INSTRUMENT EQUALIZER FOR QUERY-BY-EXAMPLE RETRIEVAL: IMPROVING SOUND SOURCE SEPARATION BASED ON INTEGRATED HARMONIC AND INHARMONIC MODELS

Katsutoshi Itoyama†* **Masataka Goto**‡ **Kazunori Komatani**†
Tetsuya Ogata† **Hiroshi G. Okuno**†

† Graduate School of Infomatics, Kyoto University * JSPS Research Fellow (DC1)
‡ National Institute of Advanced Industrial Science and Technology (AIST)
{itoyama,komatani,ogata,okuno}@kuis.kyoto-u.ac.jp m.goto@aist.go.jp

ABSTRACT

This paper describes a music remixing interface, called *Instrument Equalizer*, that allows users to control the volume of each instrument part within existing audio recordings in real time. Although query-by-example retrieval systems need a user to prepare favorite examples (songs) in general, our interface gives a user to generate examples from existing ones by cutting or boosting some instrument/vocal parts, resulting in a variety of retrieved results. To change the volume, all instrument parts are separated from the input sound mixture using the corresponding standard MIDI file. For the separation, we used an integrated tone (timbre) model consisting of harmonic and inharmonic models that are initialized with template sounds recorded from a MIDI sound generator. The remaining but critical problem here is to deal with various performance styles and instrument bodies that are not given in the template sounds. To solve this problem, we train probabilistic distributions of timbre features by using various sounds. By adding a new constraint of maximizing the likelihood of timbre features extracted from each tone model, we succeeded in estimating model parameters that better express actual timbre.

1 INTRODUCTION

One of promising approaches of music information retrieval is the query-by-example (QBE) retrieval [1, 2, 3, 4, 5, 6, 7] where a user can receive the list of musical pieces ranked by their similarity to a musical piece (example) that the user gives as a query. Although this approach is powerful and useful, a user has to prepare or find favorite examples and sometimes feels difficulty to control/change the retrieved pieces after seeing them because the user has to find another appropriate example to get better results. For example, if a user feels that vocal or drum sounds are too strong in the retrieved pieces, the user has to find another piece that has weaker vocal or drum sounds while keeping the basic mood and timbre of the piece. It is sometimes very difficult to find such a piece within a music collection.

We therefore propose yet another way of preparing an example for the QBE retrieval by using a music remixing interface. The interface enables a user to boost or cut the volume of each instrument part of an existing musical piece. With this interface, a user can easily give an alternative query with a different mixing balance to obtain refined results of the QBE retrieval. The issue in the above example of finding another piece with weaker vocal or drum sounds can thus be resolved. Note that existing graphic equalizers or tone controls on the market cannot control each individual instrument part in this way: they can adjust only frequency characteristics (e.g., boost or cut for bass and treble). Although remixing stereo audio signals [8] had reported previously, it had tackled to control only harmonic instrument sounds. Our goal is to control all instrument sounds including both harmonic and inharmonic ones.

This paper describes our music remixing interface, called *Instrument Equalizer*, in which a user can listen to and remix a musical piece in real time. It has sliders corresponding to different musical instruments and enables a user to manipulate the volume of each instrument part in polyphonic audio signals. Since this interface is independent of the succeeding QBE system, any QBE system can be used. In our current implementation, it leverages the standard MIDI file (SMF) corresponding to the audio signal of a musical piece to separate sound sources. We can assume that it is relatively easy to obtain such SMFs from the web, etc. (especially for classical music). Of course, given a SMF, it is quite easy to control the volume of instrument parts during the SMF playback, and readers might think that we can use it as a query. Its sound quality, however, is not good in general and users would lose their drive to use the QBE retrieval. Moreover, we believe it is important to start from an existing favorite musical piece of high quality and then refine the retrieved results.

2 INSTRUMENT EQUALIZER

The Instrument Equalizer enables a user to remix existing polyphonic musical signals. The screenshot of its interface is shown in Figure 1 and the overall system is shown in Figure 2. It has two features for remixing audio mixtures as follows:

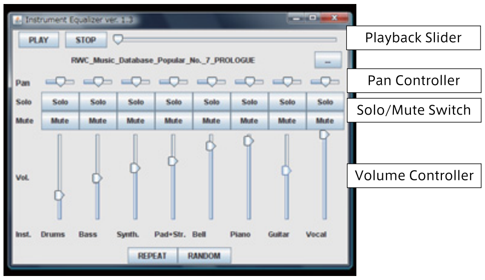

Figure 1. Screenshot of main window.

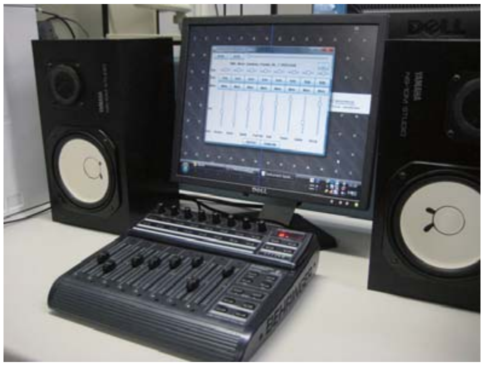

Figure 2. Instrument Equalizing System.

1. *Volume control function*. It provides the remixing function by boosting or cutting the volume of each instrument part, not by controlling the gain of a frequency band. A user can listen to the remixed sound mixture as soon as the user manipulates the volume.
2. *Interlocking with the hardware controller*. In addition to a typical mouse control on the screen, we allow a user to use a hardware controller shown in Figure 2 with multiple faders. It enables the user to manipulate the volume intuitively and quickly. This hardware controller makes it easy to manipulate the volume of multiple parts at the same time, while it is difficult on a mouse control.

To remix a polyphonic musical signal, the signal must be separated into each instrument part. We use an integrated weighted mixture model consisting harmonic-structure and inharmonic-structure tone models [9] for separating the signal, but improve the parameter estimation method of this model by introducing better prior distributions. This separation method needs a standard MIDI file (SMF) that is synchronized to the polyphonic signal. We assume that the SMF has already been synchronized with the input signal by using audio-to-score alignment methods such as [10, 11, 12]. For the separation using the integrated model, the parameters of the model are initialized by template sounds recorded from a MIDI sound generator. and gradually improved to represent actual sounds in the sound mixture.

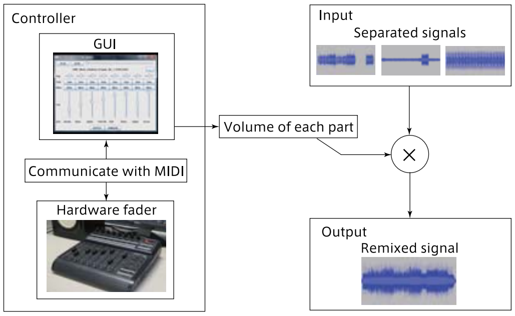

Figure 3. System architecture.

2.1 Internal architectures

This section describes the internal architectures of controlling the volume of each instrument part. The procedures described in this section are performed in real time under the assumption that the musical signals of each instrument part already have been obtained in advance from the target polyphonic musical signal, as described in Section 3. Let $x_k(c,t)$ and $y_k(c,t)$ be a separated signal and the volume of instrument k at channel c and time t, respectively. $y_k(c,t)$ satisfies the following condition:

$$\forall k, c, t: \ 0 \leq y_k(c,t) \leq 1,$$

and $y_k(c,t)$ is obtained as

$$y_k(c,t) = (\text{value of volume slider } k) \cdot (\text{value of the pan } c).$$

The overview of the architecture is shown in Figure 3.

1. *Volume control function*. The output signal, $x(c,t)$, is obtained as
$$x(c,t) = \sum_k y_k(c,t) \cdot x_k(c,t).$$
Each $y_k(t)$ is obtained in real-time from the volume sliders in the GUI in Figure 1.
2. *Interlocking with the hardware controller*. The GUI and the hardware controller communicate by MIDI. If users control the hardware fader, a MIDI message which represents the new volume is sent to the GUI, and vice varsa. Since a motor is embeded in the fader, MIDI messages from the GUI move the fader to the position corresponding value of the volume.

3 SOUND SOURCE SEPARATION CONSIDERING TIMBRE VARIETIES

In this section, we first define our sound source separation problem and the integrated model. We then describe timbre varieties and timbre feature distributions for estimating parameters of the model.

3.1 Integrated model of harmonic and inharmonic models

The sound source separation problem is to decompose the input power spectrogram, $X(c,t,f)$, into the power spectrogram corresponding to each musical note, where c, t, and f are the channel (e.g., left and right), the time, and the frequency, respectively. We assume that $X(c,t,f)$ includes K musical instruments and the k-th instrument performs L_k musical notes. We use the tone model, $J_{kl}(c,t,f)$, to represent the power spectrogram of the l-th musical note performed by the k-th musical instrument ((k,l)*-th note*), and the power spectrogram of a template sound, $Y_{kl}(t,f)$, to initialize the parameters of $J_{kl}(c,t,f)$. Each musical note of the SMF is played back on a MIDI sound generator to record the corresponding template sound. $Y_{kl}(t,f)$ is monaural because SMFs may not include any sound localization (channel) information. $Y_{kl}(t,f)$ is normalized to satisfy the following relation, where C is the total number of the channels:

$$\sum_c \iint X(c,t,f)\,dt\,df = C \sum_{k,l} \iint Y_{kl}(t,f)\,dt\,df.$$

For this source separation, we define an integrated model, $J_{kl}(c,t,f)$, as the sum of harmonic-structure tone models, $H_{kl}(t,f)$, and inharmonic-structure tone models, $I_{kl}(t,f)$, multiplied by the whole amplitude of the model, $w_{kl}^{(J)}$, and the relative amplitude of each channel, $r_{kl}(c)$:

$$J_{kl}(c,t,f) = w_{kl}^{(J)} r_{kl}(c)\big(H_{kl}(t,f) + I_{kl}(t,f)\big),$$

where $w_{kl}^{(J)}$ and $r_{kl}(c)$ satisfy the following constraints:

$$\sum_{k,l} w_{kl}^{(J)} = \iint X(c,t,f)\,dt\,df,\ \forall k,l : \sum_c r_{kl}(c) = C.$$

All parameters of $J_{kl}(c,t,f)$ are listed in Table 1. The harmonic model, $H_{kl}(t,f)$, is defined as a constrained two-dimensional Gaussian mixture model (GMM), which is a product of two one-dimensional GMMs, $\sum E_{kl}^{(H)}(m,t)$ and $\sum F_{kl}^{(H)}(n,t,f)$, and is designed by referring to the harmonic-temporal-structured clustering (HTC) source model [13]. The inharmonic model, $I_{kl}(t,f)$, is defined as a product of two nonparametric functions. The definition of these models is as follows:

$$H_{kl}(t,f) = w_{kl}^{(H)} \sum_{m=1}^{M} \sum_{n=1}^{N} E_{kl}^{(H)}(m,t) F_{kl}^{(H)}(n,t,f),$$

$$E_{kl}^{(H)}(m,t) = \frac{u_{kl}(m)}{\sqrt{2\pi}\phi_{kl}} \exp\left(-\frac{(t-\tau_{kl}-m\phi_{kl})^2}{2\phi_{kl}^2}\right),$$

$$F_{kl}^{(H)}(n,t,f) = \frac{v_{kl}(n)}{\sqrt{2\pi}\sigma_{kl}} \exp\left(-\frac{(f-n\omega_{kl}(t))^2}{2\sigma_{kl}^2}\right), \text{and}$$

$$I_{kl}(t,f) = w_{kl}^{(I)} E_{kl}^{(I)}(t) F_{kl}^{(I)}(t,f),$$

Table 1. Parameters of the integrated model.

Symbol	Description
$w_{kl}^{(J)}$	overall amplitude
$r_{kl}(c)$	relative amplitude of each channel
$w_{kl}^{(H)}, w_{kl}^{(I)}$	relative amplitude of harmonic and inharmonic tone models
$u_{kl}(m)$	coefficient of the temporal power envelope
$v_{kl}(n)$	relative amplitude of n-th harmonic component
τ_{kl}	onset time
ϕ_{kl}	diffusion of a Gaussian of power envelope
$\omega_{kl}(t)$	F0 trajectory
σ_{kl}	diffusion of a harmonic component along the freq. axis
$E_{kl}^{(I)}(t)$	power envelope of inharmonic tone model
$F_{kl}^{(I)}(t,f)$	relative amplitude of frequency f at time t of inharmonic tone model

where M is the number of Gaussian kernels representing the temporal power envelope and N is the number of Gaussian kernels representing the harmonic components. $u_{kl}(m)$, $v_{kl}(n)$, $E_{kl}^{(I)}(t)$, $F_{kl}^{(I)}(t,f)$, $w_{kl}^{(H)}$, and $w_{kl}^{(I)}$ satisfy the following conditions:

$$\forall k,l : \sum_m u_{kl}(m) = 1, \quad \forall k,l : \sum_n v_{kl}(n) = 1,$$

$$\forall k,l : \int E_{kl}^{(I)}(t)\,dt = 1, \quad \forall k,l,t : \int F_{kl}^{(I)}(t,f)\,df = 1,$$

$$\text{and} \quad \forall k,l : w_{kl}^{(H)} + w_{kl}^{(I)} = 1.$$

The goal of this separation is to decompose $X(c,t,f)$ into $J_{kl}(c,t,f)$ by estimating a spectrogram distribution function, $\Delta^{(J)}(k,l;c,t,f)$, which satisfies

$$\forall k,l,c,t,f : 0 \le \Delta^{(J)}(k,l;c,t,f) \le 1 \quad \text{and}$$

$$\forall c,t,f : \sum_{k,l} \Delta^{(J)}(k,l;c,t,f) = 1.$$

With $\Delta^{(J)}(k,l;c,t,f)$, the separated power spectrogram, $X_{kl}^{(J)}(c,t,f)$, is obtained as

$$X_{kl}^{(J)}(c,t,f) = \Delta^{(J)}(k,l;c,t,f) X(c,t,f).$$

Furthermore, let $\Delta^{(H)}(m,n;k,l,t,f)$ and $\Delta^{(I)}(k,l,t,f)$ be spectrogram distribution functions which decompose $X_{kl}^{(J)}(c,t,f)$ into each Gaussian distribution of the harmonic model and the inharmonic model, respectively. These functions satisfy

$$\forall k,l,m,n,t,f : 0 \le \Delta^{(H)}(m,n;k,l,t,f) \le 1,$$

$$\forall k,l,t,f : 0 \le \Delta^{(I)}(k,l,t,f) \le 1, \quad \text{and}$$

$$\forall k,l,t,f : \sum_{m,n} \Delta^{(H)}(m,n;k,l,t,f) + \Delta^{(I)}(k,l,t,f) = 1.$$

To evaluate the 'effectiveness' of this separation, we can use a cost function defined as the Kullback-Leibler (KL) divergence from $X_{kl}^{(J)}(c,t,f)$ to $J_{kl}(c,t,f)$:

$$Q_{kl}^{(J)} = \sum_c \iint X_{kl}^{(J)}(c,t,f) \log \frac{X_{kl}^{(J)}(c,t,f)}{J_{kl}(c,t,f)}\, dt\, df.$$

By minimizing the sum of $Q_{kl}^{(J)}$ over (k,l) pertaining to $\Delta^{(J)}(k,l;c,t,f)$, we obtain the spectrogram distribution function and model parameters (i.e., the most 'effective' decomposition).

By minimizing the $Q_{kl}^{(J)}$ pertaining to each parameter of the integrated model, we obtain model parameters estimated from the distributed spectrogram. This parameter estimation is equivalent to a maximum likelihood estimation. The parameter update equations are described in appendix A.

3.2 Timbre varieties within each instrument

Even within the same instrument, different instrument bodies have different timbres, although its timbral difference is smaller than the difference among different musical instruments. Moreover, in live performances, each musical note could have slightly different timbre according to the performance styles. Instead of preparing a set of many template sounds to represent such timbre varieties within each instrument, we represent them by using a probabilistic distribution.

We use parameters of the integrated model, $u_{kl}(m)$, $v_{kl}(n)$, and $F_{kl}^{(I)}(t,f)$, to represent the timbre variety of instrument k by training a diagonal Gaussian distribution with mean $\mu_k^{(u)}(m)$, $\mu_k^{(v)}(m)$, $\mu_k^{(F)}(f)$ and variance $\Sigma_k^{(u)}(m)$, $\Sigma_k^{(v)}(n)$, $\Sigma_k^{(F)}(f)$, respectively. Note that other probability distributions, such as a Dirichlet distribution, are available in this case. The model parameters for training the prior distribution are extracted from instrument sound database [14] (i.e., the parameters are estimated without any prior distributions).

By minimizing the cost function,

$$\begin{aligned} Q_{kl}^{(p)} =& \sum_c \iint X_{kl}^{(J)}(c,t,f) \log \frac{X_{kl}^{(J)}(c,t,f)}{J_{kl}(c,t,f)}\, dt\, df \\ &+ \frac{1}{2} \sum_m (u_{kl}(m) - \mu_k^{(u)}(m))^2 / \Sigma_k^{(u)}(m) \\ &+ \frac{1}{2} \sum_n (v_{kl}(n) - \mu_k^{(v)}(n))^2 / \Sigma_k^{(v)}(n) \\ &+ \frac{1}{2} \iint (F_{kl}^{(I)}(t,f) - \mu_k^{(F)}(f))^2 / \Sigma_k^{(F)}(f)\, dt\, df, \end{aligned}$$

where the last term is an additional cost by using the prior distribution, we obtain the parameters by taking into account the timbre varieties. This parameter estimation is equivalent to a maximum *A Posteriori* estimation.

3.3 Cost function without considering timbre feature distributions

In Itoyama's previous study [9], they used template sounds instead of timbre feature distributions to evaluate the 'goodness' of the feature vector. The cost function, $Q_{kl}^{(Y)}$, used in [9] can be obtained by replacing the negative log-likelihood, $Q_{kl}^{(p)}$, with the KL divergence, $Q_{kl}^{(Y)}$, from $Y_{kl}(t,f)$ (the power spectrogram of a template sound) to $J'_{kl}(t,f)$:

$$\begin{aligned} Q_{kl}^{(Y)} =& \sum_c \iint X_{kl}^{(J)}(c,t,f) \log \frac{X_{kl}^{(J)}(c,t,f)}{J_{kl}(c,t,f)}\, dt\, df \\ &+ \sum_c \iint r_{kl}(c) Y_{kl}(t,f) \log \frac{r_{kl}(c) Y_{kl}(t,f)}{J_{kl}(c,t,f)}\, dt\, df. \end{aligned}$$

4 EXPERIMENTAL EVALUATION

We conducted experiments to confirm whether the performance of the source separation using the prior distribution is equivalent to the one using the template sounds. In the first experiment, we separated the sound mixtures which were generated from a MIDI sound generator. In the other one, the sound mixtures were created from the signals with multiple tracks [15] which were before mixdown. In this experiment, we compared the following two conditions:

1. using the log-likelihood of timbre feature distributions (proposed method, section 3.2),
2. using the template sounds (previous method [9], section 3.3).

4.1 Experimental conditions

We used 5 SMFs from the RWC Music Database (RWC-MDB-P-2001 No. 1, 2, 3, 8, and 10) [16]. We recorded all musical notes of these SMFs by using two different MIDI sound generators made by different manufacturers. We used one of them for the test (evaluation) data and the other for obtaining the template sounds or training the timbre feature distributions in advance.

The experimental procedure is as follows:

1. initialize the integrated model of each musical note by using the corresponding template sound,
2. estimate all the model parameters from the input sound mixture, and
3. calculate the SNR in the frequency domain for the evaluation.

The SNR is defined as follows:

$$\mathrm{SNR} = \frac{1}{C(T_1 - T_0)} \sum_c \int \mathrm{SNR}_{kl}(c,t)\, dt,$$

$$\mathrm{SNR}_{kl}(c,t) = \log_{10} \int \frac{X_{kl}^{(J)}(c,t,f)^2}{\left(X_{kl}^{(J)}(c,t,f) - X_{kl}^{(R)}(c,t,f)\right)^2}\, df,$$

Table 2. Experimental conditions.

Frequency analysis	Sampling rate	44.1 kHz
	Analyzing method	STFT
	STFT window	2048 points Gaussian
	STFT shift	441 points
Parameters	C	2
	M	10
	N	30
MIDI sound generator	Test data	YAMAHA MU-2000
	Template sounds	Roland SD-90

Table 3. SNRs of the signals separated from sound mixtures generated from a MIDI tone generator. P1, P2, and P3 are the SNRs which are based on the prior distribution trained using 1, 2, and 3 instrument bodies, respectively. T is the SNR which is based on the template sounds.

	P1	P2	P3	T
P001	11.5	12.1	11.6	14.0
P002	12.3	12.3	12.5	12.3
P003	11.5	11.8	12.4	10.8
P008	8.1	7.8	8.3	4.9
P010	9.1	8.8	8.9	12.2
Ave.	10.5	10.6	10.8	10.8

Table 4. SNRs of the signals separated from sound mixtures generated from CD recordings.

	P1	P2	P3	T
P001	9.3	9.4	9.4	9.0
P002	11.4	11.5	11.7	11.6
P003	4.0	4.1	4.2	3.1
P008	7.1	7.2	7.3	6.1
P010	8.4	8.5	8.5	8.0
Ave.	8.1	8.2	8.3	7.6

where T_0 and T_1 are the beginning and ending times of the input power spectrogram, $X(c,t,f)$, and $X_{kl}^{(R)}(c,t,f)$ is the ground-truth power spectrogram corresponding to the (k,l)-th note (i.e., the spectrogram of an actual sound before mixing). We used a 40-parameters for the prior distributions (1–10 dimensions: $u_{kl}(1),\ldots,u_{kl}(M)$, 11–40 dimensions: $v_{kl}(1),\ \ldots,\ v_{kl}(N)$), where M = 10 and N = 30. Other experimental conditions are shown in Table 2.

4.2 Experimental results

The results are listed in Tables 3 and 4. In both experiments, the SNRs were improved by increasing the number of instrument bodies for training the prior distributions. Furthermore, the average SNR of P3 is equal to that of T in Table 3 and the average SNRs in Table 4 is improved from the average SNR of T. This means that although the SNRs decrease in some cases where the timbre difference between template sounds and input sounds is large, it can be resolved by using the better prior distributions.

5 CONCLUSION

In this paper, we have proposed the novel use of a music remixing interface for generating queries for the QBE retrieval, explained our Instrument Equalizer on the basis of sound source separation using an integrated model consisting of harmonic and inharmonic models, and described a new parameter estimation method for the integrated model by using the timbre feature distributions. We confirmed that this method increased separation performance for most instrument parts simply by using the basic timbre features.

Although the use of the timbre feature distributions is promising, it has not been fully exploited in our experiments. For example, we have not tried to use training data including various performance styles and instrument bodies. We plan to evaluate our method by using such various training data as well as more advanced timbre features. Some performance benchmark for audio source separation [17] will helpful to compare our separation method with other ones. Future work will also include the usability evaluation of the Instrument Equalizer for the use of the QBE retrieval.

6 ACKNOWLEDGEMENTS

This research was partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research of Priority Areas, Primordial Knowledge Model Core of Global COE program and CrestMuse Project.

7 REFERENCES

[1] Rauber, A., Pampalk, E., Merkl, D., "Using Psycho-acoustic Models and Self-organizing Maps to Create a Hierarchical Structuring of Music by Sound Similarity", *Proc. ISMIR*, pp. 71–80, 2002.

[2] Yang, C., "The MACSIS Acoustic Indexing Framework for Music Retrieval: An Experimental Study", *Proc. ISMIR*, pp. 53–62, 2002.

[3] Allamanche, E., Herre, J., Hellmuth, O., Kastner, T., Ertel, C., "A Multiple Feature Model for Musical Similarity Retrieval", *Proc. ISMIR*, 2003.

[4] Feng, Y., Zhuang, Y., Pan, Y., "Music Information Retrieval by Detecting Mood via Computational Media Aesthetics", *Proc. of Web Intelligence*, pp. 235–241, 2003.

[5] Thoshkahna, B. and Ramakrishnan, K. R., "Projekt Quebex: A Query by Example System for Audio Retrieval", *Proc. ICME*, pp. 265–268, 2005.

[6] Vignoli, F. and Pauws, S., "A Music Retrieval System Based on User-driven Similarity and Its Evaluation", *Proc. ISMIR*, pp. 272–279, 2005.

[7] Kitahara, T., Goto, M., Komatani, K., Ogata, T., Okuno, H. G., "Musical Instrument Recognizer "Instrogram" and Its Application to Music Retrieval based on Instrumentation Similarity", *Proc. ISM*, pp. 265–274, 2006.

[8] Woodruff, J., Pardo, B., and Dannenberg, R., “Remixing Stereo Music with Score-informed Source Separation”, *Proc. ISMIR*, pp. 314–319, 2006.

[9] Itoyama, K., Goto, M., Komatani, K., Ogata, T., Okuno, H.G., “Integration and Adaptation of Harmonic and Inharmonic Models for Separating Polyphonic Musical Signals”, *Proc. ICASSP*, pp. 57–60, 2006.

[10] Cano, P., Loscos, A., and Bonada, J., “Score-performance Matching using HMMs”, *Proc. ICMC*, pp. 441–444, 1999.

[11] Adams, N., Marquez, D., and Wakefield, G., “Iterative Deepening for Melody Alignment and Retrieval”, *Proc. ISMIR*, pp. 199–206, 2005.

[12] Cont, A., “Realtime Audio to Score Alignment for Polyphonic Music Instruments using Sparce Non-negative Constraints and Hierarchical HMMs”, *Proc. ICASSP*, pp. 641–644, 2006.

[13] Kameoka, H., Nishimoto, T., Sagayama, S., “Harmonic-temporal Structured Clustering via Deterministic Annealing EM Algorithm for Audio Feature Extraction”, *Proc. ISMIR*, pp. 115–122, 2005.

[14] Goto, M., Hashiguchi, H., Nishimura, T., and Oka, R., “RWC Music Database: Music Genre Database and Musical Instrument Sound Database”, *Proc. ISMIR*, pp. 229-230, 2003.

[15] Goto, M., “AIST Annotation for the RWC Music Database”, *Proc. ISMIR*, pp. 359–260, 2006.

[16] Goto, M., Hashiguchi, H., Nishimura, T., and Oka, R., “RWC Music Database: Popular, Classical, and Jazz Music Databases”, *Proc. ISMIR*, pp. 287–288, 2002.

[17] Vincent, E., Gribonbal, R., Févotte, C., “Performance Measurement in Blind Audio Source Separation”, *IEEE Trans. on ASLP*, vol. 14, No. 4, pp. 1462–1469, 2006.

A DERIVATION OF THE PARAMETER UPDATE EQUATION

In this section, we describe the update equations of each parameter derived from the M-step of the EM algorithm. By differentiating the cost function about each parameter, the update equations were obtained. Let $X^{(H)}_{klmn}(c,t,f)$ and $X^{(I)}_{kl}(c,t,f)$ be the decomposed power:

$$X^{(H)}_{klmn}(c,t,f) = \Delta^{(H)}(m,n;k,l,t,f)X^{(J)}_{kl}(c,t,f)$$
$$\text{and}\quad X^{(I)}_{kl}(c,t,f) = \Delta^{(I)}(k,l,t,f)X^{(J)}_{kl}(c,t,f).$$

A.1 $w^{(J)}_{kl}$: overall amplitude

$$w^{(J)}_{kl} = \sum_c \iint \left(\sum_{m,n} X^{(H)}_{klmn}(c,t,f) + X^{(I)}_{kl}(c,t,f)\right) dt\,df.$$

A.2 $r_{kl}(c)$: relative amplitude of each channel

$$r_{kl}(c) = \frac{C\iint \left(\sum_{m,n} X^{(H)}_{klmn}(c,t,f) + X^{(I)}_{kl}(c,t,f)\right) dt\,df}{\sum_c \iint \left(\sum_{m,n} X^{(H)}_{klmn}(c,t,f) + X^{(I)}_{kl}(c,t,f)\right) dt\,df}.$$

A.3 $w^{(H)}_{kl}$, $w^{(I)}_{kl}$: amplitude of harmonic and inharmonic tone models

$$w^{(H)}_{kl} = \frac{\sum_{c,m,n} \iint X^{(H)}_{klmn}(c,t,f)\,dt\,df}{\sum_c \iint \left(\sum_{m,n} X^{(H)}_{klmn}(c,t,f) + X^{(I)}_{kl}(c,t,f)\right) dt\,df}$$

and

$$w^{(I)}_{kl} = \frac{\sum_c \iint X^{(I)}_{kl}(c,t,f)\,dt\,df}{\sum_c \iint \left(\sum_{m,n} X^{(H)}_{klmn}(c,t,f) + X^{(I)}_{kl}(c,t,f)\right) dt\,df}.$$

A.4 $u_{kl}(m)$: coefficient of the temporal power envelope

$$u_{kl}(m) = \frac{\sum_{c,n} \iint X^{(H)}_{klmn}(c,t,f)\,dt\,df + \mu^{(u)}_k(m)}{\sum_{c,m,n} \iint X^{(H)}_{klmn}(c,t,f)\,dt\,df + 1}.$$

A.5 $v_{kl}(n)$: relative amplitude of n-th harmonic component

$$v_{kl}(n) = \frac{\sum_{c} \iint X^{(H)}_{klmn}(c,t,f)\,dt\,df + \mu^{(v)}_k(m)}{\sum_{c,n} \iint X^{(H)}_{klmn}(c,t,f)\,dt\,df + 1}.$$

A.6 τ_{kl}: onset time

$$\tau_{kl} = \frac{\sum_{c,m,n} \iint (t - m\phi_{kl}) X^{(H)}_{klmn}(c,t,f)\,dt\,df}{\sum_{c,m,n} \iint X^{(H)}_{klmn}(c,t,f)\,dt\,df}.$$

A.7 $\omega_{kl}(t)$: F0 trajectory

$$\omega_{kl}(t) = \frac{\sum_{c,m,n} \iint nf X^{(H)}_{klmn}(c,t,f)\,df}{\sum_{c,m,n} \iint n^2 X^{(H)}_{klmn}(c,t,f)\,df}.$$

A.8 ϕ_{kl}: diffusion of a Gaussian of power envelope

$$\phi_{kl} = \frac{-A^{(\phi)}_1 + \sqrt{A^{(\phi)2}_1 + 4A^{(\phi)}_2 A^{(\phi)}_0}}{2A^{(\phi)}_1}, \quad \text{where}$$

$$A^{(\phi)}_2 = \sum_{c,m,n} \iint X^{(H)}_{klmn}(c,t,f)\,dt\,df,$$

$$A^{(\phi)}_1 = \sum_{c,m,n} \iint m(t-\tau_{kl}) X^{(H)}_{klmn}(c,t,f)\,dt\,df, \text{ and}$$

$$A^{(\phi)}_0 = \sum_{c,m,n} \iint (t-\tau_{kl})^2 X^{(H)}_{klmn}(c,t,f)\,dt\,df.$$

A.9 σ_{kl}: diffusion of harmonic component along the frequency axis

$$\sigma_{kl} = \sqrt{\frac{\sum_{c,m,n} \iint (f - n\omega_{kl}(t))^2 X^{(H)}_{klmn}(c,t,f)\,dt\,df}{\sum_{c,m,n} \iint X^{(H)}_{klmn}(c,t,f)\,dt\,df}}.$$

A.10 $E^{(I)}_{kl}(t), F^{(I)}_{kl}(t,f)$: inharmonic tone model

$$E^{(I)}_{kl}(t) = \frac{\sum_c \int X^{(I)}_{kl}(c,t,f)\,df}{\sum_c \iint X^{(I)}_{kl}(c,t,f)\,dt\,df} \quad \text{and}$$

$$F^{(I)}_{kl}(t,f) = \frac{\sum_c X^{(I)}_{kl}(c,t,f) + \mu^{(F)}_k(f)}{\sum_c \int X^{(I)}_{kl}(c,t,f)\,df + 1}.$$