MUSICAL INSTRUMENT RECOGNITION IN POLYPHONIC AUDIO USING SOURCE-FILTER MODEL FOR SOUND SEPARATION

Toni Heittola, Anssi Klapuri and Tuomas Virtanen
Deparment of Signal Processing, Tampere University of Technology
toni.heittola@tut.fi, klap@cs.tut.fi, tuomas.virtanen@tut.fi

ABSTRACT

This paper proposes a novel approach to musical instrument recognition in polyphonic audio signals by using a source-filter model and an augmented non-negative matrix factorization algorithm for sound separation. The mixture signal is decomposed into a sum of spectral bases modeled as a product of excitations and filters. The excitations are restricted to harmonic spectra and their fundamental frequencies are estimated in advance using a multipitch estimator, whereas the filters are restricted to have smooth frequency responses by modeling them as a sum of elementary functions on the Mel-frequency scale. The pitch and timbre information are used in organizing individual notes into sound sources. In the recognition, Mel-frequency cepstral coefficients are used to represent the coarse shape of the power spectrum of sound sources and Gaussian mixture models are used to model instrument-conditional densities of the extracted features. The method is evaluated with polyphonic signals, randomly generated from 19 instrument classes. The recognition rate for signals having six note polyphony reaches 59%.

1. INTRODUCTION

The majority of research on the automatic recognition of musical instruments until now has been made on isolated notes or on excerpts from solo performances. A comprehensive review of proposed approaches on isolated note based recognition can be found in [1]. In recent years, there has been increasing research interest on more demanding and realistic multi-instrumental polyphonic audio. Most of the proposed techniques extract acoustic features directly from the signal, avoiding the source separation [2, 3].

In polyphonic mixtures consisting of multiple instruments, the interference of simultaneously occurring sounds is likely to limit the recognition performance. The interference can be reduced by first separating the mixture into signals consisting of individual sound sources. In addition to the analysis of mixtures of sounds, sound source separation has applications in audio manipulation and object-based coding.

Many sound source separation algorithms aim at separating the most prominent harmonic sound from the mixture. Usually they first track the pitch of the target sound and then use the harmonic structure and sinusoidal modeling in the separation. A separation system based on this approach has been found to improve the accuracy of a singer identification in background music [4, 5]. Sinusoidal components can also be grouped based on grouping cues such as common onset times, and the recognition can be done using the amplitudes of the grouped sinusoidal partials [6]. Instrument-specific harmonic models trained using instrument-specific material can achieve separation and recognition simultaneously [7].

Recently, many separation algorithms have been proposed which are based on matrix factorization of the mixture spectrogram. The methods approximate the magnitude $x_t(k)$ of the mixture spectrum in frame t and at frequency k as a weighted sum of basis functions as

$$\hat{x}_t(k) = \sum_{m=1}^{M} g_{m,t} b_m(k), \qquad (1)$$

where $g_{m,t}$ is the gain of basis function m in frame t, and $b_m(k)$, $m = 1, \ldots, M$ are the bases. This means that the signal is represented as a sum of components having a fixed spectrum and a time-varying gain. The decomposition can be done, e.g., using independent component analysis (ICA) or non-negative matrix factorization (NMF), the latter usually leading to a better separation quality [8]. The advantage of the methods is their ability to learn the spectral characteristics of each source from a mixture, enabling separation of sources which overlap in time and frequency. Instrument recognition systems based on the decomposition obtained with ICA have extracted the features from the estimated spectral basis vectors [9] or from the reconstructed time-domain signals [10].

A shortcoming of the basic spectrogram decompositions is that each pitch of each instrument has to be represented with a unique basis functions. This requires a large amount of basis functions, making the separation and classification difficult. Virtanen and Klapuri [11] proposed to model each spectral basis vector as a product of an excitation and a filter. The excitation models the time-varying pitch produced by a vibrating element such as a string, which can be shared between instruments, whereas the filter models

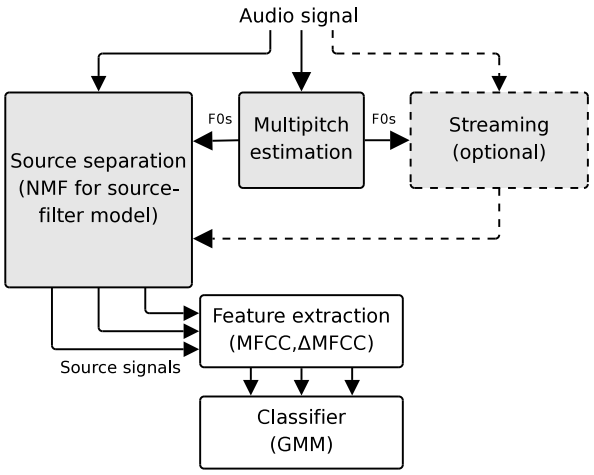

Figure 1. System overview.

the unique resonant structure of each instrument. To improve the performance of the method, FitzGerald proposed to model the excitations of different fundamental frequencies explicitly as a sum of sinusoids at the harmonic frequencies [12]. Badeau et al. [13] used also a harmonic model for the excitation, but they modeled the filter using a moving-average model, resulting in a smooth frequency response. Vincent et al. [14] modeled the spectral basis vectors as a weighted sum of harmonically constrained spectra having a limited frequency support. In this model the weights of each frequency band parametrized the rough spectral shape of the instrument.

In this paper, we present a novel approach to sound separation by using source-filter model in the context of musical instrument recognition. The mixture signal is decomposed into a sum of spectral bases modeled as a product of excitations and filters. The excitations are restricted to harmonic spectra and their fundamental frequencies are estimated in advance using a multiple pitch estimator, whereas the filters are restricted to have smooth frequency responses by modeling them as a sum of elementary functions on Mel-frequency scale. The pitch and timbre information are used in organizing individual notes into sound sources ("streaming"). Separated streams are recognized with a Gaussian mixture model (GMM) classifier. The system is evaluated with randomly mixed polyphonic signals using the Real World Computing (RWC) database [15] and sounds from 19 different instruments.

2. METHOD

An overview of the system is shown in Figure 1. Multipitch estimation is first employed to estimate the pitches in each analysis frame. The estimated pitches are used in the streaming algorithm to form temporally continuous streams of notes. Signals corresponding to individual sources are estimated using NMF for source-filter model. Features are extracted from the signals and they are classified using a GMM classifier. These processing steps are explained in detail in the following.

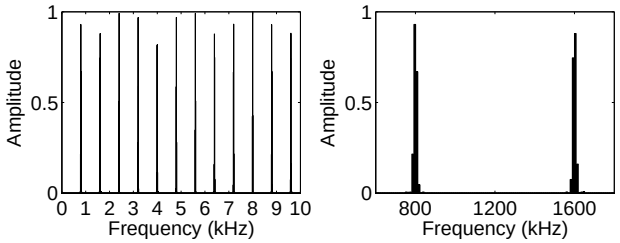

Figure 2. An example excitation spectrum $e_{n,t}(k)$ corresponding to pitch value 800 Hz. The entire spectrum is shown on the left and a closer view of a small portion of it on the right.

2.1 Signal model

In the proposed signal model, each basis $b_m(k)$ in (1) is expressed as a product of an excitation spectrum $e_{n,t}(k)$ and a filter $h_i(k)$. This leads to the model

$$\hat{x}_t(k) = \sum_{n=1}^{N} \sum_{i=1}^{I} g_{n,i,t} e_{n,t}(k) h_i(k) \qquad (2)$$

for the magnitude $x_t(k)$ of the discrete Fourier transform $x_t(k)$ of Hamming-windowed signal in frame t. The excitations $e_{n,t}(k)$ are assumed to correspond to the pitch values of individual notes $n = 1, \ldots, N$ at times $t = 1, \ldots, T$, and the filters $h_i(k)$ are assumed to correspond to the spectral shapes of instruments $i = 1, \ldots, I$. We model only magnitude spectra of the excitations and filters, and therefore they restricted to non-negative real values. All combinations of excitations and filters are allowed, since we do not know in advance which instrument has produced which note. A polyphonic signal consists of several excitation and filter combinations occurring simultaneously or in sequence.

The excitations $e_{n,t}(k)$ are generated based on pitch values obtained from a multipitch estimator. For simplicity, we assume that the number of notes (pitch values) N is the same in all frames t. The multipitch estimator finds the pitches $F_t(n)$, $n = 1, \ldots, N$ in each frame t, and based on these, the corresponding excitation spectra $e_{n,t}(k)$ are calculated which consist of sinusoidal components with unity amplitudes at integer multiples of the corresponding pitch, $F_t(n)$. Figure 2 shows the excitation spectrum corresponding to pitch 800 Hz. Variation in amplitude appears in the figure since the partial frequencies do not fall exactly on spectral bins.

The filter $h_i(k)$ is further represented as a linear combination of fixed elementary responses:

$$h_i(k) = \sum_{j=1}^{J} c_{i,j} a_j(k) \qquad (3)$$

where we chose the elementary responses $a_j(k)$ to consist of triangular bandpass magnitude responses, uniformly distributed on the Mel-frequency scale $f_{\mathrm{Mel}} = 2595 \log_{10}(1 + f_{\mathrm{Hz}}/700)$. The bases are illustrated in Fig. 3.

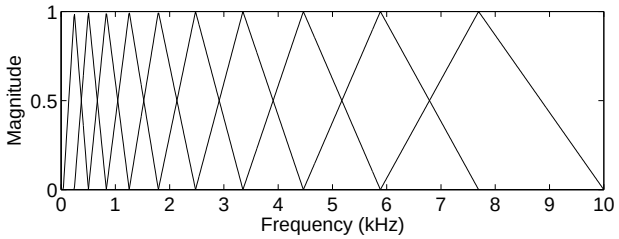

Figure 3. The elementary responses used to represent the filters $h_i(k)$. Triangular responses are uniformly distributed on the Mel-frequency scale.

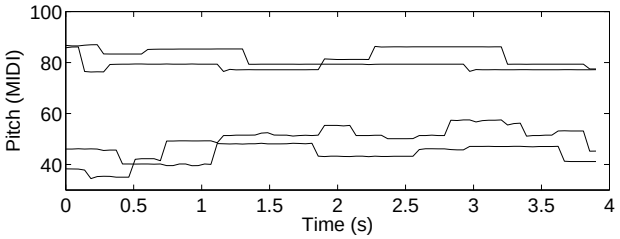

Figure 4. Output of the multipitch estimator for a mixture signal consisting of four simultaneous sounds.

Substituting (3) to (2) gives the final signal model

$$\hat{x}_t(k) = \sum_{n=1}^{N} \sum_{i=1}^{I} g_{n,i,t} e_{n,t}(k) \sum_{j=1}^{J} c_{i,j} a_j(k) \qquad (4)$$

In this model, $e_{n,t}(k)$ are obtained as described above, $a_j(k)$ are fixed in advance, and therefore only $g_{n,i,t}$ and $c_{i,j}$ remain to be estimated using the proposed augmented NMF algorithm. The coefficients $c_{i,j}$ determine the spectral shape (filter) of instrument i, $i = 1, \ldots, I$, and the gains $g_{n,i,t}$ determine the amount of contribution from instrument i to note n at time t. Note that all instruments are allowed to play the same note simultaneously. Further constraints to associate each excitation with only one filter (instrument) are described in Sec. 2.3.

The amount of parameters to estimate is much smaller in the proposed model (4) than in the traditional model (1). This is because the traditional model practically requires a separate basis spectrum for each pitch/instrument combination. In the proposed model, the different notes coming from instrument i are represented by a single basis function (filter) $h_i(k)$, multiplied by the excitation spectra $e_{n,t}(k)$ to produce different pitch values. The smaller amount of parameters improves the reliability of the estimation. Furthermore, in the traditional model, the bases $b_m(k)$ have to be clustered to their respective sources after the estimation, whereas in the proposed model this takes place automatically.

2.2 Estimating the excitation spectra $e_{n,t}(k)$

The multipitch estimator proposed by Klapuri in [16] is used to estimate the note pitches $F_t(n)$, $n = 1, \ldots, N$ in each analysis frame t. Figure 4 illustrates the output of the multipitch estimator for a polyphonic signal consisting of four simultaneous sounds. Based on the pitch value $F_t(n)$, the corresponding excitation $e_{n,t}(k)$ is constructed which consists of Hamming-windowed sinusoidal components at integer multiples of the pitch value $F_t(n)$. These "harmonic combs" extend over the entire frequency range considered and have a unity magnitude for all the harmonics. An example excitation spectrum is shown in Figure 2.

2.3 Streaming algorithm to link excitations with filters

In the described model, all combinations of excitations and filters are allowed. In other words, all instruments (filters) i can play all the detected notes (excitations) n simultaneously. In a realistic situation, however, it is more likely that each note is played by one instrument and only occasionally two or more instruments play the same note.

Parameter $g_{n,i,t}$ controls the salience of each excitation and filter combination in each frame. Robustness of the parameter estimation can be improved if the excitations $e_{n,t}(k)$ can be tentatively organized into "streams", where a stream consists of the successive notes (excitations) coming from a same instrument (filter). Here stream i corresponds to the instrument i and we assume that the number of simultaneous notes N is equal to the number of filters I. The output of the streaming is a label sequence $\ell_t(n)$, where $\ell_t(n) = i$ indicates that the note (excitation) n at time t comes from instrument i. Even though the stream formation algorithm described here is imperfect, it is very helpful in initializing the augmented NMF algorithm that will be described in Sec. 2.4.

Let us introduce a state variable q_t that corresponds to a certain stream-labelling of the excitations $e_{n,t}(k)$, $n = 1, \ldots, N$ at time t. The number of different labellings (and states) is equal to $I!$, that is, the number of different permutations of numbers $1, \ldots, I$. For convenience, the different permutations of numbers $1, \ldots, I$ are stored as columns in a matrix $[\mathbf{U}]_{n,q}$ of size $(N \times I!)$.

A candidate solution to the streaming problem can be represented as a sequence of states $\mathbf{Q} = (q_1 q_2 \ldots q_T)$. The goodness of a candidate state sequence $\mathbf{Q}$ is defined so that it is proportional to the cumulative frame-to-frame variation of acoustic features extracted within each stream. Two types of acoustic feature vectors $\mathbf{z}_t(n)$ were investigated: pitch ($\mathbf{z}_t(n) \equiv F_t(n)$) and Mel-frequency cepstral coefficients (MFCCs) calculated from a spectrum that was constructed by picking only the spectral components corresponding to the harmonic partials of excitation n from the mixture spectrum $x_t(k)$. More exactly, the goodness Γ of a candidate solution $\mathbf{Q}$ given the features $\mathbf{z}_t(n)$ is defined by

$$\Gamma(\mathbf{Q}|\left\{\mathbf{z}_t\left(n\right)\right\}_{1:T,1:N}) = \gamma(q_1) \prod_{t=2}^{T} \gamma(q_t|q_{t-1}) \qquad (5)$$

where the frame-to-frame feature similarity is calculated

by

$$\gamma(q_t|q_{t-1}) = -\sum_{n=1}^{N} \left\| \mathbf{z}_t([\mathbf{U}]_{n,q_t}) - \mathbf{z}_{t-1}([\mathbf{U}]_{n,q_{t-1}}) \right\|. \quad (6)$$

The above goodness measure basically assumes that F0s of consecutive sounds coming from the same instrument usually have only small variations, or using MFCC features, that the spectral shapes of consecutive sounds from a same instrument are similar. Initial goodness values $\gamma(q_1)$ are defined to be zero for all other states except for $q_1 = 1$ for which we set $\gamma(q_1 = 1) = 1$. This removes the ambiguity related to the ordering of different streams.

The most likely sequence $\mathbf{Q} = (q_1 q_2 \dots q_T)$ given the observed features $\mathbf{z}_t\,(n)$ is a search problem

$$\hat{\mathbf{Q}} = \arg\max_{\mathbf{Q}} \Gamma(\mathbf{Q}|\,\{\mathbf{z}_t\,(n)\}_{1:T,1:N}) \quad (7)$$

which can be straightforwardly solved using the Viterbi algorithm. The output of the streaming (associating each excitation with only one filter) is not fixed rigidly, but is used in initializing the NMF parameter estimation algorithm, as will be described below.

2.4 NMF algorithm for parameter estimation

The spectra $h_i(k)$ can be viewed as the magnitude responses of the filters, and therefore it is natural to restrict them to be entrywise non-negative. This is achieved using non-negative coefficients $c_{i,j}$. Furthermore, the model can be restricted to be purely additive by limiting the gains $g_{n,i,t}$ to be non-negative. NMF estimates the bases and their gains by minimizing the reconstruction error between the observed magnitude spectrogram $x_t(k)$ and the model $\hat{x}_t(k)$ while restricting the parameters to non-negative values.

Commonly used measures for the reconstruction error are the Euclidean distance, and divergence d, defined as

$$d(x, \hat{x}) = \sum_{k,t} x_t(k) \log \frac{x_t(k)}{\hat{x}_t(k)} - x_t(k) + \hat{x}_t(k) \quad (8)$$

The divergence is always non-negative, and zero only when $x_t(k) = \hat{x}_t(k)$ for all k and t. An algorithm that minimizes the divergence for the traditional signal model (1) has been proposed by Lee and Seung [17]. In their algorithm, the parameters are initialized to random non-negative values, and updated by applying multiplicative update rules iteratively. Each update decreases the value of the divergence.

We propose an augmented NMF algorithm for estimating the parameters of the model (4). Multiplicative updates which minimize the divergence (8) are given by

$$c_{i,j} \leftarrow c_{i,j} \frac{\sum_{n,t,k} r_t(k) g_{n,i,t} e_{n,t}(k) a_j(k)}{\sum_{n,t,k} g_{n,i,t} e_{n,t}(k) a_j(k)} \quad (9)$$

$$g_{n,i,t} \leftarrow g_{n,i,t} \frac{\sum_{j,k} r_t(k) e_{n,t}(k) c_{i,j} a_j(k)}{\sum_{j,k} e_{n,t}(k) c_{i,j} a_j(k)} \quad (10)$$

where $r_t(k) = \frac{x_t(k)}{\hat{x}_t(k)}$ is evaluated using (4) before each update. The overall estimation algorithm is given as follows:

1. Estimate excitations $e_{n,t}(k)$ using multipitch estimator and the procedure explained in Sec.2.2. Initialize the gains $g_{n,i,t}$ and the filter coefficients $c_{i,j}$ with absolute values of Gaussian noise.
2. Update the filter coefficients $c_{i,j}$ using (9).
3. Update the gains $g_{n,i,t}$ using (10)
4. Repeat steps 2-3 until the changes in parameters are sufficiently small.

In our experiments we observed that the divergence (8) is non-increasing under each update. If streaming is used, initial $g_{n,i,t}$ are multiplied with small factor 0.001 for n that do not belong to stream i. Using a small non-zero value favours the streamed excitations to be associated with the filter i, but this does not exclude the possibility of that the NMF algorithm will "correct" the streaming when the gains are updated during the algorithm. The streaming based on F0 values or MFCC values is far from perfect but yet improves the robustness of the parameter estimation with the NMF algorithm.

2.5 Reconstruction of instrument-wise spectrograms

Spectrograms corresponding to a certain instrument i can be reconstructed by using (4) and limiting the sum over i to one value only:

$$\hat{x}_{i,t}(k) = \sum_{n=1}^{N} g_{n,i,t} e_{n,t}(k) \sum_{j=1}^{J} c_{i,j} a_j(k) \quad (11)$$

Spectrogram of instrument i is reconstructed as

$$y_{i,t}(k) = \frac{\hat{x}_{i,t}(k)}{\hat{x}_t(k)} x_t(k) \quad (12)$$

where the denominator is calculated using (4) and summing over all i.

Time-domain signals are generated by using phases of the mixture signal and inverse discrete Fourier transform.

2.6 Classification

Mel-frequency cepstral coefficients (MFCC) are used to represent the coarse shape of the power spectrum of the separated instrument-wise signals. MFCCs are calculated from the outputs of a 40-channel filterbank which occupies the band from 30Hz to half the sampling rate. In addition to the static coefficients, their first time derivatives approximated with a three-point first-order polynomial fit are used to describe the dynamic properties of the cepstrum.

Gaussian mixture models are used to model instrument-conditional densities of the features. The parameters for the GMM are estimated with Expectation Maximization (EM) algorithm from the training material. Amount of Gaussian distributions in the mixture model was fixed to 32 for each class. In order to prevent acoustic mismatch between the training material and the testing material, models are trained with the separated signals. In the training stage, the perfect streaming is used with a prior knowledge about

the sources in the signals. In the classification stage, likelihoods of the features are accumulated over the signal for the instrument classes, energy weighting the likelihoods with the RMS energy in the corresponding frame, and the classification is performed with maximum-likelihood classifier.

3. EXPERIMENTS

The proposed algorithm is evaluated in the musical instrument recognition task with generated polyphonic signals. The streaming algorithm is evaluated using both the pitch and the timbre information. "no separation" denotes a system where instrument recognition is done without separation directly from the mixture signal. "no streaming" denotes a system where the NMF is initialized (step 1 in Section 2.4) with random values $g_{n,i,t}$. "streaming (given F0s)" denotes system where time-varying pitches of sources were given in advance. "streaming (est. F0s)" denotes system where the pitches were estimated with the multi-pitch estimator and used in automatic streaming. "streaming (timbre)" uses timbre information in streaming and the F0s used for estimating the timbre were given in advance. Prior information of polyphony is used in all systems.

3.1 Acoustic Data

Polyphonic signals are generated by linearly mixing samples of isolated notes from the RWC musical instrument sound database. Nineteen instrument classes are selected for the evaluations (accordion, bassoon, clarinet, contrabass, electric bass, electric guitar, electric piano, flute, guitar, harmonica, horn, oboe, piano piccolo, recorder, saxophone, trombone, trumpet, tuba) and the instrument instances are randomized either into training (70%) or testing (30%) set. The polyphonic signals are generated from these sets, 500 cases for the training and 100 cases for the testing.

Four-second polyphonic signals are generated by randomly selecting instrument instances and generating random note sequences for them. For each instrument, the first note in a note sequence is taken randomly from the uniform distribution specified by the available notes in the RWC database for the instrument instance. The next notes in the sequence are taken from a normal distribution having a previous note as the mean and the standard deviation σ (being 6 semitones if not mentioned otherwise). Unisonal notes are excluded from the note sequence. The notes are randomly truncated to have length between 100 ms and one second. Signals from each instrument are mixed with equal mean-square levels. Examples of test signals are available at `www.cs.tut.fi/~heittolt/ismir09/`.

3.2 Evaluation Results

The separation quality was measured by comparing the separated signals with the reference ones. The signal-to-noise ratio (SNR) of a separated signal is estimated as

	Polyphony				
	2	3	4	5	6
no streaming	4.9	2.6	2.1	1.5	1.2
streaming (est. F0s)	7.2	4.0	2.5	1.7	1.2
streaming (timbre)	7.6	4.4	3.3	2.4	1.9

Table 1. Average signal-to-noise rations (in dB) for different system configurations.

	Polyphony					
	1	2	3	4	5	6
no separation	62.0	18.7	12.1	13.7	24.4	29.5
no streaming	62.0	49.5	42.1	42.6	39.3	42.7
streaming (given F0s)	62.0	59.0	58.0	57.9	57.8	56.0
streaming (est. F0s)	61.0	60.2	53.5	56.7	55.2	53.8
streaming (timbre)	62.0	57.6	51.9	57.0	55.9	59.1

Table 2. F-measures (%) for different system configurations.

$$SNR = 10 \log_{10} \frac{\Sigma_t s(t)^2}{\Sigma_t (s(t) - \hat{s}(t))^2}, \qquad (13)$$

where $s(t)$ is the reference signal and $\hat{s}(t)$is the separated signal. The average signal-to-noise ratios obtained for different system configurations are given in Table 1.

In instrument recognition, balanced F-measure is used as metric in the evaluations. The recall R is calculated as the ratio of correctly recognized instrument labels to sum of the correctly recognized instrument labels and unrecognized instrument labels. The precision P is calculated as the ratio of correctly recognized instrument labels to all instrument labels produced by the system. The F-measure is calculated from these two values as $F = 2RP/(R + P)$.

The evaluation results for different system configurations are given in Table 2. The system without separation uses the prior knowledge about the polyphony of the signal to find same amount of instruments directly from the mixture signal. This increases the random guess rate as the polyphony increases. The proposed approach using separation as a pre-processing gives rather steady performance regardless of the polyphony and gives reasonable performance already without the streaming algorithm. The streaming algorithm improves the results evenly, giving 10-15% increase in performance. The pitch and the timbre information based streaming gives same level of accuracy, though the pitch information seems to give slightly more robust performance. The estimated fundamental frequencies work almost as well as the given frequencies. The evaluation results for different types of polyphonic signals

	Polyphony					
σ	1	2	3	4	5	6
3	51.0	59.9	53.5	57.3	57.6	54.6
6	62.0	57.6	51.9	57.0	55.9	59.1
12	72.0	63.1	53.7	57.5	55.4	57.8

Table 3. F-measures (%) for different polyphonic signal conditions with the timbre based streaming.

are given in Table 3. The proposed system gives quite consistent results with all levels of the polyphony and when varying the standard deviation σ from 3 to 12 semitones. The slight variations in some cases are due to the randomization of used instruments for different polyphony levels.

4. CONCLUSIONS

In this paper, we proposed a source-filter model for sound separation and used it as a preprocessing step for musical instrument recognition in polyphonic music. The experimental results with the generated polyphonic signals were promising. The method gives good results when classifying into 19 instrument classes and with the high polyphony signals, implying a robust separation even with more complex signals. When recognizing the instrument from a sequence of several notes, it seems that the remaining slight separation artefacts average out to quite neutral noise, whereas the information related to the target instrument is consistent and leads to a robust recognition. Even when the F0s are estimated automatically, they provide sufficiently accurate information to get reasonable results. [1]

5. REFERENCES

[1] P. Herrera-Boyer, A. Klapuri, and M. Davy. Automatic classification of pitched musical instrument sounds. In *Signal Processing Methods for Music Transcription*, pages 163–200. Springer, 2006.

[2] T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. Okuno. Musical instrument recognizer "instrogram" and its application to music retrieval based on instrumentation similarity. *International Symposium on Multimedia*, pages 265–274, 2006.

[3] S. Essid, G. Richard, and David.B. Instrument recognition in polyphonic music based on automatic taxonomies. *IEEE Transactions on Audio, Speech & Language Processing*, 14(1):68–80, 2006.

[4] H. Fujihara, T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno. Singer identification based on accompaniment sound reduction and reliable frame selection. In *Proc. ISMIR 2005*, pages 329–336, 2005.

[5] A Mesaros, T. Virtanen, and A. Klapuri. Singer identification and polyphonic music using vocal separation and pattern recognition methods. In *Proc. ISMIR*, pages 375–378, 2007.

[1] This work was financially supported by the Academy of Finland.

[6] J. J. Burred, A. Röbel, and T. Sikora. Polyphonic musical instrument recognition based on a dynamic model of the spectral envelope. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 173–176, 2009.

[7] P. Leveau, E. Vincent, G. Richard, and L. Daudet. Instrument-specific harmonic atoms for mid-level music representation. *IEEE Transactions on Audio, Speech & Language Processing*, 16(1):116–128, 2008.

[8] T. Virtanen. Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE Trans. Audio, Speech and Language Processing*, 15(3):1066–1074, 2007.

[9] P. Jincahitra. Polyphonic instrument identification using independent subspace analysis. In *Proceeding of the IEEE International Conference on Multimedia and Expo*, pages 1211–1214. IEEE, 2004.

[10] P. S. Lampropoulou, A. Lampropoulos, and G. Tsihrintzis. Musical instrument category discrimination using wavelet-based source separation. In *New Directions in Intelligent Interactive Multimedia*, pages 127–136. 2008.

[11] T. Virtanen and A. Klapuri. Analysis of polyphonic audio using source-filter model and non-negative matrix factorization. In *Advances in Models for Acoustic Processing, Neural Information Processing Systems Workshop*, 2006.

[12] D. FitzGerald, M. Cranitch, and E. Coyle. Extended nonnegative tensor factorisation models for musical source separation. *Computational Intelligence and Neuroscience*, 2008.

[13] R. Badeau, V. Emiya, and B. David. Expectation-maximization algorithm for multi-pitch estimation and separation of overlapping harmonic spectra. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009.

[14] E. Vincent, N. Bertin, and R. Badeau. Harmonic and inharmonic nonnegative matrix factorization for polyphonic pitch transcription. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 109–112, 2008.

[15] M. Goto, T. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Music genre database and musical instrument sound database. In *Proc. ISMIR*, pages 229–230, 2003.

[16] A. Klapuri. Multiple fundamental frequency estimation by summing harmonic amplitudes. In *Proc. ISMIR*, pages 216–221, 2006.

[17] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, October 1999.