Regular Paper

Instrogram: Probabilistic Representation of Instrument Existence for Polyphonic Music

Tetsuro Kitahara,† Masataka Goto,†† Kazunori Komatani,†
Tetsuya Ogata† and Hiroshi G. Okuno†

This paper presents a new technique for recognizing musical instruments in polyphonic music. Since conventional musical instrument recognition in polyphonic music is performed notewise, i.e., for each note, accurate estimation of the onset time and fundamental frequency (F0) of each note is required. However, these estimations are generally not easy in polyphonic music, and thus estimation errors severely deteriorated the recognition performance. Without these estimations, our technique calculates the temporal trajectory of *instrument existence probabilities* for every possible F0. The instrument existence probability is defined as the product of a *nonspecific instrument existence probability* calculated using the PreFEst and a *conditional instrument existence probability* calculated using hidden Markov models. The instrument existence probability is visualized as a spectrogram-like graphical representation called the *instrogram* and is applied to MPEG-7 annotation and instrumentation-similarity-based music information retrieval. Experimental results from both synthesized music and real performance recordings have shown that instrograms achieved MPEG-7 annotation (instrument identification) with a precision rate of 87.5% for synthesized music and 69.4% for real performances on average and that the instrumentation similarity measure reflected the actual instrumentation better than an MFCC-based measure.

1. Introduction

The goal of our study is to enable users to retrieve musical pieces based on their instrumentation. The types of instruments that are used are important characteristics for retrieving musical pieces. In fact, the names of certain musical genres, such as "piano sonata" and "string quartet", are based on instrument names. There are two strategies for instrumentation-based music information retrieval (MIR). The first allows users to directly specify musical instruments (e.g., searching for a piano solo or string music). This strategy is useful because, unlike other musical elements such as chord progressions, specifying instruments does not require any special knowledge. The other strategy is Query-by-Example, where once users specify a musical piece that they like, the system searches for pieces that have similar instrumentation to the specified piece. This strategy is also particularly useful for automatically generating playlists for background music.

The key technology for achieving the above-mentioned MIR is to recognize musical instruments from audio signals. Although musical instrument recognition studies mainly dealt with solo musical sounds in the 1990s [1], the number of studies dealing with polyphonic music has been increasing in recent years. Kashino, et al. [2] developed a computational architecture for music scene analysis called OPTIMA, which recognizes musical notes and instruments based on the Bayesian probability network. They subsequently proposed a method that identifies the instrument playing each musical note based on template matching with template adaptation [3]. Kinoshita, et al. [4] improved the robustness of OPTIMA to the overlapping of frequency components, which occurs when multiple instruments are played simultaneously, based on feature adaptation. Eggink, et al. [5] tackled this overlap problem with the missing feature theory. They subsequently dealt with the problem of identifying only the instrument playing the main melody based on the assumption that the main melody's partials would suffer less from other sounds occurring simultaneously [6]. Vincent, et al. [7] formulated both music transcription and instrument identification as a single optimization based on independent subspace analysis. Essid, et al. [8] achieved F0-estimation-less instrument recognition based on a priori knowledge about the instrumentation for ensembles. Kitahara, et al. [9] proposed techniques to solve the above-mentioned overlap

† Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University
†† National Institute of Advanced Industrial Science and Technology (AIST)

problem and to avoid musically unnatural errors using musical context.

The common feature of these studies, except for Refs. 7) and 8), is that instrument identification is performed for each frame or each note. In the former case [5),6)], it is difficult to obtain a reasonable accuracy because temporal variations in spectra are important characteristics of musical instrument sounds. In the latter case [2)~4),9)], the identification system has to first estimate the onset time and fundamental frequency (F0) of musical notes and then extract the harmonic structure of each note based on the estimated onset time and F0. Therefore, the instrument identification suffers from errors of onset detection and F0 estimation. In fact, correct data for the onset times and F0s were introduced manually in the experiments reported in Refs. 3) and 9).

In this paper, we propose a new technique that recognizes musical instruments in polyphonic musical audio signals without using onset detection or F0 estimation as explicit and deterministic preprocesses. The key concept underlying our technique is to visualize the probability that the sound of each target instrument exists at each time and with each F0 as a spectrogram-like representation called an *instrogram*. This probability is defined as the product of two kinds of probabilities, called *nonspecific instrument existence probability* and *conditional instrument existence probability*, which are calculated using the PreFEst [10)] and hidden Markov models, respectively. The advantage of our technique is that errors due to the calculation of one probability do not influence the calculation of the other probability because the two probabilities can be calculated independently.

In addition, we describe the application of the instrogram technique to MPEG-7 annotation and MIR based on instrumentation similarity. To achieve the annotation, we introduced two kinds of tags to the MPEG-7 standard. The first is designed to describe the probabilities directly, and the second is designed to obtain a symbolic representation such as an event whereby a piano sound occurs at time t_0 and continues until t_1. Such a representation can be obtained using the Viterbi search on a Markov chain with states that correspond to instruments. To achieve the MIR based on instrumentation similarity, the distance (dissimilarity) between two instrograms is calculated using dynamic time warping (DTW). A simple prototype system of similarity-based MIR is also achieved based on our instrumentation-based similarity measure.

2. Instrogram

The instrogram is a spectrogram-like graphical representation of a musical audio signal, which is useful for determining which instruments are used in the signal. In a basic format, an instrogram correponds to a specific instrument. The instrogram has horizontal and vertical axes representing time and frequency, and the intensity of the color of each point (t, f) shows the probability $p(\omega_i; t, f)$ that the target instrument ω_i is used at time t and at an F0 of f. An example is presented in **Fig. 1**. This example shows the results of analyzing an audio signal of "Auld Lang Syne" played on the piano, violin, and flute. The target instruments for analysis were the piano, violin, clarinet, and flute. If the instrogram is too detailed for some purposes, it can be simplified by dividing the entire frequency region into a number of subregions and merging the results within each subregion. A simplified version of Fig. 1 is given in **Fig. 2**. The original or simplified instrogram shows that the melodies in the high (approx. note numbers 70–80), middle (60–75), and low (45–60) pitch regions are played on flute, violin, and piano, respectively.

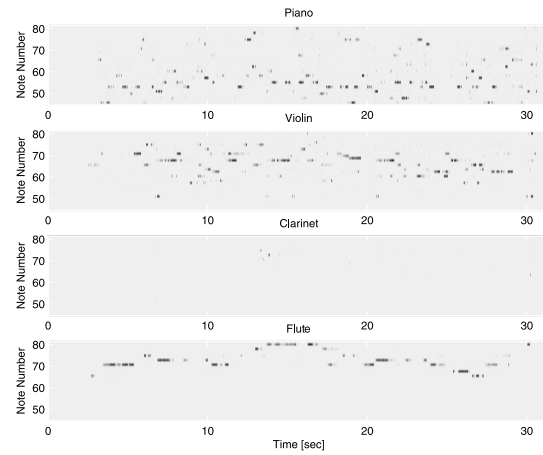


Fig. 1 Example of the instrogram ☆.

☆ Color versions of all figures are available at: http://winnie.kuis.kyoto-u.ac.jp/~kitahara/instrogram/IPSJ07/

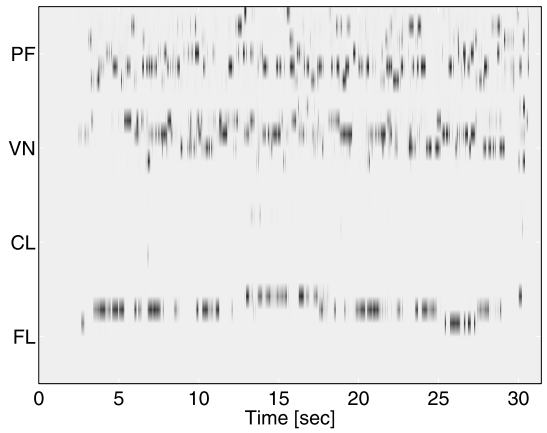


Fig. 2 Simplified (summarized) instrogram for Fig. 1.

3. Algorithm for Calculating Instrogram

Let $\Omega = \{\omega_1, \cdots, \omega_m\}$ be the set of target instruments. We then have to calculate the probability $p(\omega_i; t, f)$ that a sound of the instrument ω_i with an F0 of f exists at time t for every target instrument $\omega_i \in \Omega$. This probability is called the *instrument existence probability* (IEP). Here, we assume that multiple instruments are not being played at the same time and at the same F0, that is, $\forall \omega_i, \omega_j \in \Omega$: $i \neq j \Longrightarrow p(\omega_i \cap \omega_j; t, f) = 0$. Let ω_0 denote the silence event, which means that no instruments are being played, and let $\Omega^+ = \Omega \cup \{\omega_0\}$. The IEP then satisfies $\sum_{\omega_i \in \Omega^+} p(\omega_i; t, f) = 1$. When the symbol "X" denotes the union event of all target instruments, which stands for the existence of *some* instrument (i.e., $\mathrm{X} = \omega_1 \cup \cdots \cup \omega_m$), the IEP for each $\omega_i \in \Omega$ can be calculated as the product of two probabilities:

$$p(\omega_i; t, f) = p(\mathrm{X}; t, f)\ p(\omega_i | \mathrm{X}; t, f),$$

because $\omega_i \cap \mathrm{X} = \omega_i \cap (\omega_1 \cup \cdots \cup \omega_i \cup \cdots \cup \omega_m) = \omega_i$. Above, $p(\mathrm{X}; t, f)$, called the *nonspecific instrument existence probability* (NIEP), is the probability that the sound of some instrument with an F0 of f exists at time t, while $p(\omega_i | \mathrm{X}; t, f)$, called the *conditional instrument existence probability* (CIEP), is the conditional probability that, if the sound of some instrument with an F0 of f exists at time t, the instrument is ω_i. The probability $p(\omega_0; t, f)$ is given by $p(\omega_0; t, f) = 1 - \sum_{\omega_i \in \Omega} p(\omega_i; t, f)$.

3.1 Overview

Figure 3 shows an overview of the algorithm for calculating an instrogram. Given an audio signal, the spectrogram is first calculated. The short-time Fourier transform (STFT) shifted by

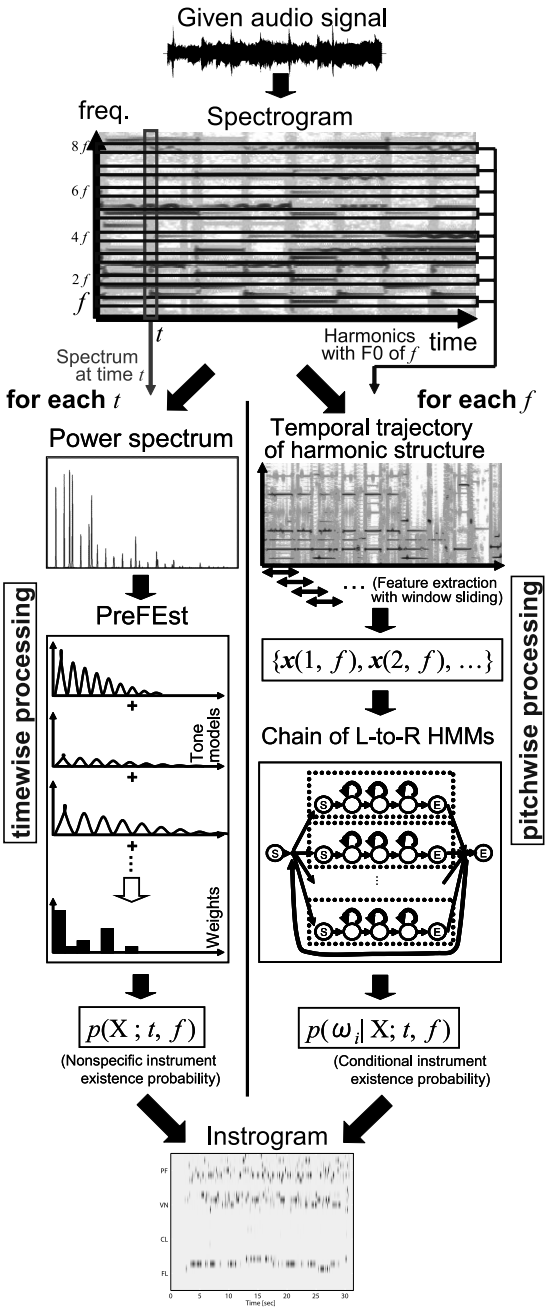


Fig. 3 Overview of our technique for calculating the instrogram.

10 ms (441 points at 44.1 kHz sampling) with an 8,192-point Hamming window is used in the current implementation. We next calculate the NIEPs and CIEPs. The NIEPs are calculated using the PreFEst [10]. The PreFEst models the spectrum of a signal containing multiple musical instrument sounds as a weighted mixture of harmonic-structure tone models at each frame.

For CIEPs, on the other hand, the temporal trajectory of the harmonic structure with every F0 is modeled with L-to-R HMMs because the temporal characteristics are important in recognizing instruments. Once the time series of feature vectors are obtained for every F0, the likelihoods of the paths in the chain of HMMs are calculated.

The advantage of this technique lies in the fact that $p(\omega_i; t, f)$ can be estimated robustly because the two constituent probabilities are calculated independently and are then integrated by multiplication. In most previous studies, the onset time and F0 of each note were first estimated, and then the instrument for the note was identified by analyzing spectral components extracted based on the results of the note estimation. The upper limit of the instrument identification performance was therefore bound by the precedent note estimation, which is generally difficult and not robust for polyphonic music ☆. Unlike such a notewise symbolic approach, our non-symbolic and non-sequential approach is more robust for polyphonic music.

3.2 Nonspecific Instrument Existence Probability

The NIEP $p(\mathrm{X}; t, f)$ is estimated by using the PreFEst on the basis of the maximum likelihood estimation without assuming the number of sound sources in a mixture. The PreFEst, which was originally developed for estimating F0s of melody and bass lines, consists of three processes: the *PreFEst-front-end* for frequency analysis, the *PreFEst-core* for estimating the relative dominance of every possible F0, and the *PreFEst-back-end* for evaluating the temporal continuity of the F0. Because the problem to be solved here is not the estimation of the predominant F0s as melody and bass lines, but rather the calculation of $p(\mathrm{X}; t, f)$ of every possible F0, we use only the PreFEst-core.

The PreFEst-core models an observed power spectrum as a weighted mixture of tone models $p(x|F)$ for every possible F0 F. The tone model $p(x|F)$, where x is the log frequency, represents a typical spectrum of harmonic structures, and the mixture density $p(x; \theta^{(t)})$ is defined as

$$p(x; \theta^{(t)}) = \int_{\mathrm{Fl}}^{\mathrm{Fh}} w^{(t)}(F) p(x|F) dF,$$

$$\theta^{(t)} = \{w^{(t)}(F) | \mathrm{Fl} \leq F \leq \mathrm{Fh}\},$$

where Fl and Fh denote the lower and upper limits, respectively, of the possible F0 range, and $w^{(t)}(F)$ is the weight of a tone model $p(x|F)$ that satisfies $\int_{\mathrm{Fl}}^{\mathrm{Fh}} w^{(t)}(F) dF = 1$. If we can estimate the model parameter $\theta^{(t)}$ such that the observed spectrum is likely to have been generated from $p(x; \theta^{(t)})$, the spectrum can be considered to be decomposed into harmonic-structure tone models and $w^{(t)}(F)$ can be interpreted as the relative predominance of the tone model with an F0 of F at time t. We can therefore calculate the NIEP $p(\mathrm{X}; t, f)$ as the weight $w^{(t)}(f)$, which can be estimated using the *Expectation-Maximization* (EM) algorithm [10]. In the current implementation, we use the tone model given by

$$p(x|F) = \alpha \sum_{h=1}^{\mathrm{N}} c(h) G(x; F + 1200 \log_2 h, \mathrm{W}),$$

$$G(x; m, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right),$$

where α is a normalizing factor, $\mathrm{N} = 16$, $\mathrm{W} = 17$ cent, and $c(h) = G(h; 1, 5.5)$. This tone model was also used in the earliest version of the PreFEst [11].

3.3 Conditional Instrument Existence Probability

The following steps are performed for every frequency f.

3.3.1 Harmonic Structure Extraction

The temporal trajectory $H(t, f)$ of the harmonic structure (10 harmonics) of which F0 is f is extracted.

3.3.2 Feature Extraction

It is important to design features that will effectively recognize musical instruments. Although mel-frequency cepstrum coefficents (MFCCs) and Delta MFCCs are commonly used in the field of speech recognition studies, we should design features that are optimized for musical instrument sounds because musical instrument sounds have complicated temporal variations (e.g., amplitude and frequency modulations). We therefore designed the 28 features listed in **Table 1** based on our previous stud-

☆ We tested robustness with respect to onset errors in identifying an instrument for every note using our previous method [9]. Giving errors following a normal distribution with a standard deviation of e [s] to onset times, we obtained the following results:

$e=0$	$e=0.05$	$e=0.10$	$e=0.15$	$e=0.20$
71.4%	69.2%	66.7%	62.5%	60.5%

Table 1 Overview of 28 features.

Spectral features	
1	Spectral centroid
2	Relative power of fundamental component
3–10	Relative cumulative power from fundamental to i-th components ($i = 2, 3, \cdots, 9$)
11	Relative power in odd and even components
12–20	Number of components haveing a duration that is p% longer than the longest duration ($p = 10, 20, \cdots, 90$)
Temporal features	
21	Gradient of straight line approximating power envelope
22–24	Temporal mean of differentials of power envelope from t to $t + iT/3$ ($i = 1, \cdots, 3$)
Modulation features	
25, 26	Amplitude and frequency of AM
27, 28	Amplitude and frequency of FM

ies [9]. For every time t (every 10 ms in the implementation), we first excerpt a T-length bit of the harmonic-structure trajectory $H_t(\tau, f)$ ($t \leq \tau < t + T$) from the entire trajectory $H(t, f)$ and then extract a feature vector $\boldsymbol{x}(t, f)$ consisting of the 28 features from $H_t(\tau, f)$. The dimensionality is then reduced to 12 dimensions using principal component analysis with a proportion value of 95%. T is 500 ms in the current implementation.

3.3.3 Probability Calculation

We train L-to-R HMMs, each consisting of 15 states ☆, for target instruments $\omega_1, \cdots, \omega_m$, and then basically consider the time series of feature vectors, $\{\boldsymbol{x}(t, f)\}$, to be generated from a Markov chain of these HMMs. Then, the CIEP $p(\omega_i|\mathrm{X}; t, f)$ is calculated as

$$p(M_i|\boldsymbol{x}(t, f)) = \frac{p(\boldsymbol{x}(t, f)|M_i)p(M_i)}{\sum_{i=1}^{m} p(\boldsymbol{x}(t, f)|M_i)p(M_i)},$$

where M_i is the HMM corresponding to the instrument ω_i. $p(\boldsymbol{x}(t, f)|M_i)$ is trained from data prepared in advance, and $p(M_i)$ is the *a priori* probability.

In the above formulation, $p(\omega_i|\mathrm{X}; t, f)$ for some instruments may become greater than zero even if no instruments are played. Theoretically, this does not matter because $p(\mathrm{X}; t, f)$ becomes zero in such cases. In practice, however, $p(\mathrm{X}; t, f)$ may not be zero, especially when a certain instrument is played at an F0 of an integer multiple or factor of f. To avoid this, we prepare an HMM, M_0, trained with feature vectors extracted from silent signals (note that some instruments may be played at non-target F0s) and consider $\{\boldsymbol{x}(t, f)\}$ to be generated from a Markov chain of the $m + 1$ HMMs $(M_0, M_1, \cdots, M_m)$. The CIEP is therefore calculated as

$$p(M_i|\boldsymbol{x}(t, f)) = \frac{p(\boldsymbol{x}(t, f)|M_i)p(M_i)}{\sum_{i=0}^{m} p(\boldsymbol{x}(t, f)|M_i)p(M_i)},$$

where we use $p(M_i) = 1/(m + 1)$.

3.4 Simplifying Instrograms

Although we calculate IEPs for every F0, some applications do not need such detailed results. If the instrogram is used for retrieving musical pieces that include a certain instrument's sounds, for example, IEPs for rough frequency regions (e.g., high, middle and low) are sufficient. We therefore divide the entire frequency region into N subregions $I_1, \cdots, I_N$ and calculate the IEP $p(\omega_i; t, I_k)$ for the k-th frequency subregion I_k. Here, this is defined as $p(\omega_i; t, \bigcup_{f \in I_k} f)$, which can be obtained by iteratively calculating the following equation because the frequency axis is practically discrete.

$$\begin{aligned} &p(\omega_i; t, f_1 \cup \cdots \cup f_i \cup f_{i+1}) \\ &= p(\omega_i; t, f_1 \cup \cdots \cup f_i) + p(\omega_i; t, f_{i+1}) \\ &\quad - p(\omega_i; t, f_1 \cup \cdots \cup f_i)\ p(\omega_i; t, f_{i+1}), \end{aligned}$$

where $I_k = \{f_1, \cdots, f_i, f_{i+1}, \cdots, f_{n_k}\}$.

4. Application

Here, we discuss the application of instrograms to MPEG-7 annotation and MIR.

4.1 MPEG-7 Annotation

There are two choices for transforming instrograms to MPEG-7 annotations. First, we can simply represent IEPs as a time series of vectors. Because the MPEG-7 standard has no tag for the instrogram annotation, we added several original tags, as shown in **Fig. 4**. This example shows the time series of eight-dimensional IEPs for the piano (line 16) with a time resolution of 10 ms (line 6). Each dimension corresponds to a different frequency subregion, which is defined by dividing the entire range from 65.5 Hz to 1,048 Hz (line 3) by 1/2 octave (line 4).

Second, we can transform instrograms into a symbolic representation. We also added several

☆ We used more states than those used in usual speech recognition studies (typically three) because the notes of musical instruments usually have longer durations than phonemes.

```
 1:<AudioDescriptor
 2:    xsi:type="AudioInstrogramType"
 3:    loEdge="65.5" hiEdge="1048"
 4:    octaveResolution="1/2">
 5:  <SeriesOfVector totalNumOfSamples="5982"
 6:      vectorSize="8" hopSize="PT10N1000F">
 7:    <Raw mpeg7:dim="5982 8">
 8:      0.0 0.0 0.0 0.0 0.718 0.017 0.051 0.0
 9:      0.0 0.0 0.0 0.0 0.724 0.000 0.085 0.0
10:      0.0 0.0 0.0 0.0 0.702 0.013 0.089 0.0
11:      0.0 0.0 0.0 0.0 0.661 0.017 0.063 0.0
12:                  ......
13:    </Raw>
14:  </SeriesOfVector>
15:  <SoundModel
16:     SoundModelRef="IDInstrument:Piano"/>
17:</AudioDescriptor>
```

Fig. 4 Excerpt of example of instrogram annotation.

```
 1:<MultimediaContent xsi:type="AudioType">
 2:  <Audio xsi:type="AudioSegmentType">
 3:    <MediaTime>
 4:      <MediaTimePoint>T00:00:06:850N1000
 5:                           </MediaTimePoint>
 6:      <MediaDuration>PT0S200N1000
 7:                           </MediaDuration>
 8:    </MediaTime>
 9:    <AudioDescriptor xsi:type="SoundSource"
10:        loEdge="92" hiEdge="130">
11:     <SoundModel
12:      SoundModelRef="IDInstrument:Piano"/>
13:    </AudioDescriptor>
14:  </Audio>
                  ......
```

Fig. 5 Excerpt of example of symbolic annotation.

original tags, as shown in **Fig. 5**. This example shows that an event for the piano (line 12) at a pitch between 92 and 130 Hz (line 10) occurs at 6.850 s (line 4) and continues for 0.200 s (line 6). To obtain this symbolic representation, we have to estimate the event occurrence and its duration within every frequency subregion I_k. We therefore obtain the time series of the instrument maximizing $p(\omega_i; t, I_k)$ and then consider it to be an output of a Markov chain, states of which are $\omega_0, \omega_1, \cdots, \omega_m$. (**Fig. 6**). The transition probabilities in the chain from a state to the same state, from non-silence states to the silence state, and from the silence state to non-silence states are greater than zero, and the other probabilities are zero. After obtaining the most likely path in the chain, we can estimate the occurrence and duration of an instrument ω_i from the transitions between the states ω_0 and ω_i. This method assumes that only one instrument is played at the same time in each frequency subregion. When multiple instruments are played in the same subregion at the same time, the most predominant instrument will be annotated.

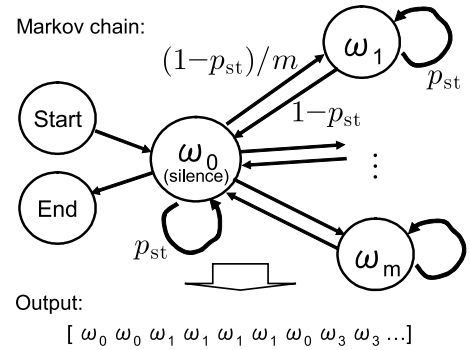


Fig. 6 Markov chain model used in symbolic annotation. The values are transition probabilities, where $p_{\rm st}$ is the probability of staying at the same state at the next time, which was experimentally determined as $1 - 10^{-16}$.

4.2 Music Information Retrieval based on Instrumentation Similarity

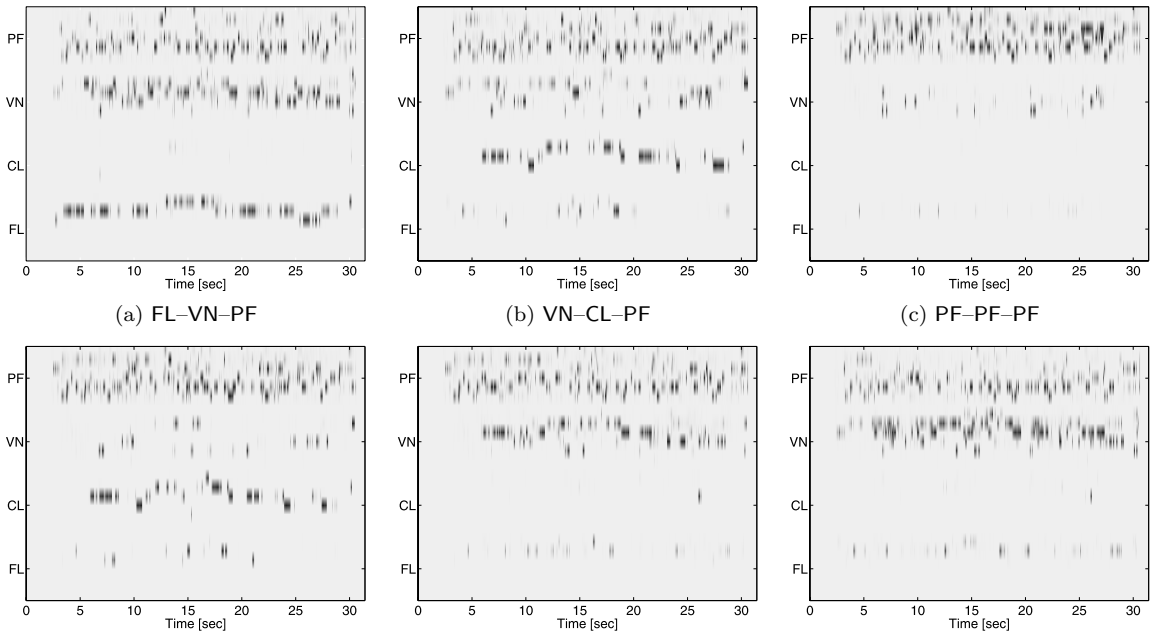
One of the advantages of the instrogram, which is a non-symbolic representation, is to provide a new instrumentation-based similarity measure. The similarity between two instrograms enables the MIR based on instrumentation similarity. As we pointed out in the Introduction, this key technology is important for automatic playlist generation and content-based music recommendation. Here, instead of calculating similarity, we calculate the distance (dissimilarity) between instrograms by using DTW as follows:

(1) A vector $\boldsymbol{p}_t$ for every time t is obtained by concatenating the IEPs of all instruments:
$$\boldsymbol{p}_t = (p(\omega_1; t, I_1), p(\omega_1; t, I_2), \cdots, p(\omega_m; t, I_N))',$$
where $'$ is the transposition operator.

(2) The distance between two vectors, $\boldsymbol{p}$ and $\boldsymbol{q}$, is defined as the cosine distance:
$$\mathrm{dist}(\boldsymbol{p}, \boldsymbol{q}) = 1 - (\boldsymbol{p}, \boldsymbol{q})/||\boldsymbol{p}|| \cdot ||\boldsymbol{q}||,$$
where $(\boldsymbol{p}, \boldsymbol{q}) = \boldsymbol{p}' R \boldsymbol{q}$, and $||\boldsymbol{p}|| = \sqrt{(\boldsymbol{p}, \boldsymbol{p})}$. $R = (r_{ij})$ is a positive definite symmetric matrix. By setting r_{ij} with respect to related elements, e.g., violin vs. viola, to a value greater than zero, one can consider the similarity between such related instruments. When R is the unit matrix, $(\boldsymbol{p}, \boldsymbol{q})$ and $||\boldsymbol{p}||$ are equivalent to the standard inner product and norm.

(3) The distance (dissimilarity) between $\{\boldsymbol{p}_t\}$ and $\{\boldsymbol{q}_t\}$ is calculated by applying DTW with the above-mentioned distance measure.

The timbral similarity was also used in previous MIR-related studies [12),13)]. The timbral similarity was calculated on the basis of spectral features, such as MFCCs, directly extracted from



(a) FL–VN–PF (b) VN–CL–PF (c) PF–PF–PF

(d) PF–CL–PF (e) PF–VN–PF (f) VN–VN–PF

Fig. 7 Results of calculating instrograms from "Auld Lang Syne" with six different instrumentations. "FL–VN–PF" means that the treble, middle, and bass parts are played on flute, violin, and piano, respectively.

complex sound mixtures. Such features sometimes do not clearly reflect actual instrumentation, as will be implied in the next section, because they are influenced not only by instrument timbres but also by arrangements, including the voicing of chords. On the other hand, because instrograms directly represent instrumentation, this will facilitate the appropriate calculation of the similarity of instrumentation. Moreover, instrograms have the following advantages:

Intuitiveness The musical meaning is intuitively clear.

Controllability By appropriately setting R, users can ignore the differences between pitch regions within the same instrument and/or the difference between instruments within the same instrument family.

5. Experiments

We conducted experiments on obtaining instrograms and symbolic annotations for both audio data generated on a computer and the recordings of real performances. In addition, we tested the calculation of the similarity between instrograms for the real performances.

5.1 Use of Generated Audio Data

We first conducted experiments on obtaining instrograms from audio signals of trio music "Auld Lang Syne" used by Kashino *et al*[3]. The audio signals were generated by mixing audio data from RWC-MDB-I-2001[14] (Variation No. 1) according to a standard MIDI file (SMF) that we input using a MIDI sequencer based on Kashino's score. The target instruments were the piano (PF), violin (VN), clarinet (CL), and flute (FL). The training data for these instruments were taken from the audio data in RWC-MDB-I-2001 with Variation Nos. 2 and 3. The time resolution was 10 ms, and the frequency resolution was every 100 cent. The width of each frequency subregion for the simplification was 600 cent. We used HTK 3.0 for HMMs.

The results are shown in **Fig. 7**. When we compare (a) and (b), (a) has high IEPs for the flute in high-frequency regions while (b) has very low (almost zero) IEPs. In contrast, (a) has very low (almost zero) IEPs for the clarinet and (b) has high IEPs. Also, (d) has high IEPs for the clarinet and almost zero IEPs for the violin whereas (e) has high IEPs for the violin and almost zero IEPs for the clarinet. In the case of (c), the IEPs only for the piano are sufficiently high. Although both (e) and (f) are played on the piano and violin, the IEPs for the violin in the highest frequency region are

different. This correctly reflects the difference between the actual instrumentations.

Based on the instrograms obtained above, we conducted experiments on symbolic annotation using the method described in Section 4.1. We first prepared ground truth (correct data) from the SMF used to generate the audio signals and then evaluated the results based on the recall rate R and precision rate P given by

$$R = \frac{\sum_{i=1}^{m}\sum_{k=1}^{N}\begin{pmatrix}\#\text{ frames correctly}\\ \text{annotated as } \omega_i \text{ at } I_k\end{pmatrix}}{\sum_{i=1}^{m}\sum_{k=1}^{N}\begin{pmatrix}\#\text{ frames that should be}\\ \text{annotated as } \omega_i \text{ at } I_k\end{pmatrix}},$$

$$P = \frac{\sum_{i=1}^{m}\sum_{k=1}^{N}\begin{pmatrix}\#\text{ frames correctly}\\ \text{annotated as } \omega_i \text{ at } I_k\end{pmatrix}}{\sum_{i=1}^{m}\sum_{k=1}^{N}\begin{pmatrix}\#\text{ frames annotated}\\ \text{as } \omega_i \text{ at } I_k\end{pmatrix}}.$$

The results are shown in **Table 2**. We achieved a precision rate of 78.7% on average. Although the recall rates were not high (14–38%), we consider the precision rates to be more important than the recall rates for MIR; a system can use recognition results even if some frames or frequency subregions are missing, whereas false results have a negative influence on MIR.

We also evaluated symbolic annotation by merging all the frequency subregions; in other words, we ignored the differences between frequency subregions. This was because instrument annotation is useful even without F0 information for MIR. For example, a task such as searching for piano solo pieces can be achieved without F0 information. The evaluation was conducted based on the recall rate R' and precision rate P'. The recall and precision rates for this evaluation are given by

$$R' = \frac{\sum_i(\#\text{frames correctly annotated as } \omega_i)}{\sum_i(\#\text{frames that should be annotated as } \omega_i)},$$

$$P' = \frac{\sum_i(\#\text{frames correctly annotated as } \omega_i)}{\sum_i(\#\text{frames annotated as } \omega_i\ )}.$$

The results are listed in **Table 3**. The average precision rate was 87.5% and the maximum was 95.4% for FL–VN–PF. The precision rates for all pieces were over 80%, while the recall rates were approximately between 30 and 60%.

5.2 Use of Real Performances

We next conducted experiments on obtaining instrograms from the recordings of real performances of classical and jazz music taken from the RWC Music Database [15]. The instrumentation of all pieces is listed in **Table 4**. We only used the first one-minute signal for each piece. The experimental conditions were basically the same as those in Section 5.1. Because the target instruments were the piano, violin, clarinet, and flute, the IEPs for the violin should also be high when string instruments other than the violin are played, and the IEPs for the clarinet should always be low. The training data were taken from both RWC-MDB-I-2001 [14] and NTTMSA-P1 (a non-public musical sound database) ☆.

The results, shown in **Fig. 8**, show that (a) and (b) have high IEPs for the violin while

Table 2 Results of symbolic annotation for "Auld Lang Syne."

	Recall	Precision
FL–CL–PF	28.7%	63.4%
FL–PF–PF	38.5%	89.4%
FL–VN–PF	37.2%	89.5%
PF–CL–PF	22.2%	79.3%
PF–PF–PF	26.0%	93.5%
PF–VN–PF	24.2%	76.6%
VN–CL–PF	21.4%	63.6%
VN–PF–PF	14.3%	76.1%
VN–VN–PF	30.2%	76.9%
Average	27.0%	78.7%

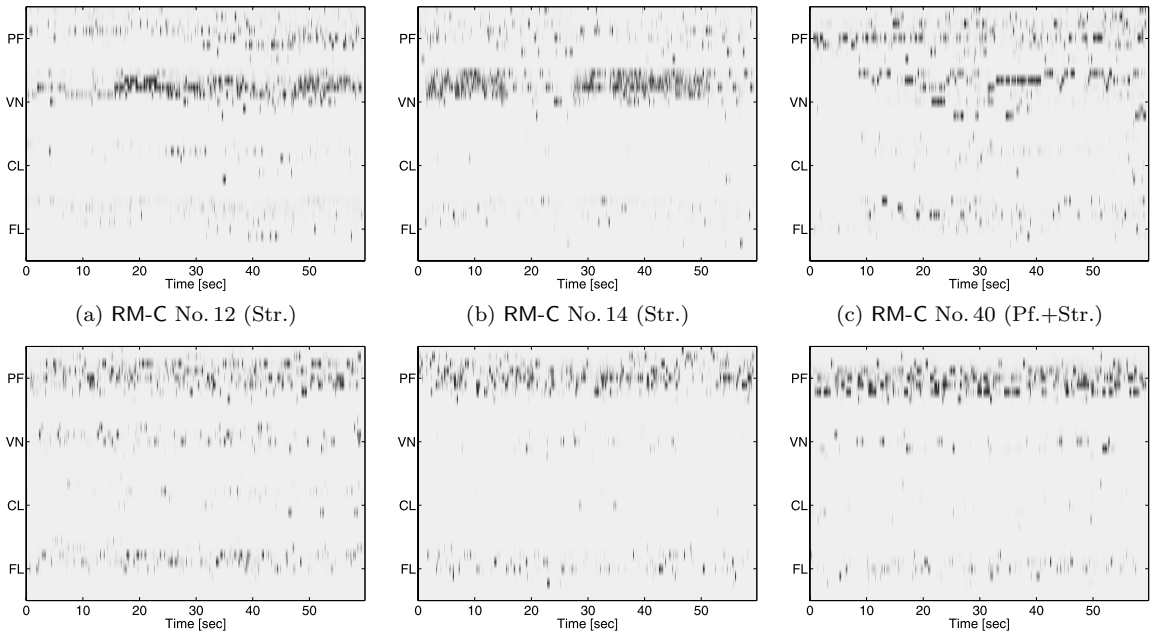
Table 3 Results of symbolic annotation for "Auld Lang Syne" (all frequency subregions merged).

	Recall	Precision
FL–CL–PF	36.0%	80.3%
FL–PF–PF	56.8%	87.6%
FL–VN–PF	44.5%	95.4%
PF–CL–PF	40.5%	84.4%
PF–PF–PF	62.2%	91.4%
PF–VN–PF	40.5%	88.1%
VN–CL–PF	29.2%	87.6%
VN–PF–PF	34.9%	86.7%
VN–VN–PF	40.8%	85.3%
Average	42.8%	87.5%

Table 4 Musical pieces used and their instrumentations.

Classical	(i) No. 12, 14, 21, 38	Strings
	(ii) No. 19, 40	Piano+Strings
	(iii) No. 43	Piano+Flute
Jazz	(iv) No. 1, 2, 3	Piano solo

☆ The database called NTTMSA-P1 consists of isolated monophonic tones played by two different individuals for each instrument. Every semitone over the pitch range is played with three different intensities for each instrument.



(a) RM-C No. 12 (Str.) (b) RM-C No. 14 (Str.) (c) RM-C No. 40 (Pf.+Str.)

(d) RM-C No. 43 (Pf.+Fl.) (e) RM-J No. 1 (Pf.) (f) RM-J No. 2 (Pf.)

Fig. 8 Results of calculating instrograms from audio signals of real performances. RM-C stands for RWC-MDB-C-2001 and RM-J stands for RWC-MDB-J-2001.

(e) and (f) have high IEPs for the piano. For (c), the IEPs for the violin increase after 10 sec, whereas those for the piano are initially high. This reflects the actual performances of these instruments. When (d) is compared to (e) and (f), the former has slightly higher IEPs for the flute than the latter, although the difference is unclear. In general, the IEPs are not as clear as those for signals generated by copy-and-pasting waveforms of RWC-MDB-I-2001. This is because the acoustic characteristics of real performances have greater variety. This could be improved by adding appropriate training data.

We also evaluated the symbolic annotation for these real-performance recordings. The evaluation was only conducted for the case in which all frequency subregions were merged because it is difficult to manually prepare a reliable ground truth for each frequency subregion ☆. The results are listed in **Table 5**. The average precision rate was 69.4% and the maximum was 84.3%, which were lower than those for synthesized music. This would also be because of the great variety in the acoustic characteristics of real performances. The recall rates, in contrast, were higher than those for synthesized music because the same instrument was often simultaneously played over multiple frequency subregions, in which the instrument was regarded as correctly recognized if it was recognized in any of these subregions.

☆ Although the SMF corresponding to each piece is available in the RWC Music Database, it cannot be used because the SMF and audio signal are not synchronized.

Table 5 Results of symbolic annotation for real recordings (all frequency subregions merged).

		Recall	Precision
RM-C	No. 12	78.0%	63.4%
	14	76.0%	74.0%
	19	45.1%	65.6%
	21	89.9%	70.0%
	38	65.1%	64.0%
	40	50.8%	71.5%
	43	49.7%	84.3%
RM-J	No. 1	62.1%	72.0%
	2	75.6%	69.3%
	3	45.9%	59.7%
Average		63.8%	69.4%

5.3 Similarity Calculation

We tested the calculation of the dissimilarities between instrograms. The results, listed in **Table 6** (a), can be summarized as follows:

- The dissimilarities within each group were generally less than 7,000 (except Group (ii)).
- Those between Groups (i) (played on strings) and (iv) (piano) were generally

Table 6 Dissimilarities in Instrumentation between musical pieces.

(a) Using IEPs (instrograms)

	(i)				(ii)		(iii)	(iv)			3-best-similarity
	C12	C14	C21	C38	C19	C40	C43	J01	J02	J03	pieces
C12	0										C21, C14, C38
C14	6429	0									C21, C12, C38
C21	5756	5734	0								C14, C12, C38
C38	7073	6553	6411	0							C21, C14, C38
C19	7320	8181	7274	7993	0						C21, C12, C38
C40	8650	8353	8430	8290	8430	0					J02, J01, C43
C43	8910	9635	9495	9729	8148	8235	0				J01, J02, J03
J01	9711	10226	10252	10324	8305	8214	6934	0			J02, J03, C43
J02	9856	10125	10033	10610	8228	8139	7216	6397	0		J01, C43, J03
J03	9134	9136	8894	9376	8058	8327	7480	6911	7223	0	J01, J02, C43

(b) Using MFCCs

	(i)				(ii)		(iii)	(iv)			3-best-similarity
	C12	C14	C21	C38	C19	C40	C43	J01	J02	J03	pieces
C12	0										C21, C40, J02
C14	17733	0									C43, C12, J02
C21	17194	18134	0								C12, J01, J02
C38	18500	18426	18061	0							J01, J02, C21
C19	17510	18759	18222	19009	0						J02, C12, J03
C40	17417	19011	18189	19099	18100	0					C12, J02, J01
C43	18338	17459	17728	18098	18746	18456	0				J01, C14, J02
J01	17657	17791	17284	17834	18133	17983	16762	0			J02, C43, J03
J02	17484	17776	17359	18009	17415	17524	17585	15870	0		J01, J03, C21
J03	17799	18063	17591	18135	17814	18038	17792	16828	16987	0	J01, J02, C21

Note: "C" and "12" of "C12", for example, represents a database (Classical/Jazz) and a piece number, respectively.

greater than 9,000, and some were greater than 10,000.

- Those between Groups (i) and (iii) (piano+flute) were also around 9,000.
- Those between Groups (i) and (ii) (piano+strings), (ii) and (iii), and (ii) and (iv) were around 8,000. As one instrument is commonly used in these pairs, these dissimilarities were reasonable.
- Those between Groups (iii) and (iv) were around 7,000. Because the difference between these groups is only the presence of the flute, these were also reasonable.

For comparison, Table 6 (b) lists the results obtained using MFCCs. The 12-dimensional MFCCs were extracted every 10 ms with a 25-ms Hamming window. No Delta MFCCs were used. After the MFCCs were extracted, the dissimilarity was calculated using the method described in Section 4.2, where $\{\boldsymbol{p}_t\}$ was a sequence of 12-dimensional MFCC vectors instead of IEP vectors. Comparing the results with the two methods, we can see the following differences:

- The dissimilarities within Group (i) and the dissimilarities between Group (i) and the others for IEPs differed more than those for MFCCs. In fact, all the three-best-similarity pieces from those in Group (i) belonged to the same group, i.e., (i), for IEPs, while those for MFCCs contained pieces out of Group (i).
- None of the three-best-similarity pieces from the four pieces without strings (Groups (iii) and (iv)) contained strings for IEPs, whereas those for MFCCs contained pieces with strings (C14, C21).

We also developed a simple prototype system that searches pieces that have similar instrumentation to that specified by the user. The demonstration of our MIR system and other materials will be available at: http://winnie.kuis.kyoto-u.ac.jp/~kitahara/instrogram/IPSJ07/.

6. Discussion

This study makes three major contributions to instrument recognition and MIR.

The first is the formulation of instrument recognition as the calculation of NIEPs and CIEPs. Because the calculation of NIEPs includes a process that can be considered to be an alternative to the estimation of onset times and F0s, this formulation has made it possible to omit their explicit estimation, which is difficult for polyphonic music. Based on similar motivations, Vincent, et al. [7] and Essid, et

al. [8] proposed new instrument recognition techniques. Vincent, et al.'s technique involves both transcription and instrument identification in a single optimization procedure. This technique is based on a reasonable formulation and is probably effective but has only been tested on solo and duo excerpts. Essid, et al.'s technique identifies the instrumentation, instead of the instrument for each part, from a pre-designed possible-instrumentation list. This technique is based on the standpoint that music usually has one of several typical instrumentations. They reported successful experimental results, but identifying instrumentations other than those prepared is impossible. Our instrogram technique, in contrast, has made it possible to recognize instrumentation without making any assumptions about instrumentation for audio data, including synthesized music and real performances that have various instrumentations.

The second contribution is the establishment of a graphical representation of instrumentation. Previous studies on music visualization were generally based on MIDI-level symbolic information [16),17)]. Although spectrograms and specmurts [18)] are useful for visualization of audio signals, it is difficult to recognize instrumentation from them. Our instrogram representation provides visual information on instrumentation, which should be useful for generating music thumbnails and other visualizations such as animations for entertainment.

The third contribution is the achievement of an MIR based on instrumentation similarity. Although both timbral similarity calculation and instrument recognition have been actively investigated, no attempts of calculating the instrumentation similarity on the basis of instrument recognition techniques have been made, because previous instrument recognition have aimed to determine the instruments that are played in given signals. The instrogram, which represents instrumentation as a set of continuous values, is an effective approach to designing a continous similarity measure.

7. Conclusion

We described a new *instrogram* representation obtained by using a new musical instrument recognition technique that explicitly uses neither onset detection nor F0 estimation. Whereas most previous studies first estimated the onset time and F0 of each note and then identified the instrument for each note, our technique calculates the instrument existence probability for each target instrument at each point on the time-frequency plane. This non-note-based approach made it possible to avoid adverse influences caused by errors of the onset detection and F0 estimation.

In addition, we presented methods for applying the instrogram technique to MPEG-7 annotation and the MIR based on instrumentation similarity. Although the note-based outputs of previous instrument recognition were difficult to apply to continuous similarity calculation, our instrogram representation provided a measure for similarities in instrumentation and achieved MIR based on this measure.

The instrogram is related to Goto's study on music understanding based on the idea that people understand music without mentally representing it as a score [19)]. Goto claims that good music descriptors should be musically intuitive, fundamental to a professional method of understanding music, and useful for various applications. We believe that the instrogram satisfies these requirements and therefore intend to apply it to the professional score-based music understanding as well as various applications including MIR.

Acknowledgments This research was supported in part by the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Grant-in-Aid for Scientific Research and Informatics Research Center for Development of Knowledge Society Infrastructure (COE program of MEXT, Japan). We used the RWC Music Database and NTTMSA-P1. We would like to thank everyone who has contributed to building these databases and wish to thank NTT Communication Science Laboratories for giving us permission to use NTTMSA-P1. The experiment presented in the appendix was conducted by Mr. Katsutoshi Itoyama. We appreciate his cooperation. We would also like to thank Dr. Shinji Watanabe (NTT) for his valuable comments.

References

1) Martin, K.D.: Sound-Source Recognition: A Theory and Computational Model, PhD Thesis, MIT (1999).
2) Kashino, K., Nakadai, K., Kinoshita, T. and Tanaka, H.: Application of the Bayesian Probability Network to Music Scene Analysis, *Computational Auditory Scene Analysis*, Rosenthal, D.F. and Okuno, H.G. (Eds.), pp.115–137,

Lawrence Erlbaum Associates (1998).
3) Kashino, K. and Murase, H.: A Sound Source Identification System for Ensemble Music based on Template Adaptation and Music Stream Extraction, *Speech Comm.*, Vol.27, pp.337–349 (1999).
4) Kinoshita, T., Sakai, S. and Tanaka, H.: Musical Sound Source Identification based on Frequency Component Adaptation, *Proc. IJCAI CASA Workshop*, pp.18–24 (1999).
5) Eggink, J. and Brown, G.J.: Application of Missing Feature Theory to the Recognition of Musical Instruments in Polyphonic Audio, *Proc. ISMIR* (2003).
6) Eggink, J. and Brown, G.J.: Extracting Melody Lines from Complex Audio, *Proc. ISMIR*, pp.84–91 (2004).
7) Vincent, E. and Rodet, X.: Instrument Identification in Solo and Ensemble Music using Independent Subspace Analysis, *Proc. ISMIR*, pp.576–581 (2004).
8) Essid, S., Richard, G. and David, B.: Instrument Recognition in Polyphonic Music, *Proc. ICASSP*, Vol.III, pp.245–248 (2005).
9) Kitahara, T., Goto, M., Komatani, K., Ogata, T. and Okuno, H.G.: Instrument Identification in Polyphonic Music: Feature Weighting with Mixed Sounds, Pitch-dependent Timbre Modeling, and Use of Musical Context, *Proc. ISMIR*, pp.558–563 (2005).
10) Goto, M.: A Real-time Music-scene-description System: Predominant-F0 Estimation for Detecting Melody and Bass Lines in Real-world Audio Signals, *Speech Comm.*, Vol.43, No.4, pp.311–329 (2004).
11) Goto, M.: A Robust Predominant-F0 Estimation Method for Real-time Detection of Melody and Bass Lines in CD Recordings, *Proc. ICASSP*, Vol.II, pp.757–760 (2000).
12) Tzanetakis, G. and Cook, P.: Musical Genre Classification of Audio Signals, *IEEE Trans. Speech Audio Process.*, Vol.10, No.5, pp.293–302 (2002).
13) Aucouturier, J.-J. and Pachet, F.: Music Similarity Measure: What's the Use?, *Proc. ISMIR*, pp.157–163 (2002).
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., Hashiguchi, H., Nishimura, T. and Oka, R.: RWC Music Database: Popular, Classical, and Jazz Music Databases, *Proc. ISMIR*, pp.287–288 (2002).
16) Hiraga, R.: A Look of Performance, *IEEE Visualization* (2002).
17) Hiraga, R., Miyazaki, R. and Fujishiro, I.: Performance Visualization—A New Challenge to Music through Visualization, *ACM Multimedia* (2002).
18) Sagayama, S., Takahashi, K., Kameoka, H. and Nishimoto, T.: Specmurt Anasylis: A Piano-roll-visualization of Polyphonic Music by Deconvolution of Log-frequency Spectrum, *Proc. SAPA* (2004).
19) Goto, M.: Music Scene Description Project: Toward Audio-based Real-time Music Understanding, *Proc. ISMIR*, pp.231–232 (2003).
20) Kameoka, H., Nishimoto, T. and Sagayama, S.: Harmonic-temporal-structured Clustering via Deterministic Annealing EM Algorithm for Audio Feature Extraction, *Proc. ISMIR*, pp.115–122 (2005).

Appendix

A.1 Experiment on Notewise Instrument Recognition

This Appendix presents the results of an experiment on notewise instrument recognition. The problem that we deal with here is to detect a note sequence played on a specified instrument. We first estimate the onset time and F0 of every note using harmonic-temporal-structured clustering (HTC) [20]. For every note, we then calculate the likelihood that the note would be played on the specified instrument using our previous instrument identification method [9]. Finally, we select the note that has the maximal likelihood every time. Note that we assume that multiple notes are not simultaneously played on the specified instrument. We used "Auld Lang Syne" played on the piano, violin, and flute as a test sample. This was synthesized by mixing audio data from the RWC-MDB-I-2001 [14] similarly to the experiment reported in Section 5.1. Only if a detected note is actually played on the specified instrument and at the correct F0 (note number) and the error of the estimated onset time is less than e [s], the note is judged to be correct. Recognition was evaluated through the following:

$$R = \frac{\#\text{ correctly detected notes}}{\#\text{ notes that should be detected}},$$

$$P = \frac{\#\text{ correctly detected notes}}{\#\text{ detected notes}}.$$

The results are shown in **Table 7**. Recognition especially for the violin and flute was not sufficiently accurate. Although the results of this experiment cannot be directly compared with

Table 7 Results of experiment on notewise approach.

	$e = 0.2$			$e = 0.4$			$e = 0.6$		
	PF	VN	FL	PF	VN	FL	PF	VN	FL
R	59%	36%	31%	59%	42%	34%	60%	47%	48%
P	86%	26%	17%	86%	24%	19%	87%	27%	27%

those obtained using our approach because they were evaluated in different ways, we can see that the notewise approach is not robust.

(Received May 8, 2006)
(Accepted October 3, 2006)
(Online version of this article can be found in the IPSJ Digital Courier, Vol.3, pp.1–13.)

Tetsuro Kitahara received his B.S. from the Tokyo University of Science in 2002 and his M.S. from Kyoto University in 2004. He is currently a Ph.D. student at Graduate School of Informatics of Kyoto University. He has been a Research Fellow of the Japan Society for the Promotion of Science since 2005. His research interests include music informatics. He has received several awards including the TELECOM System Technology Award for Student and the IPSJ 67th National Convention Best Paper Award for Young Researcher.

Masataka Goto received his Doctor of Engineering degree in electronics, information, and communication engineering from Waseda University, Japan, in 1998. He is currently a Senior Research Scientist of the National Institute of Advanced Industrial Science and Technology (AIST). He served concurrently as a Researcher in Precursory Research for Embryonic Science and Technology (PRESTO), Japan Science and Technology Corporation (JST) from 2000 to 2003, and has been an Associate Professor of the Department of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, University of Tsukuba, since 2005. He has received 18 awards, including the IPSJ Best Paper Award, IPSJ Yamashita SIG Research Awards, and Interaction 2003 Best Paper Award.

Kazunori Komatani received his B.E., M.S. and Ph.D. degrees from Kyoto University, Japan, in 1998, 2000, and 2002. He is currently an Assistant Professor at the Graduate School of Informatics of Kyoto University, Japan. He received the 2002 FIT Young Researcher Award and the 2004 IPSJ Yamashita SIG Research Award, both from the Information Processing Society of Japan. His research interests include spoken dialogue systems.

Tetsuya Ogata received his B.E., M.S. and Ph.D. degrees from Waseda University, Japan, in 1993, 1995, and 2000. He is currently an Associate Professor at the Graduate School of Informatics of Kyoto University, Japan. He served concurrently as a Visiting Associate Professor of the Humanoid Robotics Institute of Waseda University, and Vising Scientist of the Brain Science Institute of RIKEN. His research interests include multi-modal active sensing and robot imitation.

Hiroshi G. Okuno received his B.A. and Ph.D. from the University of Tokyo in 1972 and 1996. He worked for NTT, JST and the Tokyo University of Science. He is currently a professor of in the Graduate School of Informatics at Kyoto University. He is currently engaged in computational auditory scene analysis, music scene analysis and robot audition. He has received various awards including the 1990 Best Paper Award of JSAI, the Best Paper Award of IEA/AIE-2001 and 2005, and IEEE/RSJ Nakamura Award for IROS-2001 Best Paper Nomination Finalist. He was also awarded 2003 Funai Information Science Achievement Award. He edited with David Rosenthal "Computational Auditory Scene Analysis" (LEA, 1998) and with Taiichi Yuasa "Advanced Lisp Technology" (T&F, 2002).