{1}$$

where $\mathbf{w}_r$ is the r^{th} column of $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_R]$ which contains the basis vectors, and $\mathbf{h}_r$ is the r^{th} row of $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_R]^T$ which contains the weights or coefficients of each basis function in matrix $\mathbf{W}$. As a prior knowledge, given the mixture of musical sounds containing two sources (e.g. piano and violin), two different types of basis functions are learnt from the decomposition by the NMF algorithm. The magnitude spectrograms of the basis components (notes) of the two different sources in the mixture are obtained by multiplying the columns of the basis matrix **W** to the corresponding rows of the weight matrix **H**. The columns of matrix **W** contain the information of musical notes in the mixture and corresponding rows of matrix **H** describe the strength of notes. Some rows in **H** do not contain useful information and are therefore considered as noise. The noise components are considered separately in the classification process to improve the quality of the separated sources.

2.2 Feature Extraction

Feature extraction is a special form of dimensionality reduction by transforming the high dimensional data into a lower dimensional feature space. It is used in both the training and classification processes in our proposed system. The audio features that we used in this work are the MFCCs. The MFCCs are extracted on a frame-by-frame basis. In the training process, the MFCCs are extracted from a training database, and the feature vectors are then formed from these coefficients. In the classification stage, the MFCCs are extracted similarly from the decomposed notes obtained by the NMF algorithm. In our experiments, the frame size of 40 ms is used, which equals to 1764 samples when the sampling frequency is 44100 Hz. In each frame, a 13 dimensional MFCCs vector is computed.

2.3 Classification of Musical Notes

The main objective of classification is to maximally extract patterns on the basis of some conditions and is to separate one class from another. The K-NN classifier, which uses a classification rule without having the knowledge of the distribution of measurements in different classes, is used in this paper for the separation of piano and violin notes. The basic steps in music note classification include preprocessing, feature extraction or selection, classifier design and optimization. The main steps used in our system are detailed in Table 1.

The main disadvantage of the classification technique based on simple "majority voting" is that the classes with more frequent examples tend to come up in the K-nearest neighbors when the neighbors are computed from a large number of training examples [3]. Therefore, the class with more frequent training examples tends to dominate the prediction of the new vector. One possible technique to solve this problem is to weight the classification based on the distance from the test pattern to all of its K nearest neighbors.

2.4 K-NN Classifier

This section briefly describes the K-NN classifier used in our algorithm. K-NN is a simple technique for pattern classification and is particularly important for

Table 1. The musical note classification algorithm

1) Calculate the 13-D MFCCs feature vectors of all the musical examples in the training database with class labels. This creates a feature space for the training data.
2) Extract similarly the MFCCs feature vectors of all separated components whose class labels need to be determined.
3) Assign the labels to all the feature vectors in the separated components to the appropriate classes via the K-NN algorithm.
4) The majority vote of feature vectors determines the class label of the separated components.
5) Optimize the classification results by different choices of K.

non-parametric distributions. The K-NN classifier labels an unknown pattern x by the majority vote of its K-neatest neighbors [3], [5]. The K-NN classifier belongs to a class of techniques based on non-parametric probability density estimation. Suppose, there is a need to estimate the density function $P(x)$ from a given dataset. In our case, each signal in the dataset is segmented to 999 frames, and a feature vector of 13 MFCC coefficients are computed for each frame. Therefore, the total number of examples in training dataset is 52947. Similarly, an unknown pattern x is also a 13 dimensional MFCCs feature vector whose label needs to be determined based on the majority vote of the nearest neighbors. The volume V around an unknown pattern x is selected such that the number of nearest neighbors (training examples) within V are 30. We are dealing with the two-class problem with prior probability $P(\omega_i)$. The measurement distribution of the patterns in class ω_i is denoted by $P(x \mid \omega_i)$. The measurement of posteriori class probability $P(\omega_i \mid x)$ decides the label of an unknown feature vector of the separated note. The approximation of $P(x)$ is given by the relation [3], [6]

$$P(x) \simeq \frac{K}{NV} \tag{2}$$

where N is the total number of examples in the dataset, V is the volume surrounding unknown pattern x and K is the number of examples within V. The class prior probability depends on the number of examples in the dataset

$$P(\omega_i) = \frac{N_i}{N} \tag{3}$$

and the mesurement distribution of patterns in class ω_i is defined as

$$P(x \mid \omega_i) = \frac{K_i}{N_i V} \tag{4}$$

According to the Bayes theorem, the posteriori probability becomes

$$P(\omega_i \mid x) = \frac{P(x \mid \omega_i) P(\omega_i)}{P(x)} \tag{5}$$

Based on the above equations, we have [6]

$$P(\omega_i \mid x) = \frac{K_i}{K} \tag{6}$$

The discriminant function $g_i(x) = \frac{K_i}{K}$ assigns the class label to an unknown pattern x based on the majority of examples K_i of class ω_i in volume V.

2.5 Parameter Selection

The most important parameter in the K-NN algorithm is user-defined constant K. The best value of K depends upon the given data for classification [3]. In general, the effect of noise on classification may be reduced by selecting a higher value of K. The problem arises when a large value of K is used for less distinct boundaries between classes [20]. To select good value of K, many heuristic techniques such as cross-validation may be used. In the presence of noisy or irrelevant features the performance of K-NN classifier may be degraded severely [3]. The selection of feature scales according to their importance is another important issue. For the improvement of classification results, a lot of effort has been devoted to the selection or scaling of the features in a best possible way. The optimal classification results are achieved for most datasets by selecting $K = 10$ or more.

2.6 Data Preparation

For the classification of separated components from mixture, the features i.e. the MFCCs, are extracted from all the signals in the training dataset and put the label on all feature vectors according to their classes (piano or violin). The labels of the feature vectors of the separated components are not known which need to be classified. Each feature vector consist of 13 MFCCs. When computing the MFCCs, the training signals and the separated components are all divided into frames with each having a length of 40 ms and 50 percents overlap between the frames is used to avoid discontinuities between the neighboring frames. The similarity measure of the feature vectors of the separated components to the feature vectors obtained from the training process determines which class the separated notes belong to. This is achieved by the K-NN classifier. If majority vote goes to the piano, then a piano label is assigned to the separated component and vice-versa.

2.7 Phase Generation and Source Reconstruction

The factorization of magnitude spectrogram by the NMF algorithm provides frequency-domain basis functions. Therefore, the reconstruction of source signals from the frequency-domain bases is used in this paper, where the phase information is required. Several phase generation methods have been suggested in the literature. When the components do not overlap each other significantly in time and frequency, the phases of the original mixture spectrogram produce good synthesis quality [16]. In the mixture of piano and violin signals, significant overlapping occurs between musical notes in the time domain but the degree of overlapping is relatively low in the frequency domain. Based on this observation, the phases of the original mixture spectrogram are used to reconstruct the source signals in this work. The reconstruction process can be summarised briefly as follows. First, the phase information is added to each classified component to obtain its complex spectrum. Then the classified components from the above sections are combined to the individual source streams, and finally the inverse discrete Fourier Transform (IDFT) and the overlap-and-add technique are applied to obtain the time-domain signal. When the magnitude spectra are used as the basis functions, the frame-wise spectra are obtained as the product of the basis function with its gain. If the power spectra are used, a square root needs to be taken. If the frequency resolution is non-linear, additional processing is required for the re-synthesis using the IDFT.

3 Evaluations

Two music sources (played by two different instruments, i.e. piano and violin) with different number of notes overlapping each other in the time domain, were used to generate artificially an instantaneous mixture signal. The lengths of piano and violin source signals are both 20 seconds, containing 6 and 5 notes respectively. The K-NN classifier constant K was selected as $K = 30$. The signal-to-noise ratio (SNR), defined as follows, was used to measure the quality of both the separated notes and the whole source signal.

$$SNR(m,j) = \frac{\sum_{s,t}[\mathbf{X}_m]^2_{s,t}}{\sum_{s,t}([\mathbf{X}_m]_{s,t} - [\mathbf{X}_j]_{s,t})^2} \tag{7}$$

where s and t are the row and column indices of the matrix respectively. The SNR was computed based on the magnitude spectrograms $\mathbf{X}_m$ and $\mathbf{X}_j$ of the m^{th} reference and the j^{th} separated component to prevent the reconstruction process from affecting the quality [15]. For the same note, $j = m$. In general, higher SNR values represent better separation quality of the separated notes and source signals, vice-versa. The training database used in the classification process was provided by the McGill University Master Samples Collection [10], University of Iowa website [14]. It contains 53 music signals with 29 of which are piano signals and the rest are violin signals. All the signals are sampled at 44100 Hz. The reference source signals were stored for the measurement of separation quality.

Figure 2 shows a separation example of the proposed system, where (a) and (b) are the piano and violin sources respectively, (c) is the single channel mixture of these two sources, and (d) and (e) are the separated sources respectively. From this figure, we can observe that, although most notes are correctly separated and classified into the corresponding sources, there exist notes that were wrongly classified. The separated notes with the highest SNR is the first note of the violin signal, for which the SNR equals to 9.7dB, while the highest SNR of the note within the piano signal is 6.4dB. The average SNRs for piano and violin are respectively 3.7 dB and 1.3 dB. According to our observation, the separation quality of the notes varies from notes to notes. In average, the separation quality of the piano signal is better than the violin signal. The system is still under development and we expect to present more experimental results on this conference. One of the issues that we are trying to address is to improve the classification accuracy by incoporating additional information from the harmonic structure of the music signals.

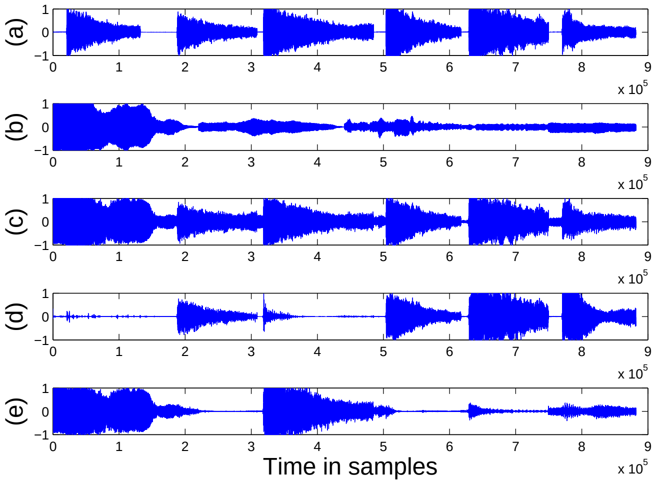

Fig. 2. A separation example of the proposed system. (a) and (b) are the piano and violin sources respectively, (c) is the single channel mixture of these two sources, and (d) and (e) are the separated sources respectively. The vertical axes are the amplitude of the signals.

4 Conclusions

We have presented a new system for the single channel music sound separation problem. The system essentially integrates two techniques, automatic note decomposition using NMF, and note classification based on the K-NN algorithm. A main assumption with the proposed system is that we have the prior knowledge about the type of instruments used for producing the music sounds. Preliminary simulation results show that the system produces a reasonable performance for this challenging source separation problem. We are currently investigating to further improve the note classification accuracy and the overall performance of the separation system.

References

1. Abdallah, S.A., Plumbley, M.D.: Polyphonic Transcription by Non-Negative Sparse Coding of Power Spectra, International Conference on Music Information Retrieval, Barcelona, Spain, October (2004)
2. Barry, D., Lawlor, B., Coyle, E.: Real-time Sound Source Separation: Azimuth Discrimination and Re-synthesis, AES (2004)
3. Devijver, P.A., Kittler, J.: Pattern Recognition - A Statistical Approach, Prentice Hall International (1982)
4. Fevotte, C., Bertin, N., Durrieu, J.-L.: Nonnegative Matrix Factorization With the Itakura-Saito Divergence. With Application to Music Analysis, Neural Computation, 21, 793-830 (2009)
5. Fukunage, K., Introduction to Statistical Pattern Recognition, Academic Press Inc., second edition (1990)
6. Gutierrez-Osuna, R.: Lecture 12: K Nearest Neighbor Classifier [Online]. Available: http://research.cs.tamu.edu/prism/lectures [Accessed 17 January 2010]
7. Hoyer, P.: Non-Negative Sparse Coding, IEEE Workshop on Networks for Signal Processing XII, Martigny, Switzerland (2002)
8. Lee, D.D., Seung, H.S.: Learning the Parts of Objects by Non-Negative Matrix Factorization, Nature, October (1999)
9. Lee, D.D., Seung, H.S.: Algorithms for Non-negative Matrix Factorization, Neural Information Processing Systems, Denver (2001)
10. Opolko, F., Wapnick, J.: McGill University master samples, McGill Univ., Montreal, QC, Canada, Tech. Rep. (1987)
11. Pedersen, M.S., Wang, D.L., Larsen, J., Kjems, U.: Two-Microphone Separation of Speech Mixtures, IEEE Trans. on Neural Networks, 19, 475-492 (2008)
12. Rickard, S., Balan, R., Rosca, J.: Real-time Time-Frequency based Blind Source Separation, In 3rd International Conference on Independent Component Analysis and Blind Source Separation, San Diego, CA, December (2001)
13. Smaragdis, P.: Non-Negative Matrix Factor Deconvolution, Extraction of Multiple Sound Sources From Monophonic Inputs, in Proc. 5th Int. Conf. on Independent Component Analysis and Blind Signal Separation, pp. 494-499, Granada, Spain, Sept. 22-24, Lecture Notes on Computer Science (LNCS 3195) (2004)
14. The University of Iowa Musical Instrument Samples Database [Online].Available: http://theremin.music.uiowa.edu.
15. Virtanen, T.: Sound Source Separation Using Sparse Coding with Temporal Continuity Objective, International Computer Music Conference, Singapore (2003)
16. Virtanen, T.: Separation of Sound Sources by Convolutive Sparse Coding. In Proceedings of ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing, Jeju, Korea (2004)
17. Virtanen, T.: Monaural Sound Source Separation by Non-Negative Matrix Factorization with Temporal Continuity and Sparseness Criteria, IEEE Transactions on Audio, Speech, and Language Processing, 15, 1066-1073 (2007)
18. Wang, W., Luo, Y., Chambers, J.A., Sanei, S.: Note Onset Detection via Nonnegative Factorization of Magnitude Spectrum, EURASIP Journal on Advances in Signal Processing, vol. 2008, Article ID 231367, 15 pages, doi:10.1155/2008/231367, June (2008)
19. Wang, W., Cichocki, A., Chambers, J.A.: A Multiplicative Algorithm for Convolutive Non-negative Matrix Factorization Based on Squared Euclidean Distance, IEEE Transactions on Signal Processing, 57, 2858-2864, July (2009)
20. Webb, A.: Statistical Pattern Recognition. Wiley, New York, second edition (2005)