DRUM TRANSCRIPTION WITH NON-NEGATIVE SPECTROGRAM FACTORISATION

Jouni Paulus, Tuomas Virtanen

Institute of Signal Processing, Tampere University of Technology
Korkeakoulunkatu 1, FI-33720, Tampere, Finland
phone: + (358) 3 3115 4790, fax: + (358) 3115 3857, email: jouni.paulus@tut.fi, tuomas.virtanen@tut.fi
web: http://www.cs.tut.fi/~paulus/research.html

ABSTRACT

This paper describes a novel method for the automatic transcription of drum sequences. The method is based on separating the target drum sounds from the input signal using non-negative matrix factorisation, and on detecting sound onsets from the separated signals. The separation algorithm factorises the spectrogram of the input signal into a sum of instrument spectrograms, each having a fixed spectrum and a time-varying gain. The spectra are calculated from a set of training signals, and the time-varying gains are estimated with an algorithm stemming from non-negative matrix factorisation. Onset times of the instruments are detected from the estimated time-varying gains. The system gave better results than two state-of-the-art methods in simulations with acoustic signals containing polyphonic drum sequences, and overall hit rate of 96% was accomplished. Demonstrational signals are available at http://www.cs.tut.fi/~paulus/demo/.

1. INTRODUCTION

Automatic music transcription and music information retrieval have recently become more popular as the needed computational power has become available. In general, automatic music transcription can be divided into separate tasks of transcribing the tonal parts and the percussive parts (drums). This paper will concentrate on the drum transcription task, which can be defined as the task of estimating the temporal locations of percussive sound events and recognising the instruments which have been used to produce them. In many systems, like in the one proposed here, the task is modified to detecting the temporal locations of pre-defined percussive sound events.

One of the earliest works on automatic drum transcription was by Schloss, whose system transcribed percussion-only music in which only one instrument is present at a time [12]. The system located the sound event onsets based on rapid increases on amplitude envelope. Each located sound event was classified to one of the groups trained for the system based on subband-energy related features.

Another early work was introduced by Goto and Muraoka [5]. In their system, drum transcription was used as an aid in a beat tracking polyphonic music signals. The onset detection in their system was done by locating power increases in frequency domain. Bass drums and snare drums were sought from the located onsets by inspecting peaks in the spectral content of the onsets. This work was continued by Yoshii et al. in [17], where the event recognition was done by matching template spectrograms of individual bass drum and snare drum events to the detected sound onset locations in polyphonic music. The templates were automatically adapted to the target signal, because the drum sounds used in the signal to be analysed may differ from the template sounds.

Traditional pattern recognition approaches have also been utilised in several ways. Herrera et al. made a thorough comparison of different features and classification techniques for analysing individual drum sound events [6]. In drum transcription, these methods generally first locate possible sound onsets using, e.g., the method suggested by Klapuri [8]. Then a set of features is extracted from the signal at the locations of the detected onsets. The detected onsets are labelled using standard pattern recognition techniques, for example, k-nearest neighbours [11], support vector machines (SVM) [4, 14], or Gaussian mixture models [4, 10]. None of these methods seem to perform clearly better than the others, so some advanced techniques and higher-level processing have been developed to increase the performance, such as, language modelling with explicit N-grams [10] or hidden Markov models [4], or choosing best feature subset dynamically [11].

1.1 Separation of drum sounds

Even though individual drum sound events can be recognised quite reliably [6], the recognition from polyphonic music is a difficult task, because of other simultaneously occurring sounds [4]. Separation of sound sources has been used to address the problem, e.g., by using methods based on independent subspace analysis (ISA) [1, 2], and sparse coding [16].

In the case of music signals, ISA and sparse coding have been used to separate the input signal into a sum of sources, each of which has a fixed spectrum and a time-varying gain. This model suits quite well for representing drum signals. The signal model for spectrum $X_t(f)$ in frame t can be written as a weighted sum of source spectra $S_n(f)$:

$$X_t(f) \approx \sum_{n=1}^{N} a_{n,t} S_n(f), \tag{1}$$

where N is the number of sources, n is the source index, $a_{n,t}$ is the gain of the n^{th} source in frame t, and f is the discrete frequency index.

There are several different criteria for estimating $a_{n,t}$ and $S_n(f)$, including the independence of the sources [1], non-negativity [13], or sparseness of the sources [16]. In some systems, the sources are estimated blindly, i.e., there is no prior knowledge of the parameters of the sources. Also, some proposals for the use of pre-trained sources have been made [15].

Prior subspace analysis (PSA) proposed by FitzGerald simplifies the decomposition by initialising the spectral subspaces $S_n(f)$ with values calculated from a large sample set [3]. Then the time-varying gains $a_{n,t}$ are calculated using matrix inverse, passed through independent component analysis (ICA) and finally subjected to onset detection. The main problem with PSA is that $a_{n,t}$ can have also negative values which do not have a reasonable physical counterpart.

Recently, non-negative matrix factorisation (NMF) has been successfully used in several unsupervised learning tasks [9] and also in the analysis of music signals, e.g., by Smaragdis and Brown [13]. In NMF, both the spectra $S_n(f)$ and gains $a_{n,t}$ are restricted to be non-negative. In the case of audio source separation, this can be interpreted so that the spectrograms are purely additive. It has turned out that the non-negativity constraint alone is sufficient for separating sources, to some degree.

1.2 Improvements in the proposed method

The proposed method combines the ideas of PSA and NMF. The spectrogram of the mixture signal is decomposed into spectrograms of target drum instruments using pre-defined fixed spectra $S_n(f)$ and non-negativity constraints in the estimation of the gains $a_{n,t}$.

Natural drum sounds do not have an exactly fixed spectrum over time. When examined with high frequency resolution, the spectrograms exhibit stochastic nature within an individual sound event. In addition, there are differences between the occurrences of a same drum instrument sound events. The variation of the spectrum is reduced by using a coarse frequency resolution, and the signal model of Equation (1) can be used. The spectrum of a drum instrument is approximately fixed on a coarse frequency grid, e.g., bass drums have low-frequency energy and hi-hats have high-frequency energy.

Some publications have discussed the matter of recognising drum patterns from polyphonic music [5, 11, 17]. Before aiming directly to that level, the transcription task in this paper is restricted to material consisting only of a limited number of different drum instruments. Namely, only bass drum, snare drum, and hi-hat occurrences are transcribed.

2. PROPOSED METHOD

The proposed method consists of three stages: at first, source spectra $S_n(f)$ are estimated for each instrument using training material, as will be described in Section 2.1. At the second stage, each drum instrument is separated from the input signal using the trained spectra and the method that will be described in Section 2.2. Finally, the temporal locations of sound events are sought from the separated signals with the method that will be described in Section 2.3.

The magnitude spectrogram is used as a mid-level signal representation. Since drum transcription requires a good temporal resolution, the length of the analysis frame is 24 ms with 75 % overlap between consecutive frames, leading into temporal resolution of 6 ms.

The segregation between different drum classes can be made using a coarse frequency resolution. Only five bands (20-180 Hz, 180-400 Hz, 400-1000 Hz, 1-10 kHz and 10-20 kHz) were used in the simulations. The number and locations of the bands were not specifically optimised for the transcription, but these yielded the best result from the ones tested (e.g., linearly spaced 512 frequency bins or 25 critical bands). The magnitude spectrogram $X_t(f)$ is obtained by using short-time Fourier transform, summing the squared magnitudes within each band to obtain bandwise energies, and by taking the square root.

2.1 Estimation of the source spectra

There are several possibilities for obtaining the instrument spectra $S_n(f)$ from the training data. In our simulations the best results were obtained by using the following procedure. A set of recordings of individual examples of a certain drum instrument n is taken. NMF [9, 13] is used to dismantle the magnitude spectrogram $Y_t^i(f)$ of each example event i into a product of non-negative spectrum $W^i(f)$ and non-negative time-varying gain h_t^i, so that $Y_t^i(f) \approx W^i(f)h_t^i$. The spectral basis vectors $W^i(f)$ are then averaged over i to produce the instrument spectra $S_n(f)$ of drum instrument n. The procedure is repeated for all instruments $n \in [1,N]$.

2.2 Estimation of the time-varying gains

The separation algorithm estimates time-varying gains $a_{n,t}$ for each drum instrument n in each frame t, so that the magnitude spectrum $X_t(f)$ of the input signal is presented as a weighted sum of the fixed spectra $S_n(f)$, as represented in Equation (1). The estimation is done by minimising a cost function between the observed spectrum $X_t(f)$ and the model $M_t(f) = \sum_{n=1}^{N} a_{n,t}S_n(f)$. The gains $a_{n,t}$ are restricted to be non-negative. The method does not make any explicit assumptions of the independence or sparseness of the gains.

The best transcription result was obtained using the divergence proposed by Lee and Seung [9] as the cost function. The divergence D between $X_t(f)$ and $M_t(f)$ is defined as

$$D(X_t(f)||M_t(f)) = \sum_{t,f} d(X_t(f), M_t(f)), \tag{2}$$

where the function d is defined as

$$d(p,q) = p\log(\frac{p}{q}) - p + q. \tag{3}$$

The divergence is minimised by an iterative algorithm, which uses multiplicative updates, given as

$$a_{n,t} \leftarrow a_{n,t} \frac{\sum_f X_t(f)S_n(f)/M_t(f)}{\sum_f S_n(f)}. \tag{4}$$

The iterative estimation algorithm is given by the following procedure:

1. Initialise each $a_{n,t}$ to unity.
2. Set $M_t(f) = \sum_{n=1}^{N} a_{n,t}S_n(f)$.
3. Update each $a_{n,t}$ using the update rule (4)
4. Evaluate cost function 2 and repeat steps 2 to 4 until the value of the cost function converges.

In our experiments with three sources and a five-band spectral representation, the algorithm took approximately 20-30 iterations to converge.

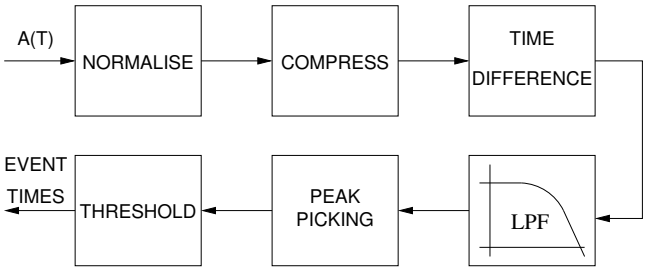

Figure 1: Block diagram of procedure for detecting onsets from an estimated time-varying gain $a_{n,t}$.

2.3 Onset detection

Onset detection of the instrument n is done from the corresponding time-varying gain $a_{n,t}$ with a procedure motivated by the one proposed by Klapuri [8]. Only the time-varying gain is used instead of several sub-band amplitude envelope signals used in the reference. This simplification can be done since the spectrum associated with each gain is fixed, causing all sub-band amplitude envelopes to be of identical form. The algorithm is motivated by the human auditory system, which is sensitive for relative changes in signal level.

The block diagram of the onset detection procedure is illustrated in Figure 1. First, the gain is normalised to range $[0,1]$ to obtain a better control of subsequent steps of the onset detection procedure. The normalised gain $\tilde{a}_{n,t}$ is compressed with $\hat{a}_{n,t} = \log(1 + J\tilde{a}_{n,t})$, where J is a fixed compression factor. The algorithm is not sensitive for the exact value of J; a value of 100 was found to be suitable. The compressed gain is differentiated with $a'_{n,t} = \hat{a}_{n,t} - \hat{a}_{n,t-1}$.

The difference signal $a'_{n,t}$ contains low-amplitude ripple, which is reduced by low-pass filtering. The system is not sensitive to the exact filter characteristics; in our implementation a fourth order Butterworth filter with cut-off frequency 0.25π[1] was used. Finally, the filtered signal is subjected to peak picking. Peaks in the signal represent perceptually salient onsets. Thresholding is used to pick

[1] sampling rate being 167 Hz

only the most prominent peaks. The threshold value can be different for each instrument.

The thresholds needed in the onset detection are estimated automatically from training material with the following procedure. The training signals are separated with the proposed method, and onset are located. By comparing the located onsets to the reference onsets, the threshold value is chosen so that the number of undetected onsets and extraneous detections is minimised. The threshold is calculated for each drum instrument independently.

3. EVALUATION

The performance of the proposed transcription method was evaluated and compared to two other systems using acoustic signals. We used a four-fold cross-validation setup for acoustic material from 4 recording sets, so that three sets were used for training and one set for testing at a time.

3.1 Acoustic material

The simulation database consists of acoustic drum sequences and individual drum samples. Three different drum kits and three different recording locations were used. One of the kits was recorded in two different locations, resulting to total of four recording sets:

1. an entry level kit recorded in a medium sized room,
2. a studio grade kit recorded in a medium sized room,
3. a heavy metal kit recorded in an acoustically damped hall, and
4. an entry level kit recorded in an anechoic chamber.

The acoustic information was recorded using close microphones for bass drums and snare drums, and overhead microphones for hi-hats. Recorded signals were mixed to yield two mix-downs: an unprocessed one, and a "production-grade" processed one where multiband compression, equalisation, and reverberation were used. The reference onsets were acquired by using piezo triggers on bass drums and snare drums. The hi-hats were annotated by hand. The temporal accuracy of the annotated onsets was estimated to be better than 10 ms.

The drum sequences in the evaluation database are fairly simple, consisting only of bass drums, snare drums and hi-hats. Different playing styles are not discriminated, e.g., open, closed and pedal hi-hats are treated as equal. The sequences used were 8-beat, 16-beat, stomp, shuffle and triole, resulting in total of 20 signals. The sequences do not contain only several repetitions of the same pattern, but the players were encouraged to make some variations while playing. Only 15-second excerpts of the sequences were used in the evaluation.

In addition to the sequences, individual drum hits were recorded with 20 repetitions of each. These were used to obtain the spectra $S_n(f)$, as explained in Section 2.1.

3.2 Performance metrics

For each drum instrument, the performance was measured by comparing the transcribed onsets with the reference onsets. A transcribed onset was judged to be correct if it deviated less than 30 ms from a reference onset. The transcribed and reference onsets were matched using the following procedure. At first, the algorithm calculates a $V \times L$ matrix Z of absolute time differences between all transcribed and reference events $Z_{v,l} = |(t_v - t_l)|, v = 1 \ldots V, l = 1 \ldots L$, where V is the number of transcribed events and L is the number of events in reference data. Then the events v and l having the smallest time difference are paired and removed from the distance matrix. This pairing is continued until all remaining time differences are larger than 30 ms or either event set runs out of available events. The b remaining unmatched transcribed events are insertions and the c remaining unmatched reference events are deletions leading to *instrument hit rate* of $R_h = 1-(b+c)/L$. Also, *precision rate* $R_p = (V-b)/V$ and *recall rate* $R_r = (L-c)/L$ were calculated. Precision rate is the ratio of correct detections to all detections, and recall rate is the ratio of correct detections to number of events in the reference annotation. The overall hit rate was calculated as the mean of individual instrument hit rates. The presented performance measures are calculated over the four cross validation iterations.

unprocessed		B	S	H	avg
SVM	R_p %	99	93	92	94
	R_r %	99	93	86	89
	R_h %	98	86	77	**87**
PSA	R_p %	91	77	80	82
	R_r %	95	91	70	78
	R_h %	86	70	46	**67**
NSF	R_p %	100	100	98	99
	R_r%	100	94	96	96
	R_h %	100	93	94	**96**

processed		B	S	H	avg
SVM	R_p %	99	100	95	97
	R_r %	99	93	91	93
	R_h %	98	93	86	**92**
PSA	R_p %	77	83	80	80
	R_r %	92	84	73	78
	R_h %	71	67	51	**63**
NSF	R_p %	98	100	96	97
	R_r %	98	94	96	96
	R_h %	95	94	93	**94**

Table 1: Results for the unprocessed (upper table) and "production-grade" processed (lower table) test signals. *B* denotes bass drums, *S* snare drums, *H* hi-hats, and *avg* the average of *B*, *S* and *H*. *SVM* is the method described in [4], *PSA* the method described in [3], and *NSF* the proposed method.

3.3 Comparisons to other systems

Two other transcription systems were used for comparison with similar evaluation setup. The systems by Gillet et al. [4] and FitzGerald et al. [3] were tested. The method by Gillet et al. initially detects all sound event onsets from the signal, then extracts a set of features from the locations of the detected onsets, and finally uses an SVM classifier for recognising the events. The presence of each drum instrument in the event is detected with a binary classifier, and no sequence modeling is used. The classifiers were trained with the acoustic sequences in the training set. The algorithm implementation was based on the information given in the reference, and the SVM implementation by Joachims was used [7].

In PSA, initially, the spectral basis vectors are calculated from the individual drum hits in the training set. Then, the corresponding time-varying gains are estimated with a matrix inverse and independent component analysis. Finally, the sound onsets are detected from the estimated time-varying gains. The implementation of the original authors was used.

3.4 Results and discussion

The performance evaluation results are presented in Table 1. The proposed method performed better in total than both comparison methods with unprocessed and processed drum signals, though the difference is smaller with processed signals. This is most likely due to the fact that when handling the processed signals, the features used by the SVM method were trained with processed signals, while the spectral models used by the PSA and the proposed method were trained with individual unprocessed hits in both scenarios, because there were no processed individual hits available.

Some preliminary experiments were made to utilise the pro-

posed method also for more complex signals, i.e., signals containing also other drums or melodic instruments. It was noted that the performance of the proposed system degrades if the analysed signal does not fit the model, i.e., other interfering sounds are present in addition to the modelled instruments. Similar performance degradation was noted also with the two reference systems. It is possible that the SVM method may be able to handle more complex input signals, as it's operation does not rely on direct assumptions on the structure of the signal. This problem of method generalisation remains as a subject of further development.

4. CONCLUSIONS

In this paper, we have presented a method for drum transcription. It uses pre-calculated spectra and non-negativity constraints for the gains of the spectra for separating different instruments. The proposed method has been evaluated with simulations and the performance of the presented method has been compared with two reference methods. The proposed method performed better than the two reference methods in simulations with polyphonic drum sequences.

Acknowledgements

The drum database was recorded and annotated by Teemu Karjalainen at TUT.

REFERENCES

[1] M. A. Casey and A. Westner. Separation of mixed audio sources by independent subspace analysis. In *Proc. International Computer Music Conference*, pages 154–161, Berlin, Germany, Aug. 2000.

[2] D. FitzGerald, E. Coyle, and B. Lawlor. Sub-band independent subspace analysis for drum transcription. In *Proc. 5th Internationl Conference on Digital Audio Effects*, pages 65–69, Hamburg, Germany, Sept. 2002.

[3] D. FitzGerald, B. Lawlor, and E. Coyle. Prior subspace analysis for drum transcription. In *Proc. 114th Audio Engineering Society Convention*, Amsterdam, The Netherlands, Mar. 2003.

[4] O. Gillet and G. Richard. Automatic transcription of drum loops. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, May 2004.

[5] M. Goto and Y. Muraoka. A beat tracking system for acoustic signals of music. In *Proc. ACM Multimedia*, pages 365–372, 1994.

[6] P. Herrera, A. Yeterian, and F. Gouyon. Automatic classification of drum sounds: a comparison of feature selection methods and classification techniques. In *Proc. 2nd International Conference on Music and Artificial Intelligence*, pages 69–80, Edinburgh, Scotland, UK, Sept. 2002.

[7] T. Joachims. Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT-Press, 1999.

[8] A. Klapuri. Sound onset detection by applying psychoacoustic knowledge. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, Arizona, USA, 1999.

[9] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In T. Leen, T. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems*, volume 13, pages 556–562. MIT Press, 2001.

[10] J. K. Paulus and A. P. Klapuri. Conventional and periodic N-grams in the transcription of drum sequences. In *Proc. IEEE International Conference on Multimedia and Expo*, volume 2, pages 737–740, Baltimore, Maryland, USA, July 2003.

[11] V. Sandvold, F. Gouyon, and P. Herrera. Percussion classification in polyphonic audio recordings using localized sound models. In *Proc. 5th International Conference on Music Information Retrieval*, Barcelona, Spain, Oct. 2004.

[12] W. A. Schloss. *On the automatic transcription of percussive music – from acoustic signal to high-level analysis*. PhD thesis, Center for Computer Research in Music and Acoustics, Stanford University, Stanford, California, USA, May 1985.

[13] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Proc. 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 177–180, New Platz, New York, USA, Oct. 2003.

[14] D. Van Steelant, K. Tanghe, S. Degroeve, B. De Baets, M. Leman, and J.-P. Martens. Classification of percussive sounds using support vector machines. In *Proc. The Annual Machine Learning Conference of Belgium and The Netherlands*, Brussels, Belgium, Jan. 2004.

[15] E. Vincent and X. Rodet. Music transcription with ISA and HMM. In *Proc. Independent Component Analysis and Blind Signal Separation*, pages 1197–1204, Granada, Spain, Sept. 2004.

[16] T. Virtanen. Sound source separation using sparse coding with temporal continuity objective. In *Proc. International Computer Music Conference*, Singapore, Oct. 2003.

[17] K. Yoshii, M. Goto, and H. G. Okuno. Automatic drum sound description for real-world music using template adaptation and matching methods. In *Proc. 5th International Conference on Music Information Retrieval*, Barcelona, Spain, Oct. 2004.