Proceedings of the 2nd European Workshop on the Integration of Knowledge, Semantic and Digital Media Technologies (EWIMT2005), pp.205-212, November 2005.

INTER:D: A DRUM SOUND EQUALIZER FOR CONTROLLING VOLUME AND TIMBRE OF DRUMS

Kazuyoshi Yoshii† **Masataka Goto**‡ **Hiroshi G. Okuno**†

†Department of Intelligence Science and Technology
Graduate School of Informatics, Kyoto University, Japan

‡National Institute of Advanced Industrial
Science and Technology (AIST), Japan

yoshii@kuis.kyoto-u.ac.jp m.goto@aist.go.jp okuno@i.kyoto-u.ac.jp

Keywords: Interactive music-playback interface, drum sound annotation, drum transcription, music information retrieval.

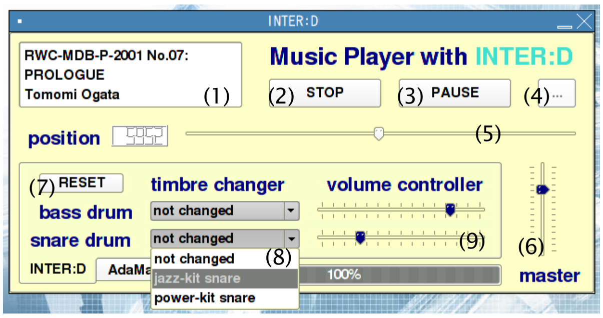
Abstract

A drum sound equalizer, called INTER:D, is described that enables a listener to control the volume and timbre of bass and snare drum sounds in commercial compact-disc recordings. Although the characteristics of the drum sounds are often closely related to the impression made by a musical piece, conventional graphic equalizers cannot adjust their characteristics because they have volume sliders only for the *frequency bands*. INTER:D provides *drum-specific volume sliders* that directly control the frequency components of drum sounds, enabling a listener to intuitively cut or boost the volume of each drum. In addition, INTER:D enables a listener to replace the original timbre of each drum with another timbre selected from a dropdown list. These interactive functions are achieved using an automatic music content analysis system based on low-level audio signal processing. This system can estimate the power spectrogram of each drum sound and detect the onset times in a musical piece. Subjective experiments showed that INTER:D can add a new dimension to the way users experience music.

1 Introduction

The development of interactive music-playback interfaces has recently been facilitated by the application of low-level audio processing techniques to music content analysis. For example, an automatic chorus detection method for compact disc (CD) recordings led to a new music listening station, called SmartMusicKIOSK [7]. Customers using conventional listening stations in music stores often want to jump directly to the chorus of a popular song, and SmartMusicKIOSK provides a "jump to chorus" button that enables the listener to jump to the next detected chorus section while skipping the sections of no interest. This is an *active listening* environment in which a listener can control the listening experience by interactively changing the playback position. Pachet and Delerue [13] proposed an active listening environment in which listeners can interactively spatialize sound sources. Unlike with SmartMusicKIOSK, however, this is achieved by using MIDI sound modules without audio-based automatic content analysis.

In this paper, we describe a new active listening environment based on automatic music content analysis of musical instruments. Listeners often focus on the melody line (e.g., a sung melody in popular music) in musical pieces. Even if the listener wants to emphasize the melody line in polyphonic mixtures by cutting the volume of accompaniment instruments (e.g., drums), he cannot do so with conventional graphic equalizers. Even if the listener is not comfortable with the timbre of the drums in a CD recording, she cannot replace it with a preferred one. To provide this capability, we developed a drum sound equalizer, called *INTER:D* (Instrument Equalizer for Drums). In this paper, we use the term "*equalizing*" to mean controlling the volume and timbre of instruments.



RWC-MDB-P-2001 No.07:
PROLOGUE
Tomomi Ogata

bass drum: not changed
snare drum: not changed

(1) display of music title and artist name (2) stop button
(3) play/pause toggle button (4) music file browser
(5) playback slider (6) master volume slider
(7) default restoring button
(8) dropdown lists for selecting another timbre
(9) drum-specific volume sliders

Figure 1: Screen snapshot of music player with INTER:D.

To equalize the bass and snare drum sounds, we cannot use the conventional graphic equalizers that are often incorporated into software or hardware music players. These equalizers cut or boost the power in various frequency bands by applying bandpass filters to the audio signal. However, they cannot control the volume of *only* the bass or snare drum. Moreover, it is impossible to change the timbre of a drum without changing the timbre of other instruments. For instance, if the listener tries to cut the volume of the bass drum, the graphic equalizer will attenuate only all the low-frequency sounds: it attenuates the spectral components derived *not only* from the bass drum but also from other musical instruments (e.g., snare drum and bass guitar).

On the other hand, INTER:D can control the volume and timbre of only the bass or snare drum. Figure 1 shows the interactive console of a music player equipped with INTER:D. This player includes the conventional playback interface (playback slider and stop/pause/play buttons) in the upper part of the window. The added equalizing interface (volume sliders and dropdown lists) is in the lower part of the window. Listeners can boost the volume of each drum sound by moving the appropriate slider from left (muted) to right (maximum volume) and vice versa. They can change the original timbre of each drum to their favorite timbre by selecting another one from the dropdown list. In this way, listeners can interactively control their listening experience while receiving auditory feedback in real time.

INTER:D is based on automatic content analysis for detection of the onsets of drum sounds and estimation of their spectrograms. For this analysis, we use our template-based drum transcription system called *AdaMast* [21], which uses the spectrograms of drum sounds as templates. AdaMast is composed of template-adaptation and template-matching methods. First, an initial spectrogram called a *seed template* is prepared for each drum sound and adapted to the actual drum-sound spectrograms in the target piece. Next, every onset time at which the adapted template is included in the spectrogram of the target piece is detected by using a carefully designed distance measure that is robust for spectral overlapping in sound mixtures.

The rest of this paper is organized as follows. Section 2 describes the equalizing functions of INTER:D. Section 3 explains why is needed AdaMast, and Section 4 explains its implementation. Section 5 describes the subjective experiments. Section 6 discusses our developed technique, and Section 7 summarizes the key points.

2 Equalizing Functions

INTER (Instrument Equalizer) is based on a novel equalizing concept — the frequency components of *each instrument*, not *each frequency band*, are adjusted. By applying this concept to drum sounds, we developed a drum-sound equalizer called INTER:D (INTER for Drums), which has two functions: volume control (cutting or boosting the volume) and timbre change for bass and snare drum sounds. A listener can easily control these functions by using a familiar easy-to-use interface, which is shown in Figure 1. In this section, we explain the implementation of these functions and their improvements.

2.1 Volume Control

INTER:D provides a volume cutting/boosting function for the bass and snare drum sounds in a musical piece containing sounds of various instruments. Let X denote the target drum sound to be equalized. The algorithm is as follows (Figure 2):

1. *Spectrogram Estimation*: The power spectrogram of drum sound X used in the musical piece is estimated. Let P_X denote that spectrogram. $P_X(t, f)$ ($1 \leq t \leq T$ [frames], $1 \leq f \leq F$ [bins]) represents the local frequency component in the time-frequency domain (T and F are fixed values).
2. *Onset Detection*: INTER:D detects the onset times of drum sound X in the monaural polyphonic audio signal of the piece.

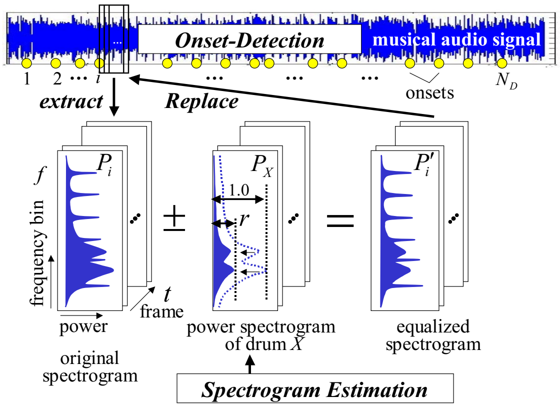


Figure 2: Overview of cutting/boosting volume of drums.

3. Starting from each onset time, spectrogram segment P_i ($i = 1, \cdots, N_D$) which has the same length as power spectrogram P_X is extracted from the input audio signal, where N_D is the number of detected onsets.
4. Power spectrogram P_X is added to each spectrogram segment P_i, weighted by arbitrary power change ratio r (cut: $-1 \leq r < 0$, boost: $0 < r$):

$$P'_i(t, f) = P_i(t, f) + r \cdot P_X(t, f), \tag{1}$$

   where P'_i is an equalized spectrogram segment with a phase the same as that of the original segment before equalizing.
5. An equalized spectrogram of the input audio signal is obtained by replacing P_i with its P'_i.
6. An equalized audio signal is obtained by applying overlap-add synthesis to the equalized power spectrogram of the input audio signal.

To analyze input audio signals sampled at 44.1 [kHz], we used short-time Fourier transform (STFT) with a Hanning window (4096 points) with a shifting interval of 441 points (i.e., one frame is equivalent to 10 [ms]). T and F were experimentally set to 10 [frames] and 2048 [bins].

As described above, the system must automatically estimate the spectrograms of the drum sounds and detect their onset times. To annotate them, we applied our drum transcription system called AdaMast, which is based on low-level signal processing. The details are described in Section 3.

2.2 Timbre Change

INTER:D also provides a function for changing the timbre of drum sound X to that of another drum sound (let Y be that drum sound). The algorithm is as follows:

1. To obtain an equalized audio signal in which the sounds of drum X are muted, power change ratio r is set to -1.0, and the volume control operations described above are performed.
2. the audio signal of drum Y (a solo tone) is added to the equalized audio signal at each detected onset time.

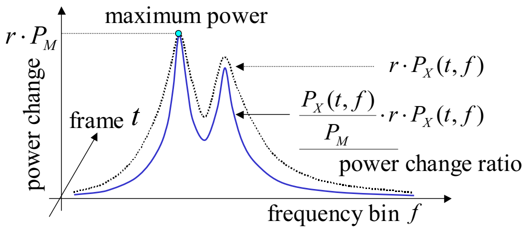


Figure 3: Adjusting power change ratio in each frame and each frequency bin according to power in that position.

2.3 Quality Improvement

Although the basic operations described in Sections 2.1 and 2.2 are effective, there is much room for improving the sound quality. We therefore propose two methods for generating higher-quality equalized audio signals: *adjustment of the power change ratio* and *restoration of the temporal continuity*.

2.3.1 Adjustment of Power Change Ratio

The estimated power spectrogram P_X actually includes small local frequency components of other musical instrument sounds (i.e., P_X cannot be the same as the precise power spectrogram of an isolated tone) because it is estimated from a complex polyphonic spectrogram. Therefore, if we use only P_X for the equalizing process, frequency components not derived from the target drum sound are also unnecessarily adjusted.

To solve this problem, we dynamically adjust power change ratio r to minimize unnecessary power adjustment. As shown in Figure 3, we decrease r when adjusting the small local frequency components in P_X. We replace r in Equation (1) with:

$$R(t, f) = \frac{P_X(t, f)}{P_M} \cdot r, \qquad (2)$$

where P_M is the maximum local power in P_X.

2.3.2 Restoration of Temporal Continuity

When power spectrogram P_X is added to or subtracted from an original spectrogram segment, the temporal continuity is deteriorated before and after the equalized spectrogram segment, as depicted in Figure 4. This temporal discontinuity may have a negative influence on human music perception.

To solve this problem, we perform spectral smoothing in the neighbor frames of each discontinuous frame. For each frequency band, frequency components in the range of 4 [frames] (i.e., 2 [frames] before and after a discontinuous frame) are smoothed using Savitzky-Golay's smoothing method [18].

3 Drum Transcription Technique

To implement INTER:D, we must estimate the power spectrograms of the individual bass and snare drums (*spectrogram estimation*) and detect the onset times of each drum sound (*onset detection*). Both are difficult because each musical piece has different power spectrograms (sound characteristics) for the drum sounds and there are many onset times of other sounds in a real-world sound mixture. To solve these problems, we used our AdaMast drum transcription system [21]. It uses template-adaptation and template-matching methods, using the power spectrogram of each drum sound as a template. In this section, we discuss the advantages of AdaMast for implementing INTER:D.

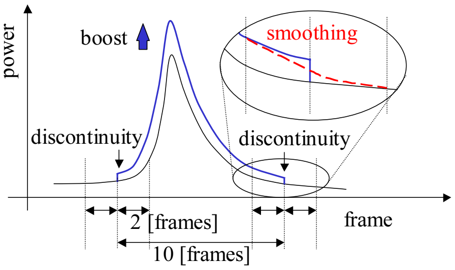


Figure 4: Smoothing frequency components along time axis before and after equalized spectrogram segment.

3.1 Requirements

The *spectrogram estimation* and *onset detection*, must be performed for both the bass and snare drums.

Spectrogram Estimation The power spectrogram of the individual tone of each drum sound must be estimated in the polyphonic spectrogram of the musical piece. The spectrogram of each drum sound is not registered with the system in advance because the sound characteristics depend on the piece.

Onset Detection The onset times of each drum must be detected in the polyphonic spectrogram of the musical piece. This is difficult because various musical instrument sounds often overlap with the drum sounds.

3.2 Approach

To meet these requirements, we use the AdaMast drum transcription system. The template-adaptation method of AdaMast first yields the power spectrogram (estimated sound characteristics) of each drum used sound in the input audio signal. The template-matching method of AdaMast then detects its onset times from the same audio signal.

Template Adaptation obtains the power spectrograms of the individual bass and snare drum sounds in the input audio signal. Before using this method, we must prepare individual power spectrograms (we called them *seed templates* in a previous study [21]) for both drums — two templates in total. Note that the seed templates are different from the actual power spectrograms of the drum sounds in the input audio signal. This method adapts each seed template to its corresponding power spectrograms of drum sounds included in the input audio signal.

Template Matching detects all the onset times of the bass and snare drum sounds in the polyphonic audio signal, even if other instrument sounds overlap them. To enable this, we designed a distance measure that is robust for polyphonic mixtures. Using this distance measure, we compare the adapted power spectrogram obtained by template adaptation with the power spectrogram of the input audio signal to identify the onset times.

The adaptation and matching are sequential and are independently performed for transcription of the bass and snare sounds. The detailed algorithms are described in Section 4.

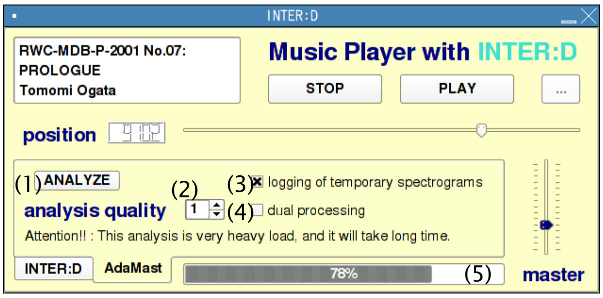
3.3 Related Work

From the viewpoint of methodology, drum transcription methods are roughly categorized into three types: feature-based classification, sound source separation, and template-based detection. They can also be categorized by focusing on the complexity of the input audio signals: individual tones, drum tracks, or musical pieces such as popular songs.

Feature-based classification methods are based on acoustic feature models trained using a database. Herrera *et al.* [11] compared conventional classifiers in experiments on identifying individual drum sounds. To transcribe drum sounds in drums-only audio signals, the use of N-grams [14], probabilistic models [15], and HMM&SVM [6] have been proposed. To identify drum sounds extracted from polyphonic audio signals, Sandvolt *et al.* [17] proposed a feature-model adaptation method that is robust to the distortion of features since feature distortion caused by other sounds is a major problem.

Sound source separation methods, which are commonly used, originated from spectrogram decomposition formulation in independent subspace analysis (ISA) [1]. To transcribe drum sounds in audio signals of drum tracks, various assumptions are made in decomposing a single music spectrogram into multiple spectrograms of drum tracks; ISA [3, 19] assumes the statistical independence of sources, non-negative matrix factorization (NMF) [16] assumes their non-negativity, and sparse coding [20] assumes their non-negativity and sparseness. Further developments were made by FitzGerald *et al.* [4, 5]. They proposed prior subspace analysis (PSA) [5], which assumes the prior frequency characteristics of drum sounds, and applied it to transcribe drum sounds in the presence of harmonic sounds [4]. For the same purpose, Dittmar and Uhle [2] adopted non-negative ICA that considers the non-negativity of sources. To attain good separation results in any case, it is necessary to estimate the number of sources, but it is difficult to precisely estimate it in general.

Template-based detection methods are based on a typical pattern recognition approach — the distance between a template and an input pattern is calculated. Goto and Muraoka [9] proposed a template-matching method that uses spectrogram templates, and transcribes drum sounds in drum-track audio signals consisting of drum sounds only. Gouyon *et al.* [10] proposed a method that classifies mixed sounds extracted from polyphonic audio signals into two categories (bass and snare drums). To detect these drum sounds to be classified, they developed a template-adaptation method that uses waveform templates. It can deal with drum-sound variations found in musical pieces. Zils *et al.* [22] extended Gouyon's method to extraction of bass and snare drum sounds from CD recordings. In general, it is difficult to deal with the difference between a template and an actual pattern used in a musical piece. To deal with this difference in the time-frequency domain and achieve more robust performance, AdaMast [21] was developed by integrating Goto's matching method and Zils' adaptation method.


INTER:D

RWC-MDB-P-2001 No.07:
PROLOGUE
Tomomi Ogata

Music Player with INTER:D

analysis quality 1

☒ logging of temporary spectrograms

□ dual processing

Attention!! : This analysis is very heavy load, and it will take long time.

78%

(1) analyze/cancel toggle button
(2) spin box for indicating analysis quality
(3) checkbox for enabling/disabling logging of temporary spectrograms
(4) checkbox for enabling/disabling dual processing
(5) progress indicator of content analysis (spectrogram estimation and onset detection)

Figure 5: Console panel of AdaMast controller.

3.4 Advantages of AdaMast

AdaMast is fast enough to achieve a stress-free music appreciation environment. This was an important factor in our decision to use AdaMast to transcribe drum sounds. Figure 5 shows the console panel used to control AdaMast. Because content analysis (drum transcription) has a heavier load than the equalizing processing, the analysis parameters should be adjusted to match machine performance. If all options are applied, it will take less than the length (playing time) of a musical piece — i.e., it is faster than the real-time performance — to complete content analysis on a dual 3.0-GHz Xeon machine.

4 Implementation of AdaMast

As described above, AdaMast [21] is a template-based drum-sound transcription system that consists of successive template-adaptation and template-matching. The former yields power spectrogram P_X, and the latter detects the onset times of the corresponding drum sounds. These results are used by the equalizing functions of INTER:D.

4.1 Algorithms

The template-adaptation and template-matching methods can be executed in parallel for the bass and snare drums.

4.1.1 Template Adaptation

This method obtains a spectrogram template that is adapted to its corresponding drum-sound spectrogram in the polyphonic

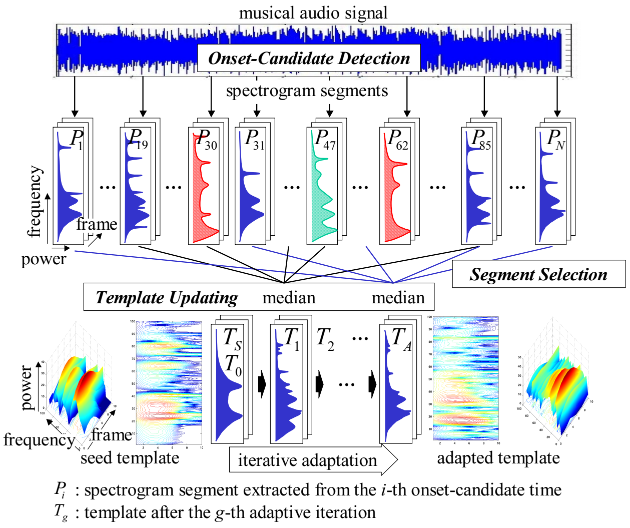


Figure 6: Overview of template adaptation method.

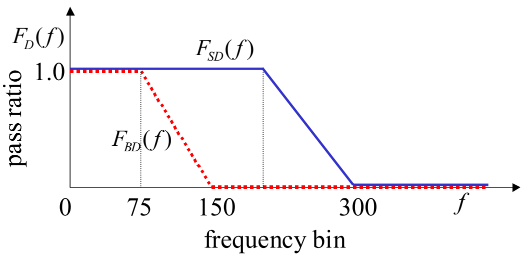


Figure 7: Low-pass filter functions F_{BD} and F_{SD} represent typical frequency characteristics of bass and snare drums.

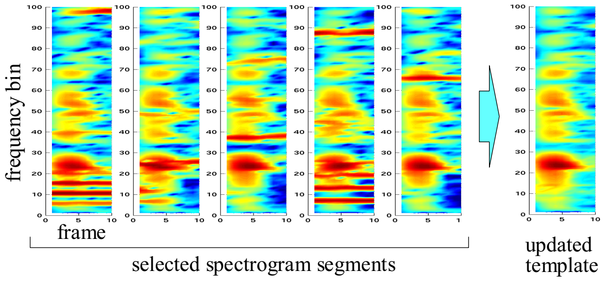


Figure 8: Template updating by collecting median power at each frame and each frequency bin for selected spectrogram segments.

audio signal. The adapted template is used as power spectrogram P_X for the equalizing processing. Before starting the adaptation, it is necessary to prepare an arbitrary seed template for each bass and snare drum sound.

Our method is based on an iterative adaptation algorithm. An overview is shown in Figure 6. First, *onset-candidate detection* stage roughly detects onset candidates in the input audio signal of a musical piece. Starting from each onset candidate, a spectrogram segment with a fixed time length is extracted from the power spectrogram of the input audio signal. Then, using the seed template and all the spectrogram segments, the iterative algorithm successively applies *segment selection* and *template updating* to obtain an adapted template.

Let T_i $(i = 1 \cdots N_D)$ denote a frame detected as an onset candidate and P_i denote a spectrogram segment extracted from T_i (N_D is the number of detected onset candidates). These selection and updating work as follows:

1. *Segment selection* calculates the reliability R_i that spectrogram segment P_i includes the drum sound spectrogram. The reliability is defined as the reciprocal of the Euclidean spectral distance:

   $$R_i = \frac{1}{\sqrt{\sum_{t=1}^{10} \sum_{f=1}^{2048} \left(\acute{T}_g(t,f) - \acute{P}_i(t,f) \right)^2}}, \tag{3}$$

   where T_g is the template after the g-th adaptive iteration. In practice, we used a modified version of this measure. $\acute{T}_g$ and $\acute{P}_i$ are low-pass filtered spectrograms:

   $$\acute{T}_g(t,f) = F_D(f)\, T_g(t,f), \tag{4}$$
   $$\acute{P}_i(t,f) = F_D(f)\, P_i(t,f), \tag{5}$$

   where $F_D(f)$ $(\text{D} = \text{BD}, \text{SD})$ is a low-pass filter function, as shown in Figure 7. We assume that it represents the typical frequency characteristics of bass drum sounds (BD) and snare drum sounds (SD). Spectrogram segments with high reliabilities are then selected; this selection is based on a fixed ratio to the total number of segments.

2. *Template updating* then reconstructs an updated template by estimating the power that is defined, at each time and each frequency bin, as the median power among the selected spectrogram segments (Figure 8). The median operation can suppress harmonic components in the updated template. The template is thus adapted to the current piece and used for the next adaptive iteration. The updated template, $\acute{T}_{g+1}$, is weighted by filter function F_D and is obtained by

   $$\acute{T}_{g+1}(t,f) = \underset{1 \le i \le N_S}{\mathrm{median}}\, \acute{P}^{(i)}(t,f), \tag{6}$$

   where $P^{(i)}$ $(i = 1, \cdots, N_S)$ are the spectrogram segments selected by *segment selection*. N_S is the number of selected spectrogram segments, which is set to $0.1 \times N_D$ in this paper.

4.1.2 Template Matching

This method detects all the onset times of the drum sounds in the polyphonic audio signal, even if other musical instrument sounds overlap the drum sounds. To find the actual onset times, this method determines *whether the drum sound actually occurs at each onset candidate time*, as shown in Figure 9. The matching distance is calculated using Goto's distance measure [9]. Since this method focuses on whether the adapted template is included in a spectrogram segment, it can calculate an appropriate distance even if the drum sound is overlapped by other musical instrument sounds.

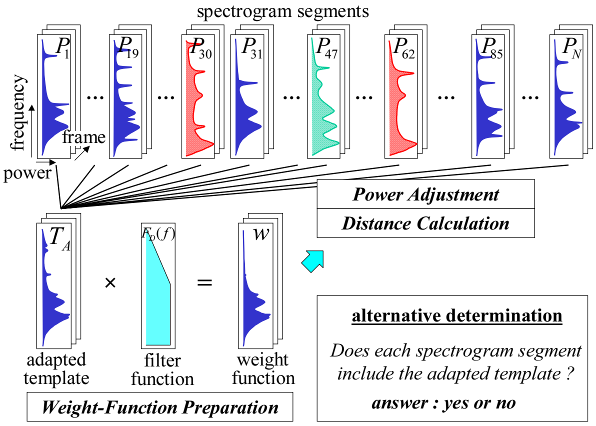


Figure 9: Overview of template-matching method.

1. *Weight-function preparation* generates a function that represents the spectral saliency of each frequency component in the adapted template. This function is used for selecting characteristic frequency bins. The weight function w is defined as

$$w(t, f) = F_D(f)\, T_A(t, f), \qquad (7)$$

where T_A is the adapted template, which is equivalent to power spectrogram P_X used for the equalizing processing, and F_D is the filter function.

2. *Power adjustment* calculates the power difference between the template and each spectrogram segment by focusing on the characteristic frequency bins. If the power difference is larger than a threshold, it judges that the drum sound spectrogram does not appear in that segment and does not execute the subsequent processing. Otherwise, the power of that segment is adjusted to compensate for the power difference. Let P_i' be a power-adjusted spectrogram segment.

3. *Distance calculation* calculates the spectral distance between adapted template T_A and each P_i'. If $P_i'(t, f)$ is larger than $T_A(t, f)$, Goto's distance measure regards $P_i'(t, f)$ as a mixture of frequency components not only of the drum sounds but also of other musical instrument sounds. In other words, if we determine that $P_i'(t, f)$ includes $T_A(t, f)$, then the local distance at frame t and frequency bin f is minimized. Therefore, the local distance is defined as

$$\gamma_i(t, f) = \begin{cases} 0 & \text{if } \left(P_i'(t, f) - T_A(t, f) \geq \Psi\right), \\ 1 & \text{otherwise}, \end{cases} \qquad (8)$$

where Ψ is a negative constant, which is set to -12.5 [dB] in this paper.

Total distance Γ_i is calculated by integrating local distance γ_i in the time-frequency domain, weighted by w:

$$\Gamma_i = \sum_{t=1}^{10} \sum_{f=1}^{2048} w(t, f)\, \gamma_i(t, f). \qquad (9)$$

To determine whether the target drum sound occurred at a time corresponding to spectrum segment P_i', distance Γ_i is compared with threshold Θ_Γ. If $\Gamma_i < \Theta_\Gamma$, we conclude that the target drum sound occurred. The Θ_Γ is automatically determined using Otsu's thresholding algorithm [12].

4.2 Advantages of Using Two Distance Measures

We use two different distance measures between the template adaptation and matching methods. In the adaptation method, it is desirable to detect only semi-pure drum sounds that have little overlap with other sounds. Those drum sounds tend to result in a good adapted template that includes few frequency components of other sounds. Because it is not necessary to detect all the onset times of the target drum sounds, the distance measure does not need to consider spectral overlapping of other sounds. In the matching method, on the other hand, we used Goto's distance measure because it is necessary to exhaustively detect all the onset times even if the target drum sounds are overlapped by other sounds.

5 Evaluation

We performed subjective experiments using five songs taken from a popular music database, "*RWC-MDB-P-2001*" developed by Goto *el al.* [8]. These songs contain sounds of vocals and various instruments, as songs in commercial CDs typically do. All original data were sampled at 44.1 kHz with 16 bits, stereo. We converted them to monaural signals.

5.1 AdaMast

We first evaluated the onset detection accuracy of AdaMast. The results are shown in Table 1. The power spectrograms and onset times of the bass and snare drums were obtained using the template-adaptation and template-matching methods. The power spectrograms were accurately estimated.

5.2 Volume Control

We tested INTER:D with five subjects to evaluate the quality of volume control for two cases: a 10-[dB] cut and a 5-[dB] boost. Each subject listened to audio signals in which the volume of the bass or snare drum sounds was changed by using INTER:D and audio signals in which the low-frequency sounds were equalized by using a traditional graphic equalizer. They compared these signals in terms of the impression of the negative effects on sounds other than the target drum sounds (bass or snare drum sounds). The comparison results were numerically scored: a value of 10 (reference in comparison) corresponds to the impression of negative effects caused by the graphic equalizer; a value of 0 corresponds to the impression of almost no negative effects.

The five subjects each gave four scores corresponding to four cases (only the bass drum was attenuated, only the snare

Table 1: Onset detection accuracy.

piece	bass	snare	average
No. 7	98.0 %	85.7 %	91.9 %
No. 21	77.3 %	84.7 %	81.0 %
No. 35	76.1 %	74.5 %	75.3 %
No. 47	94.7 %	74.8 %	84.8 %
No. 52	99.3 %	90.8 %	95.1 %

Table 2: Volume control evaluation results for INTER:D.

	10 [dB] cut		5 [dB] boost	
piece	bass	snare	bass	snare
No. 7	3.6	5.0	2.4	3.4
No. 21	3.2	4.0	2.4	2.4
No. 35	3.6	5.0	2.2	3.0
No. 47	3.8	5.0	2.6	3.2
No. 52	3.0	4.0	2.6	2.8
average	3.4	4.6	2.4	3.0

Note: This table shows the average score for negative effects on other instrument sounds when the bass or snare drum sounds were attenuated or amplified. The lower the score, the better the impression of the sounds equalized by INTER:D. The scores were averaged over the five subjects.

drum was attenuated, only the bass drum was amplified, and only the snare drum was amplified) for each of the five songs[1].

Table 2 shows the results of this subjective evaluation. These results of low scores indicate that the negative influence caused by INTER:D was much less than that caused by the graphic equalizer. The scores for snare drum sounds were higher (worse) than those for bass drum sounds: equalizing snare drum sounds was more difficult than equalizing bass drum sounds because frequency components of snare drum sounds are distributed in wide frequency bands. In addition, these results also indicate that the attenuation was more difficult (had worse scores) than the amplification. By comparing Table 1 with Table 2, the detection accuracy was not related to the evaluated scores at least in our experiments.

5.3 Timbre Change

We tested INTER:D with the same five subjects to evaluate the quality of timbre change for two cases, timbre replacement and timbre muting. Each subject listened to audio signals in which the bass and snare drum sounds used were replaced with other drum sounds with different timbres and to audio signals in which the bass and snare drum sounds were muted (attenuated completely). The subject was then asked to complete a subjective questionnaire about the impression made by those signals.

Most subjects answered that the timbre change (timbre replacement) was an interesting function because good-quality audio signals were generated even though the impression made by those signals with different drum sounds often greatly differed from that made by the original signals. We found that accurate onset detection was more important for improving the sound quality when changing the timbre than when controlling the volume. On the other hand, most subjects thought that the audio signals in which the drum sounds were muted sounded artificial. Muting drum sounds resulted in low-quality audio signals because frequency components that did not belong to the drum sounds were also largely attenuated. When other drum sounds were added in the timbre replacement, however, those additional sounds compensated for the undesired spectral attenuation, resulting in good-quality audio signals.

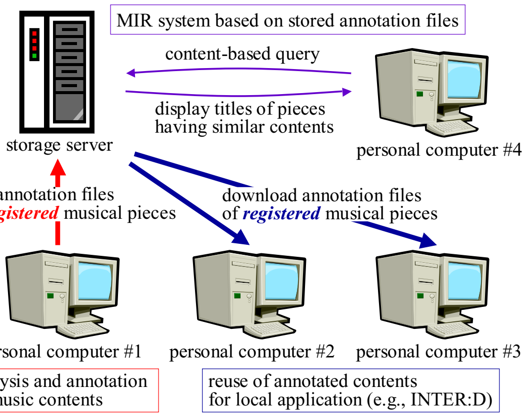


Figure 10: Music information retrieval (MIR) system based on automatic annotation and sharing of music contents.

6 Discussion

The importance of music content analysis for audio signals has been increasing in the field of music information retrieval (MIR). MIR aims at retrieving musical pieces by executing a query about not only text information such as artist names and music titles but also musical contents such as rhythms and melodies. Although the amount of digitally recorded music available over the Internet is rapidly increasing, there are only a few ways of using text information to efficiently find target musical pieces in a huge music database. Music content analysis enables MIR systems to automatically understand the contents of musical pieces and to deal with them even if they do not have metadata about the artists and titles. Although onset time information for drum sounds is low-level music content, it can be used as a basis for higher-level music content analysis concerning the rhythm such as beat tracking and tempo estimation.

One reason we used AdaMast for drum transcription is that it features fast content analysis, as described in Section 3.4. It also is suitable for transmitting the analysis results over the Internet. Once the power spectrograms and onset times of the drum sounds in a piece are analyzed, the drum-sound annotation files should be shared among people who have the same piece in order to prevent computer resources from being wasted. This framework of music content sharing can be considered an extended version of the text-based CD Database (CDDB) service. Since INTER:D requires only a single spectrogram and

[1] The sound files are available at http://winnie.kuis.kyoto-u.ac.jp/members/yoshii/EWIMT2005/.

the onset times for each drum sound, the size of each annotation file is only about 300 [KB]. If we did not use AdaMast and instead used a sound source separation method to implement INTER:D, it would be necessary to transmit all the signals for the drum tracks for equalizing the music to another computer. However, the annotation file is too large (approximately over 100 [MB]) to be shared efficiently through the Internet. It is also difficult to store many annotation files on a server. The fast and compact drum-sound annotation capability of AdaMast is thus important for practical use.

Figure 10 shows a diagram of an MIR system based on the integration of low-level digital audio technology, semantic annotation, and content sharing. A user using a personal computer can download annotation files for musical pieces if they are registered in a storage server or upload annotation files if they are not registered. A user can also retrieve musical pieces that have contents similar to a query. Music content annotation systems such as AdaMast can thus form the basis of Semantic Web for music.

7 Conclusion

We have described a drum sound equalizer, INTER:D, with a simple and easy-to-use interface that has sliders to control the volume and dropdown lists to select the timbre. A listener can control the volume of bass or snare drum sounds without changing the volume of other sounds by moving the corresponding sliders. In addition, the listener can change the original timbre of each drum sound by selecting a different one from the timbre list. These functions are achieved by automatically detecting the onsets of each drum sound while estimating its power spectrogram. In other words, drum-sound annotation based on low-level signal processing enables new music interaction in active appreciation of music.

Testing results of popular song equalization on a real-time music-playback system we implemented demonstrated the effectiveness of the concept of active music listening. We plan to apply our equalizing method to various kinds of instruments. and to apply the drum-sound annotation technique to the development of a music information retrieval system sharing content-based music annotations.

Acknowledgements

This research was partially supported by MEXT, Grant-in-Aid for Scientific Research, and by the COE program of MEXT.

References

[1] M. A. Casey and A. Westner, “Separation of mixed audio sources by independent subspace analysis,” *ICMC*, 2000.

[2] C. Dittmar and C. Uhle, “Further steps towards drum transcription of polyphonic music,” *AES 116th Conv.*, 2004.

[3] D. FitzGerald, E. Coyle, and B. Lawlor, “Sub-band independent subspace analysis for drum transcription,” *DAFX*, 2002, pp. 65–69.

[4] D. FitzGerald, B. Lawlor, and E. Coyle, “Drum transcription in the presence of pitched instruments using prior subspace analysis,” *ISSC*, 2003, pp. 202–206.

[5] ——, “Prior subspace analysis for drum transcription,” *AES, 114th Conv.*, 2003.

[6] O. Gillet and G. Richard, “Automatic transcription of drum loops,” *ICASSP*, 2004, pp. 269–272.

[7] M. Goto, “SmartMusicKIOSK: Music listening station with chorus-search function,” *UIST*, 2002, pp. 31–40.

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

[9] M. Goto and Y. Muraoka, “A sound source separation system for percussion instruments,” *IEICE Trans. D-II*, vol. J77-D-II, no. 5, pp. 901–911, May 1994.

[10] F. Gouyon, F. Pachet, and O. Delerue, “On the use of zero-crossing rate for an application of classification of percussive sounds,” *DAFX*, 2000.

[11] P. Herrera, A. Yeterian, and F. Gouyon, “Automatic classification of drum sounds: A comparison of feature selection methods and classification techniques,” *ICMAI, LNAI2445*, 2002, pp. 69–80.

[12] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE Trans. Sys., Man, and Cybern.*, vol. 6, no. 1, pp. 62–66, 1979.

[13] F. Pachet and O. Delerue, “Annotations for real time music spatialization,” *Int. Workshop on Knowledge Representation for Interactive Multimedia Systems*, 1998.

[14] J. Paulus and A. Klapuri, “Conventional and periodic N-grams in the transcription of drum sequences,” *ICME*, 2003, pp. 737–740.

[15] ——, “Model-based event labeling in the transcription of percussive audio signals,” *DAFX*, 2003, pp. 73–77.

[16] ——, “Drum transcription with non-negative spectrogram factorisation,” *EUSIPCO (in press)*, 2005.

[17] V. Sandvold, F. Gouyon, and P. Herrera, “Percussion classification in polyphonic audio recordings using localized sound models,” *ISMIR*, 2004, pp. 537–540.

[18] A. Savitzky and M. Golay, “Smoothing and differentiation of data by simplified least squares procedures,” *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[19] C. Uhle, C. Dittmar, and T. Sporer, “Extraction of drum tracks from polyphonic music using independent subspace analysis,” *ICA*, 2003, pp. 843–848.

[20] T. Virtanen, “Sound source separation using sparse coding with temporal continuity objective,” *ICMC*, 2003, pp. 231–234.

[21] K. Yoshii, M. Goto, and H.G. Okuno, “Automatic drum sound description for real-world music using template adaptation and matching methods,” *ISMIR*, 2004, pp. 184–191.

[22] A. Zils, F. Pachet, O. Delerue, and F. Gouyon, “Automatic extraction of drum tracks from polyphonic music signals,” *WEDELMUSIC*, 2002, pp. 179–183.