DRUM TRANSCRIPTION USING SEMI-SUPERVISED NON-NEGATIVE MATRIX FACTORIZATION

First author
Affiliation1
author1@ismir.edu

Second author
Retain these fake authors in submission to preserve the formatting

Third author
Affiliation3
author3@ismir.edu

ABSTRACT

A drum transcription algorithm using semi-supervised non-negative matrix factorization is presented. The proposed method allows users to separate percussive activities from harmonic activities with pre-trained drum templates and detect drum events from the extracted percussive activation matrix. It is efficient and is robust even with a minimal training set. A subset from ENST drum data set has been used for training and testing of the algorithm. The system has been evaluated using different dictionary rank settings. The system can achieve 61 to 77% recognition rate on multiple drums in polyphonic music.

1. INTRODUCTION

Automatic music transcription is one of the most intensively researched areas in Music Information Retrieval (MIR). The reliable extraction of a score (or a score-related representation) from the audio signal will be the core technology of a large number of applications in fields such as music education, systematic musicology, and music visualization. Furthermore, a reliable transcription would enable high-level representations of music signals with the potential of improving virtually any MIR task.

A complete transcription system comprises many related sub-tasks such as multi-pitch detection, onset detection, instrument recognition, and rhythm extraction [2]. While the main focus is mostly on pitched instruments, a considerable amount of publications deal with the transcription of percussive sounds in mixture of tonal and percussive instruments. The drum track in popular music conveys information about tempo, rhythm, style, and possibly the structure of a song. A drum transcription system alone enables applications in active listening [20], music education, and interactive music performance [18].

This study explores the application of the popular transcription method of Non-Negative Matrix Factorization (NMF) for drum transcription in polyphonic music. The paper is structured as follows: Section 2 provides an overview of the research in this area. In Section 3 we present our approach; evaluation results are being presented and discussed in Section 4. Section 5 provides a summary, conclusion, and directions of future work.

2. RELATED WORK

The early attempts to transcribe percussive sounds mainly focused on the classification of signals containing only drum sounds. For these systems, standard approaches with a feature extractor and a subsequent classification engine are able to produce results with high accuracy [10, 11].

For many real-world applications, however, the input file is a mixture of percussive and harmonic sound sources. Therefore, a drum transcription system is expected to work on this mixture of sounds instead of exclusively on percussive sounds. Gillet and Richard divide systems for the drum transcription from mixtures into three categories [8]: (i) *segment and classify*, (ii) *separate and detect*, and (iii) *match and adapt*.

Systems of the first category (segment and classify) usually segment the audio signal into a series of events by applying automatic onset detection and extract various features from time or spectral domain. Each event segment is then classified based on the extracted features. This approach seems to perform well when features are well chosen [3, 6, 17]. However, a sufficient amount of training data and carefully adjusted pre-processing is required in order to get good results. Furthermore, the possibility of simultaneous sounds increases the number of classes significantly.

The second type of approaches (separate and detect) is based on the assumption that the music signal is a superposition of different sound sources. By decomposing the signal and into source templates with corresponding activation functions, the music content could be transcribed by identifying the templates and analysing the activities for each template. Different methods such as Independent Subspace Analysis [4], Prior Subspace Analysis [5], and Non-negative Matrix Factorization (NMF, see below) [1, 13, 14] fall into this category. The advantage of these approaches is that they usually are easier to interpret since most of the decompositions are done on the spectrogram of the signal. Furthermore, the handling of simultaneous and overlapping events is inherent to the approach. However, in the context of NMF with a pre-determined dictionary matrix, whether or not the templates are representative enough is one potential problem. Another difficulty is the determination of the rank required for the decomposition process.

The third type of approaches (match and adapt) uses pre-trained templates to detect drum events [21, 22]. The templates are searched for the closest match and adapted in an iterative process.

3. METHOD

3.1 Algorithm Description

In this paper, we present a method based on NMF. Although NMF has been proven to be effective in music transcription [15], it is still difficult to determine the number of sources in the target signal for a correct rank parameter setting. The goal of our system is the transcription of drum tracks in complex mixtures while only requiring a few drum samples for training prior to the decomposition process.

The basic concept of NMF is to approximate a non-negative matrix V with two non-negative matrices W and H as shown in Eq. (1).

$$V \approx WH \tag{1}$$

Given a $m \times n$ matrix V, NMF will decompose the matrix into the product of a $m \times r$ matrix W and a $r \times n$ matrix H, with r being the rank of the NMF decomposition. W is the dictionary matrix, and H is the activation matrix. In most audio applications, V is the spectrogram to be decomposed, W contains the magnitude spectra of the salient components, and H indicates the activation of these components with respect to time. The matrices W and H are obtained through an iterative process that minimizes a distance measure between the target spectrogram V and its approximation. [12].

The application of this method to music transcription, however, offers some challenges. First, the number of sound sources and notes within a music recording is usually unknown. It is therefore difficult to estimate a correct rank number and obtain a clear representation of the decomposed components. Second, it is hard to identify the corresponding instrument of every component in the dictionary matrix W, especially when the rank is very high or very low. Third, when multiple similar entries exist in the dictionary matrix, the corresponding activation matrix could be activated at these entries simultaneously making it harder to interpret the results. Different methods have been proposed in the previous studies to address these issues. Virtanen trained an SVM to separate drum components from the harmonic components; the rank number was derived empirically during the factorization process [9]. The identified drum components and their corresponding activities could later be used to reconstruct the drum signal, resulting in a system for drum source separation. Virtanen's approach requires a significant amount of training data for the classifier and, more importantly, the results can be expected to be very susceptible to choice of rank. Yoo et al. proposed a co-factorization algorithm [19] to simultaneously factorize a prior drum track and a target signal, and use the basis matrix from the drum track to identify the drum components in the target signal. This method ensures that the drum components in both dictionary matrices remain percussion only over the iterations, and thus proper isolation of the harmonic components from the drum components. Since they focus on drum separation rather than drum transcription, they can work at very high ranks, but the approach is not directly applicable to the transcription problem because of the probable lack of interpretability of the dictionary matrix.

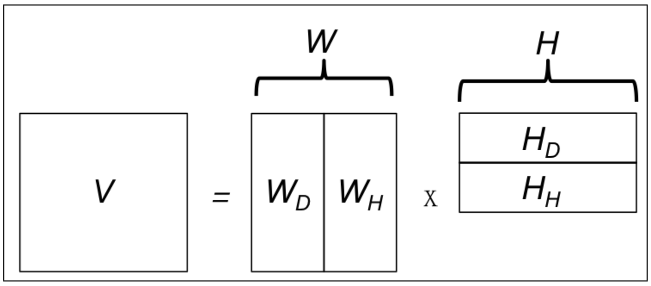

Figure 1. Illustration of the factorization process. The pre-trained drum basis matrix W_D will not be updated over the iterations.

Nevertheless, their work inspired our approach to drum transcription. Figure 1 visualizes the basic concept from the work of Yoo et al.: the matrices W and H are split into the matrices W_D and W_H, and H_D and H_H, respectively. Instead of using co-factorization, however, we propose to initialize the matrix W_D some drum templates and to not modify it during the factorization process. Matrices W_H, H_H, and H_D are initialized with random numbers. The cost function as shown in Eq. (2) is minimized by applying gradient decent and multiplicative update rules, the matrices W_H, H_H, and H_D will be updated according to Eqs. (3)–(5).

$$C = \frac{1}{2}||V - W_D H_D - W_H H_H||^2 \tag{2}$$

$$H_D \leftarrow H_D \frac{W_D^T V}{W_D^T W_D H_D + W_D W_H H_H} \tag{3}$$

$$W_H \leftarrow W_H \frac{V H_H^T}{W_H H_H H_H^T + W_D H_D H_H^T} \tag{4}$$

$$H_H \leftarrow H_H \frac{W_H^T V}{W_H^T W_H H_H + W_H W_D H_D} \tag{5}$$

To summarize, the method described above consists of the following steps:

1. Construct a $m \times r_D$ dictionary matrix W_D, r_D is the number of drum components to be detected.

2. Given a pre-defined rank r_H, initialize a $m \times r_H$ matrix W_H, a $r_D \times n$ matrix H_D and a $r_H \times n$ matrix H_H.

3. Normalize W_D and W_H.

4. Update H_D, W_H, and H_H using Eqs. (3)–(5).

5. Calculate the cost of current iteration using Eq. (2).

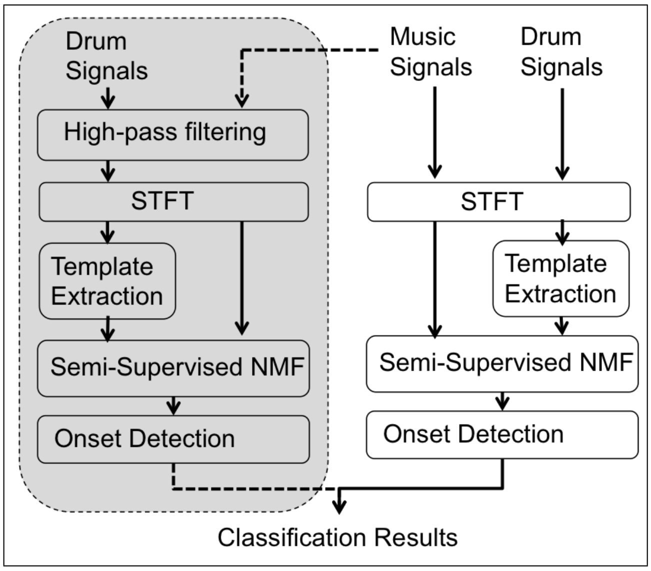

Figure 2. Flowchart of the drum transcription system

6. Repeat step 3 to step 5 until convergence.

The time positions of the drum events can then be extracted with peak-picking the matrix H_D.

3.2 Implementation

Figure 2 shows the flow chart of the implemented system. The basic setup is shown on the right hand side. The STFT of the signals will be calculated using a window size and a hop size of 2048 and 512, respectively. A pre-trained dictionary matrix will be constructed from the training set, consisting of isolated drum sounds (see Section 3.2.1). Next, the semi-supervised NMF will be performed with rank r as described above. Finally, the activation Matrix H_D is evaluated to determine the onset positions and their corresponding classes (see Section 3.2.1).

The gray area in Figure 2 is an optional addition to the system. Preliminary tests showed improved Hi-Hat (HH) detection accuracy when applying a high-pass filter to both the template and the music signal. We used a 2nd order Butterworth high-pass filter with a heuristically determined cutoff frequency of 8000 Hz.

The following subsections provide more details on the template extraction and activity detection.

3.2.1 Template Extraction

As mentioned above, the dictionary matrix W_D is created by extracting a template spectrum from isolated training drum samples. The template magnitude spectrum is median spectrum of all individual events of one drum class in the training set. The length of each event is approximately 80 ms. The templates are extracted for the three classes Hi-Hat (HH), Bass Drum (BD) and Snare Drum (SD).

3.2.2 Activity Detection

High activity values in the activation matrix H_D indicate the presence of a drum event. More specifically, the activity difference of each row of the dictionary matrix could be considered as the onset novelty function of each individual drum. We use a median filter as a standard approach to create an adaptive threshold for pick peaking result.

4. EVALUATION

4.1 Dataset Description

The experiments have been conducted on the *minus one* subset from the ENST public drum data set [7]. This data set consists of recordings from three different drummers performing on their own drum kits. The set for each drummer contains individual hits, short phrases of drum beats, drum solos, and short excerpts played with the accompaniments. The minus one subset has 64 tracks of polyphonic music. Each track in this subset has a length of approximately 70 s with varying style. More specifically, the subset features many drum playing techniques such as ghost notes, flam, and drag; these techniques are considered difficult to identify with existing drum transcription systems. Since we are only interested in the three classes HH, BD, and SD, tracks missing one of these instruments or featuring specific playing techniques have been discarded, leaving a subset of 53 out of 64 tracks.

The accompaniments are mixed with the drum tracks in the data set without any modification (e.g., no level adjustment). The distribution of onset counts per class per drummer is shown in Table 1.

The drums template (see Section 3.2.1) have been generated from the tracks with single hits. Each track contains 5 to 6 single hits on different drums. The onset position of these single hits were determined using the annotated ground truth.

	Drum. 1	**Drum. 2**	**Drum. 3**	**Total**
HH	1942	2145	1813	5900
BD	2140	1488	1378	5006
SD	2165	2079	1994	6238
Total	6247	5712	5185	17144

Table 1. Onset counts in selected data set

4.2 Evaluation Procedure

We evaluate two different combinations of training and test data.

First, we use training samples from all three drummers to train the drum basis matrix, and test the system on all 53 tracks. In this test run, we also evaluate the results in case of filtered training and testing signals.

Second, we investigate cross-performer validation as shown in Figure 3. The training samples are selected only from one drummer, and the test samples will be only the other drummers' recordings. This scenario should be similar to a real-world use case for which the trained drum sounds are not necessarily similar to the drum sounds in the target signals.

The evaluation metrics follow by the standard calculation of the precision (P), recall (R), and F-measure (F). An onset

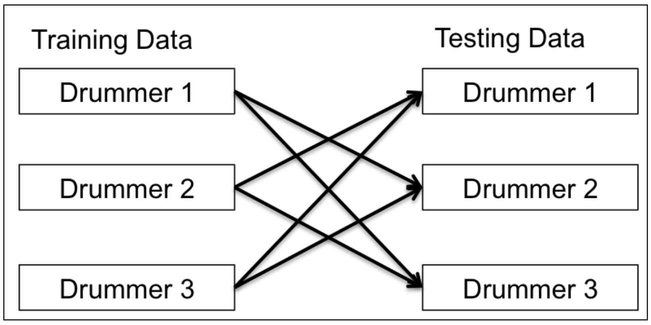

Figure 3. Cross-performer validation process.

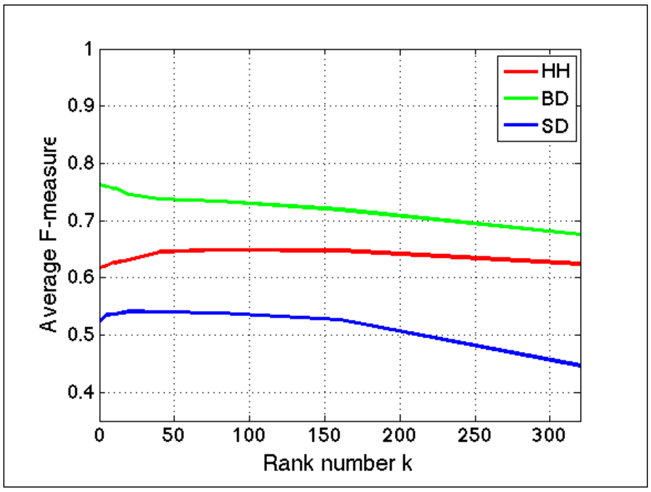

Figure 4. Average F-measure versus NMF rank

is considered to be a match with the ground truth if the time deviation between the annotated and detected onset is less or equal to 50 ms.

4.3 Evaluation Results

4.3.1 Pre-Evaluation Test

In an initial test to determine the rank r_H of the algorithm, $r_H = 5, 10, 20, 40, 80, 160, 320$ have been tested. The resulting individual F-measures are shown in Figure 4. A general trend of decreasing performance with increasing r_H can be observed, especially for lower frequency sounds such as SD and BD. Based on this observation, a rank number $r_H = 10$ is chosen in current setup.

4.3.2 Evaluation I

Table 2 shows the results (precision, recall, and F-measure) when using the complete training and test set. The results for the standard setup (the white section of Figure 2) are given in the column named *Standard*. The middle column (*HPF+LPF+BPF*) presents the results for signals preprocessed with filters. More specifically, we use three filters corresponding to the three instruments we intend to detect; the filter frequencies are derived from the median magnitude spectra:

- HH: $f_{hp} = 8000\,\text{Hz}$
- SD: $f_{bp} = 200 - -500\,\text{Hz}$
- BD: $f_{lp} = 150\,\text{Hz}$

The last column uses a similar setup, but only one high-pass filter for the HH class.

		Standard	HPF+LPF+BPF	HPF only
HH	P	0.605	0.671	0.675
	R	0.713	0.737	0.736
	F	0.654	0.702	**0.704**
BD	P	0.723	0.662	0.740
	R	0.837	0.855	0.839
	F	0.776	0.746	**0.786**
SD	P	0.678	0.591	0.676
	R	0.566	0.583	0.569
	F	0.617	0.587	**0.618**

Table 2. Transcription results using all training templates

It can be observed that using the filters does not necessarily improve the results. The combination of three filters in particular seems to improve results slightly for HH, but the drop in precision of the other classes results in a lower F-measure. In the third, high-pass-only, case we can identify a general trend for improved results in all classes, although the increase for SD and especially BD is rather negligible. These small increase might due the random initialization of the harmonic dictionary, but the deviations are not significant. Comparing with the reported F-measure of 77.7%, 65.0% and 64.8% for HH, BD and SD in [8], our results show better performance on BD, but slightly worse performance in SD and HH. However, in our method, the training process require less amount of training data to achieve the results at same level, which could potential provide a better use case for a more generic drum transcription system.

The results in the middle column of Table 2 indicate that by filtering the templates and signals with constant cutoff frequencies, the performance of BD and SD drop slightly. Some possible reasons might contribute to this result. First, although the filtering process could suppress contents in the unwanted frequency bins, it also could remove many information that might help to differentiate instruments such as BD and Bass Guitar. Second, in current parameter settings, the band-pass and low-pass filtered signals might only have few bins to represent the spectrum, which might increase the difficulty for the semi-NMF to adapt. Third, when the signals are filtered, the current rank setting r_H might not be optimal for the task.

4.3.3 Evaluation II

In order to investigate the dependency of our approach with respect to the similarity of training and test drum sounds, we conduct a cross-performer evaluation as shown in Figure 3. No filters have been applied during the process. Their results are listed in Table 3.

The results show a simple trend: the test set containing drummer 2 give nearly always the best results, regardless of the training set. Also, when training with different drummer's recordings, the F-measure from HH and SD are mostly within the same range as the results reported in

Training		Dr1	Dr2	Dr3	Avg.
Testing		Dr2+Dr3	Dr1+Dr3	Dr1+Dr2	
HH	P	0.593	0.601	0.561	0.585
	R	0.721	0.683	0.696	0.700
	F	**0.651**	0.639	0.621	0.637
BD	P	0.816	0.583	0.831	0.743
	R	0.907	0.728	0.948	0.861
	F	0.859	0.647	**0.886**	0.797
SD	P	0.694	0.577	0.590	0.620
	R	0.547	0.518	0.566	0.544
	F	**0.612**	0.546	0.578	0.578

Table 3. Transcription results of cross-performer validation.

Section 4.3.2 except for BD. This could be due to the fact that Bass Drum of drummer 2 is easy to detect. These results indicate that this algorithm is relatively robust against differences between the drum template and the sound of the drum to be detected. This would allow to construct a template from different sound sources independent of the recording to be analyzed allowing far more general applications. However, the performance of this setup still needs to be confirmed with a cross-data set validation.

5. CONCLUSION

We have presented a drum transcription system for polyphonic music using semi-supervised NMF. This method uses a pre-trained dictionary matrix to decompose the target signal and extract the activation matrix. The evaluation results show that this method is able to achieve 61 to 77% accuracy for detecting 3 classes in complex mixtures of music. The presented method has the following advantages: First, the fixed dictionary matrix in the model makes it easier to interpret the corresponding activation matrix for transcription tasks. Second, simultaneous sounds can be detected separately without the need of training extra classes. Third, adjustment of the parameter r_H allows the algorithm to adapt to different different types of polyphonic music. Fourth, cross-performer evaluation results indicate a robustness against template mismatches, possibly allowing the application in situations with minimum prior knowledge. Fifth, the approach requires only trivial extensions to be able to be used as a drum separation system as an extension to the current transcription system. Last but not least, the approach only requires a few drum samples to train the dictionary matrix, and the evaluation results indicate that the performance is comparable with state-of-the art methods at lower algorithmic complexity.

Possible directions for future work are: a comparison between this approach with Probabilistic Latent Component Analysis (PLCA) [16]. Looking for means to slightly adapt the template during the decomposition might be a way to further generalize the current method. More generally speaking, we plan to evaluate other distance metrics, cost functions, and adaptation rules in the future. Furthermore, we intend to investigate robust methods to automatically estimate the rank r_H. To achieve a complete drum transcription system in polyphonic music, however, more factors such as playing techniques and more drum classes still need to be addressed in the future.

6. REFERENCES

[1] David S Alves, Jouni Paulus, and José Fonseca. Drum transcription from multichannel recordings with nonnegative matrix factorization. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Glasgow, 2009.

[2] Emmanouil Benetos, Simon Dixon, Dimitrios Giannoulis, Holger Kirchhoff, and Anssi Klapuri. Automatic music transcription: Challenges and future directions. *Journal of Intelligent Information Systems*, 41(3):407—434, December 2013.

[3] Christian Dittmar. Drum detection from polyphonic audio via detailed analysis of the time frequency domain. In *Proceedings of the Music Information Retrieval Evaluation eXchange (MIREX)*, 2005.

[4] Derry FitzGerald and Bob Lawlor. Sub-band independent subspace analysis for drum transcription. In *Proceedings of the International Conference on Digital Audio Effects (DAFX)*, Hamburg, 2002.

[5] Derry FitzGerald, Bob Lawlor, and Eugene Coyle. Drum transcription in the presence of pitched instruments using prior subspace analysis. In *Proceedings of the Irish Signals & Systems Conference (ISSC)*, Limerick, 2003.

[6] Olivier Gillet and Gaël Richard. Automatic transcription of drum loops. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages iv–269–iv–272 vol.4, May 2004. 00062.

[7] Olivier Gillet and Gaël Richard. ENST-Drums: an extensive audio-visual database for drum signals processing. In *ISMIR*, Victoria, 2006.

[8] Olivier Gillet and Gaël Richard. Transcription and separation of drum signals from polyphonic music. *Transactions on Audio, Speech, and Language Processing*, 16(3):529—540, March 2008.

[9] Marko Helen and Tuomas Virtanen. Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Talya, 2005.

[10] Perfecto Herrera, Amaury Dehamel, and Fabien Gouyon. Automatic labeling of unpitched percussion sounds. In *Proceedings of the 114th Audio Engineering Society Convention*. AES, March 2003.

[11] Perfecto Herrera, Alexandre Yeterian, and Fabien Gouyon. Automatic classification of drum sounds: a comparison of feature selection methods and classification techniques. In *Proceedings of the 2nd International Conference on Music and Artificial Intelligence (ICMAI)*, 2002.

[12] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*, 2000.

[13] Arnaud Moreau and Arthur Flexer. Drum transcription in polyphonic music using non-negative matrix factorisation. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 353—354, 2007.

[14] Jouni Paulus and Tuomas Virtanen. Drum transcription with non-negative spectrogram factorisation. In *Proceedings of the 13th European Signal Processing Conference (EUSIPCO)*, page 4, Talya, 2005.

[15] Paris Smaragdis and Judith C Brown. Non-negative matrix factorization for polyphonic music transcription. In *Proceedings of the Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA))*, New Paltz, 2003. IEEE.

[16] Paris Smaragdis, Cedric Fevotte, Gautham J. Mysore, Nasser Mohammadiha, and Matthew Hoffman. Static and Dynamic Source Separation Using Nonnegative Factorizations: A unified view. *IEEE Signal Processing Magazine*, 31(3):66–75, May 2014.

[17] Koen Tanghe, Sven Degroeve, and Bernard De Baets. An algorithm for detecting and labeling drum events in polyphonic music. In *Proceedings of the Music Information Retrieval Evaluation eXchange (MIREX)*, 2005.

[18] Gil Weinberg, Aparna Raman, and Trishul Mallikarjuna. Interactive jamming with shimon: a social robotic musician. In *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, pages 233—234. ACM, 2009.

[19] Jiho Yoo, Minje Kim, Kyeongok Kang, and Seungjin Choi. Nonnegative matrix partial co-factorization for drum source separation. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pages 1942—1945, Dallas, 2010. IEEE.

[20] Kazuyoshi Yoshii, Masataka Goto, and Kazunori Komatani. Drumix: An audio player with real-time drum-part rearrangement functions for active music listening. *IPSJ Digital Courier*, 3:134—144, 2007.

[21] Kazuyoshi Yoshii, Masataka Goto, and Hiroshi G. Okuno. Automatic drum sound description for real-world music using template adaptation and matching methods. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Barcelona, 2004.

[22] Kazuyoshi Yoshii, Masataka Goto, and Hiroshi G Okuno. Drum sound recognition for polyphonic audio signals by adaptation and matching of spectrogram templates with harmonic structure suppression. *Transactions on Audio, Speech and Language Processing*, 15(1):333—345, January 2007.