Computational Musical Instrument Recognition and Its Application to Content-based Music Information Retrieval

Tetsuro KITAHARA

Abstract

The current capability of computers to recognize auditory events is severely limited when compared to human ability. Although computers can accurately recognize sounds that are sufficiently close to those trained in advance and that occur without other sounds simultaneously, they break down whenever the inputs are degraded by competing sounds.

In this thesis, we address computational recognition of non-percussive musical instruments in polyphonic music. Music is a good domain for computational recognition of auditory events because multiple instruments are usually played simultaneously. The difficulty in handling music resides in the fact that signals (events to be recognized) and noises (events to be ignored) are not uniquely defined. This is the main difference from studies of speech recognition under noisy environments. Musical instrument recognition is also important from an industrial standpoint. The recent development of digital audio and network technologies has enabled us to handle a tremendous number of musical pieces and therefore efficient music information retrieval (MIR) is required. Musical instrument recognition will serve as one of the key technologies for sophisticated MIR because the types of instruments played characterize musical pieces; some musical forms, in fact, are based on instruments, for example "piano sonata" and "string quartet."

Despite the importance of musical instrument recognition, studies have until recently mainly dealt with monophonic sounds. Although the number of studies dealing with polyphonic music has been increasing, their techniques have not yet achieved a sufficient level to be applied to MIR or other real applications. We investigate musical instrument recognition in two stages. At the first stage, we address instrument recognition for monophonic sounds to develop basic technologies for handling musical instrument sounds. In instrument recognition for monophonic sounds, we deal with two issues: (1) the pitch dependency of timbre and (2) the input of non-registered instruments. Because musical instrument sounds have wide pitch ranges in contrast to other kinds of sounds, the pitch dependency of timbre is an important issue. The second issue, that is, handling

instruments that are not contained in training data, is also an inevitable problem. This is because it is impossible in practice to build a thorough training data set due to a virtually infinite number of instruments. At the second stage, we address instrument recognition in polyphonic music. To deal with polyphonic music, we must solve the following two issues: (3) the overlapping of simultaneously played notes and (4) the unreliability of precedent note estimation process. When multiple instruments simultaneously play, partials (harmonic components) of their sounds overlap and interfere. This makes the acoustic features different from those of monophonic sounds. The overlapping of simultaneous notes is therefore an essential problem for polyphonic music. In addition, note estimation, that is, estimating the onset time and fundamental frequency (F0) of each note, is usually used as a preprocess in a typical instrument recognition framework. It remains, however, a challenging problem for polyphonic music.

In Chapter 3, we propose an *F0-dependent multivariate normal distribution* to resolve the first issue. The F0-dependent multivariate normal distribution is an extension of a multivariate normal distribution where the mean vector is defined as a function of F0. The key idea behind this is to approximate variation of each acoustic feature from pitch to pitch as a function of F0. This approximation makes it possible to separately model the pitch and non-pitch dependencies of timbres. We also investigate acoustic features for musical instrument recognition in this chapter. Experimental results with 6,247 solo tones of 19 instruments showed improvement of the recognition rate from 75.73% to 79.73% on average.

In Chapter 4, we solve the second issue by recognizing non-registered instruments at the category level. When a given sound is registered, its instrument name, *e.g.* violin, is recognized. Even if it is not registered, its category name, *e.g.* strings, can be recognized. The important issue in achieving such recognition is to adopt a musical instrument taxonomy that reflects acoustical similarity. We present a method for acquiring such a taxonomy by applying hierarchical clustering to a large-scale musical instrument sound database. Experimental results showed that around 77% of non-registered instrument sounds, on average, were correctly recognized at the category level.

In Chapter 5, we tackle the third issue by weighting features based on how much they are affected by overlapping; that is, we give lower weights to features affected more and higher weights to features affected less. For this kind of weighting, we have to evaluate the influence of the overlapping on each feature. It was, however, impossible in previous

studies to evaluate the influence by analyzing training data because the training data were only taken from monophonic sounds. Taking training data from polyphonic music (called a *mixed-sound template*), we evaluate the influence as the ratio of the within-class variance to the between-class variance in the distribution of the training data. We then generate feature axes using a weighted mixture that minimizes the influence by means of linear discriminant analysis. We also introduced musical context to avoid musically unnatural errors (*e.g.*, only one clarinet note within a sequence of flute notes). Experimental results showed that the recognition rates obtained using the above were 84.1% for duo music, 77.6% for trio music, and 72.3% for quartet music.

In Chapter 6, we describe a new framework of musical instrument recognition to solve the fourth issue. We formulate musical instrument recognition as the problem of calculating *instrument existence probabilities* at every point on the time-frequency plane. The instrument existence probabilities are calculated by multiplying two kinds of probabilities, one of which is calculated using the PreFEst and the other of which is calculated using hidden Markov models. The instrument existence probabilities are visualized in the spectrogram-like graphical representation called the *instrogram*. Because the calculation is performed for each time and each frequency, not for each note, estimation of the onset time and F0 of each note is not necessary. We obtained promising results for both synthesized music and recordings of real performances of classical and jazz music.

In Chapter 7, we describe an application of the instrogram analysis to similarity-based MIR. Because most previous similarity-based MIR systems used low-level features such as MFCCs, similarities for musical elements such as the melody, rhythm, harmony, and instrumentation could not be separately measured. As the first step toward measuring such music similarity, we develop a music similarity measure that reflects instrumentation only based on the instrogram representation. We confirmed that the instrogram can be applied to content-based MIR by developing a prototype system that searches for musical pieces that have instrumentation similar to that specified by the user.

In Chapter 8, we discuss the major contributions of this study toward research fields including computational auditory scene analysis, content-based MIR, and music visualization. We also discuss remaining issues and future directions of research.

Finally, we present our conclusions of this work in Chapter 9.

Acknowledgments

This work was carried out at Okuno Laboratory, Graduate School of Informatics, Kyoto University. I wish to express my gratitude to everyone who has supported this work.

First and foremost, I would like to thank Professor Hiroshi G. Okuno for his thoughtful supervision throughout my studies at Tokyo University of Science (TUS) and Kyoto University. He respects initiative of individual students and allowed me to freely choose research subjects based on my own interests. His various advice, ranging from general advice on how research should be conducted to advice regarding specific problems in my research, has formed the foundations of my research style.

Another person who has greatly supported my research is Dr. Masataka Goto (AIST). His profound and meticulous guidance helped me in all aspects of the development of my research. His guidance covered a broad range of topics, from signal processing techniques to how to write technical articles. Without their supervision, I could not have completed my work.

Professors Tatsuya Kawahara and Tetsuya Ogata provided me with essential and insightful comments as co-advisors. Furthermore, Professor Kawahara supervised me from 2002 to 2003 and Professor Ogata has supervised me since 2003 as Associate Professors at Okuno Laboratory. Professor Kawahara is one of the most famous researchers in the field of speech recognition. His comments as a specialist in audio signal processing and pattern recognition consistently helped me improve the technical aspects of my research. Professor Ogata is a famous researcher in robotics. Although his research field is different from that of this thesis, his comments from the broader perspective made discussions of the significance of this research more comprehensive. His stance of emphasizing his own philosophy about robotics has also influenced my research stance.

This work has also been supported by other many people through discussions at academic conferences. Professor Haruhiro Katayose (Kwansei Gakuin University), Dr. Kunio Kashino (NTT), and Dr. Kazuhiro Nakadai (Honda Research Institute Japan Co., Ltd.)

gave me valuable comments as pioneering researchers on automatic music transcription. Many researchers who are engaged in automatic speech recognition and sound source separation, including Dr. Tomohiro Nakatani, Dr. Takafumi Hikichi, and Dr. Shinji Watanabe (NTT), advised me based on their professional research experience. Mr. Hideki Asoh (AIST) gave me various comments on machine learning and pattern recognition techniques. Mr. Hirokazu Kameoka (the University of Tokyo) has greatly influenced my research stance, in particular, on the statistical modeling of auditory phenomena. Discussion with him always provided me with important suggestions and enabled me to keep my motivation high. Discussion with other members of the Special Interest Group on Music (SIGMUS) at the Information Processing Society of Japan (IPSJ) also provided many suggestions for this research.

I have been engaged in musical audio analysis in collaboration with other students of Okuno Laboratory. I studied automatic music transcription with Mr. Yohei Sakuraba (currently with Sony Corporation), drum sound recognition with Mr. Kazuyoshi Yoshii, chord recognition with Mr. Takuya Yoshioka (currently with NTT), singer identification with Mr. Hiromasa Fujihara, inter-media relation analysis with Mr. Masahiro Nishiyama, specified part transcription with Mr. Katsutoshi Itoyama, and environmental sound annotation with Mr. Akihiro Taguchi (currently with Inui Laboratory). I greatly appreciate their cooperation. Also, Dr. Kazunori Komatani gave me practical comments at meetings in our laboratory. Ms. Miki Nishii provided clerical support at our laboratory. The other members of Okuno Laboratory contributed to make a friendly atmosphere at our laboratory and I thank them for the experience.

Mr. Scott Brown gave me comments for improving English at weekly English classes.

The Support Center for Advanced Telecommunications Technology Research (SCAT) and the Japan Society for the Promotion of Science (JSPS) financially supported my life as a researcher.

I wish to thank my parents for allowing me to be a student for a long period and to stay absorbed in my study.

Finally, I wish to thank my family, Maiko and Fumika, for their unconditional support.

Contents

List of Figures

List of Tables

List of Symbols

$\Omega = \{\omega_1, \cdots, \omega_m\}$	Set of target instruments
ω_0	Silence event denoting no instruments being played (in Chapter 6)
$\Omega^+ = \Omega \cup \{\omega_0\}$	Set of target instruments including silence event (in Chapter 6)
$n_1, \cdots, n_K$	Notes contained in given signal (in Chapter 5)
$\boldsymbol{x}$	Feature vector (in Chapter 3)
$\boldsymbol{x}_k$	Feature vector for note n_k (in Chapter 5)
$\boldsymbol{x}(t, f)$	Feature vector at time t and at F0 of f (in Chapter 6)
$p(\omega_i \mid \boldsymbol{x})$	A posteriori probability
$p(\omega_i)$	A priori probability
$p(\boldsymbol{x} \mid \omega_i)$	Probability density function
$\mathcal{N}_{\mathrm{F0}}(\boldsymbol{\mu}_i(f), \Sigma_i)$	F0-dependent multivariate normal distribution
$\boldsymbol{\mu}_i(f)$	F0-dependent mean function for instrument ω_i
Σ_i	F0-normalized covariance for instrument ω_i
χ_i	Set of training data of instrument ω_i
$D^2_{\mathrm{M}}(\boldsymbol{x}; \boldsymbol{\mu}_i(f), \Sigma_i)$	Squared Mahalanobis distance
$\mathcal{H}$	Harmonic structure (in Chapter 3)
$\mathcal{H}(n_k)$	Harmonic structure extracted from note n_k (in Chapter 5)
$\mathcal{H}(t, f)$	Harmonic structure with F0 of f (in Chapter 6)
$\mathcal{H}_t(\tau, f)$	Harmonic structure with F0 of f staring at time t (in Chapter 6)
$F_i(t)$	Frequency of i-th partial at time t
$A_i(t)$	Amplitude of i-th partial at time t
$s_{\mathrm{h}}(n_k), s_{\mathrm{l}}(n_k)$	Maximum number of simultaneously played notes in higher or lower pitch range when note n_k is being played (in Chapter 5)
$\mathcal{N}$	Set of notes extracted for context (in Chapter 5)

$p(\omega_i; t, f)$	Instrument existence probability (IEP) (in Chapter 6)
$p(\mathrm{X}; t, f)$	Nonspecific instrument existence probability (NIEP) (in Chapter 6)
$(\omega_i \mid \mathrm{X}; t, f)$	Conditional instrument existence probability (CIEP) (in Chapter 6)
$M_0, \cdots, M_m$	Hidden Markov models for calculating CIEPs (in Chapter 6)
$I_1, \cdots, I_k$	Frequency subregions for simplifying instrogram (in Chapter 6)
$p(\omega_i; t, I_k)$	IEP for frequency subregion I_k (in Chapter 6)

Chapter 1

Introduction

This chapter briefly describes the motivation, goal, issues, and approaches of this thesis.

1.1 Motivation

One of the major functions lacking in the current computing technology is recognition of the real world. Humans use various information obtained from the real world through their eyes and ears to judge situations and appropriate behavior in everyday life. Computers' capability to recognize auditory and visual scenes is, however, strictly limited. In particular, there have been relatively few attempts to investigate sound recognition, except speech recognition studies. Techniques for recognizing a variety of sounds, not limited to speech, will be important to realize sophisticated computers that extensively use the real-world information.

One major reason why it is difficult for computers to recognize auditory scenes is that the auditory scenes in the real world usually contain multiple simultaneous sources of sound. Because conventional speech recognition studies have assumed that the input sounds to be recognized are voices spoken by a single speaker, they did not deal with situations where multiple sources simultaneously present sound. Although there have been a number of attempts to recognize speech under noisy environments, the number of the source to be recognized is always one; the other sound sources are regarded as noise.

Music is a good domain for studying computational recognition of auditory scenes because multiple instruments are usually played simultaneously. The difficulty in handling music resides in the fact that signals (sources to be recognized) and noises (sources to be ignored) are not uniquely defined. Therefore, multiple simultaneous sources should be modeled and recognized in parallel. This is the main difference from studies on speech

recognition under noisy environments.

Music recognition has a long history. After the fast Fourier transform was invented by Cooley and Tukey in the 1960s [1], attempts to apply fundamental frequency (F0) estimation techniques to musical audio signals were started. Because F0 estimation is directly connected to automatic music transcription, which aims at automatically obtaining a musical score representation from musical audio signals, it has been the main subject in music recognition research. Until the 1980s, however, the signals to be used were solo (monophonic) sounds [2]. Recently, since the 1990s, the targets of F0 estimation have been polyphonic music [3–19] (see Section 2.1.1 for details). On the other hand, recognition of other aspects of music, *e.g.*, instrument recognition, have not been studied as extensively as F0 estimation. In fact, until recently the main targets of most previous studies of instrument recognition were solo sounds [20–36], although the number of studies dealing with polyphonic music has been increasing in recent years [37–42].

Music recognition is also significant from an industrial viewpoint. The recent development of digital audio and network technologies has enabled us to handle a tremendous number of musical pieces. The newest portable music players can store over 20,000 pieces and allow us to choose favorite pieces from a huge collection and to listen to them anywhere. In addition, recent digital music distribution services have given us nearly unlimited access to music. Despite such developments, technologies for helping users find the musical pieces they want are not sufficient. To develop such technologies, music recognition, that is, extraction of musically meaningful information from musical audio signals, will play an important role.

1.2 Goals and Issues

In the light of the circumstances described above, we deal with recognition of non-percussive musical instruments in polyphonic music. Information on the instruments played in the audio signal is expected to have an important role because it characterizes musical pieces. In fact, the names of some musical forms are based on instrument names, such as "piano sonata" and "string quartet." When a user, therefore, wants to search for certain types of musical pieces, such as a piano sonata or string quartet, a retrieval system can use information on musical instruments. This information can also be used for jumping to the point when a certain instrument begins playing.

In this thesis, we investigate musical instrument recognition in two stages. At the first stage, we address instrument recognition of monophonic sounds in order to develop basic technologies for handling musical instrument sounds. In instrument recognition for monophonic sounds, we deal with the following two issues:

[Issue 1] Pitch dependency of timbre

In contrast to other sound sources including human voices, musical instruments have wide pitch ranges. For example, the pitch range of the piano covers more than seven octaves. Such a wide pitch range makes timbres quite different from pitch to pitch. This is a factor that makes musical instrument recognition difficult.

[Issue 2] Input of non-registered instruments

Although most existing studies on musical instrument recognition have used training data containing a limited number of musical instruments and have assumed that all input instruments were contained in the training data, this assumption is not always satisfied. Because there are numerous kinds of musical instruments in the world, it is practically impossible to prepare training data that cover all of them. In addition, the recent development of digital audio technology has made it possible to create novel and infinite kinds of original musical sounds (from sounds similar to natural instruments to sounds of instruments that do not actually exist). It is therefore essential to deal with non-registered musical instruments when recognizing musical instrument sounds.

At the second stage, we scale up the target of musical instrument recognition from monophonic sounds to polyphonic sounds. To deal with polyphonic music, we must solve the following two issues:

[Issue 3] Overlapping of simultaneously played notes

The difficulty of dealing with polyphonic music lies in the fact that it is impossible to extract the acoustical features of each instrument without blurring because of the overlapping of partials (harmonic components). If a clear sound for each instrument could be obtained with sound separation technology, the recognition of instruments in polyphonic music might become equivalent to the recognition of monophonic sounds. In practice, however, it is very difficult to separate a mixture of sounds without distortion.

[Issue 4] Unreliability of the precedent note estimation process

In the conventional musical instrument recognition framework, the instrument that plays each note is identified (notewise processing). The onset time and F0 of each note must therefore be accurately estimated before the identification phase. However, these estimations are generally not easy to make in polyphonic music, and thus estimation errors severely deteriorate the recognition performance. A new framework that can avoid the influence of these unreliable preprocesses is required in order to achieve instrument recognition in polyphonic music.

1.3 Overview of Our Approaches

We tackle the above-mentioned issues through the following approaches:

[Solution 1] F0-dependent Timbre Modeling

We propose a pitch-dependent model of timbres, which we term *F0-dependent multivariate normal distribution*. The F0-dependent multivariate normal distribution has two parameters: *F0-dependent mean function* and *F0-normalized covariance*. The F0-dependent mean function is defined as a function of F0, which represents the pitch dependency of each feature, while the F0-normalized covariance represents the non-pitch dependency. This modeling approximates the phenomenon in which tone features at different pitches have different positions (means) of distributions in the feature space even for the same instrument.

[Solution 2] Category-level Recognition of Non-registered Musical Instruments

We solve the non-registered instrument problem by recognizing such instruments at the category level. For example, a musical instrument sound that is similar to a violin and a viola but is not the same (for example, a sound made from the two instruments using a synthesizer) is recognized as "strings." Humans listening to this sound for the first time, would think, "I do not know this instrument, but it must be a kind of strings." This solution aims to achieve such human-like recognition on a computer.

[Solution 3] Feature Weighting based on Mixed-sound Template

We approach the overlap problem by weighting each feature based on how much the feature is affected by the overlapping. If we can give higher weights to features

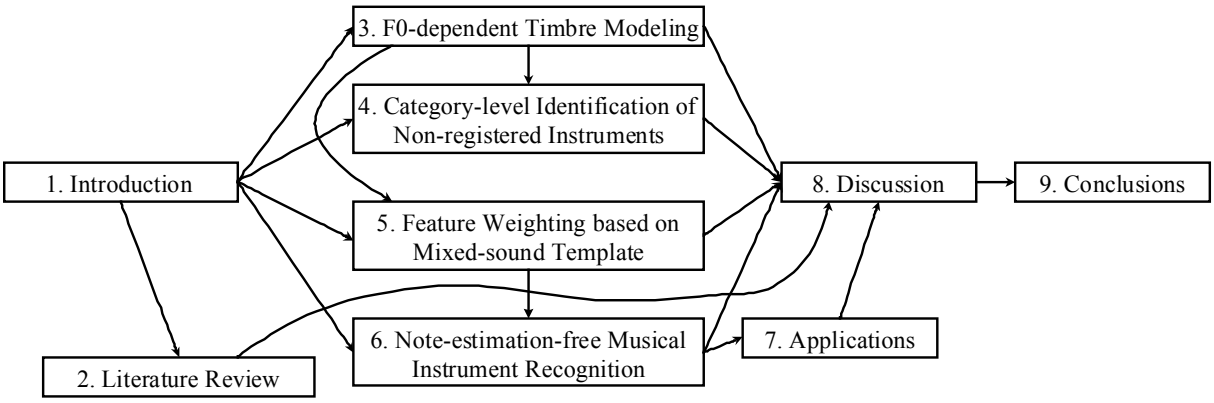


Figure 1.1: Organization of this thesis.

suffering less from this problem and lower weights to features suffering more, it will facilitate robust instrument recognition in polyphonic music. We achieve this feature weighting by using a *mixed-sound template* (*i.e.*, a set of instrument-labeled feature vectors extracted from polyphonic signals) and *linear discriminant analysis*.

[Solution 4] Note-estimation-free Musical Instrument Recognition

We propose a new framework for musical instrument recognition that does not use either onset detection nor F0 estimation of each note as the explicit preprocess. The key concept underlying this is to visualize the probability that the sound of each target instrument exists at each time and each F0 (called *instrument existence probability* (IEP)) as a spectrogram-like representation called an *instrogram*. The instrument existence probability is defined as the product of two probabilities: the *nonspecific instrument existence probability* and the *conditional instrument existence probability*. Because calculation of the former implicitly includes the onset detection and F0 estimation, our technique can avoid their explicit processing. In addition, because the two probabilities can be calculated independently, errors due to the calculation of one probability do not influence the calculation of the other probability.

1.4 Thesis Organization

The organization of this thesis is shown in Figure 1.1.

Chapter 2 provides a review of the literature in related fields and discusses the positioning of this thesis.

Chapter 3 describes the F0-dependent multivariate normal distribution, which is our solution to the problem of the pitch dependency of timbres. After we formulate the F0-dependent multivariate normal distribution, we describe the acoustic features we use for instrument recognition, comparing them with those of previous studies. We then report experimental results with an isolated solo musical instrument sound database that consists of 6,247 sounds of 19 instruments. We discuss the effectiveness of our approach comparing it with the usual (F0-independent) multivariate normal distribution. We also compare our approach with the k-NN classifier and the approach of appending F0 to the feature vector.

Chapter 4 describes category-level recognition of non-registered instruments. Because the category-level recognition requires a taxonomy of musical instruments, we first discuss a musical instrument taxonomy that is appropriate for category-level recognition. Specifically, we point out that the musical instrument taxonomy for category-level recognition should reflect similarity of timbres (acoustic features) and build a musical instrument taxonomy automatically based on acoustic similarity. This taxonomy is called *TimbreTree*. We then describe a method for category-level recognition and report experimental results of recognizing electric instruments with a recognizer that has been trained for only natural instruments.

Chapter 5 provides our solution to the overlapping problem in instrument recognition in polyphonic music based on feature weighting using the mixed-sound template. The key idea behind this is to extract training data from audio signals of polyphonic music. To perform the feature weighting, we have to extract training data from audio signals of polyphonic music. We therefore discuss extracting training data from polyphonic music signals after explaining the basic framework of instrument recognition in polyphonic music. The main issue in this is to design an appropriate subset of polyphonic sound mixture because there are an infinite number of possible combinations of musical sounds. We provide a simple solution to this and confirm its effect through experiments with synthesized audio signals of duo, trio, and quartet music. We also introduce musical context to avoid musically unnatural errors (only one clarinet note within a sequence of flute notes). We report that the use of musical context effectively improves instrument recognition in polyphonic music as long as the given musical pieces have rare crosses in pitch among simultaneous parts.

Chapter 6 proposes the note-estimation-free instrument recognition framework called

the instrogram analysis. First, we propose a probabilistic representation of instrumentation called an *instrogram*. The instrogram is a graphical visualization of IEPs calculated for every time and every F0. We then formulate the IEP and describe the algorithm of calculating the IEP. The effectiveness of the instrogram analysis is tested on recordings of both synthesized music and real performances.

Chapter 7 presents an application of the instrogram analysis to similarity-based MIR. Because the instrogram representation is directly connected to instrumentation, it can provide a new music similarity measure that reflects instrumentation. Because instrumentation is an important factor determining the impression of music, the instrogram-based music similarity measure will play an important role when we design a music similarity measure where various musical elements are separately handled and their weights can be determined adaptively to the users' preference.

Chapter 8 discusses major contributions of this study to different research fields including computational auditory scene analysis, content-based MIR, and music visualization. Remaining issues and future directions are also discussed.

Finally, Chapter 9 concludes the thesis.

Chapter 2

Literature Review

This chapter provides a review of the literature related to musical instrument recognition and content-based music information retrieval to clarify the positioning of this thesis within related fields.

2.1 Musical Instrument Recognition

In this section, we provide reviews of studies on several related fields to musical instrument recognition to make it possible to discuss the positioning of our musical instrument recognition study from different perspectives.

2.1.1 Musical Audio Signal Processing

Research on musical audio signal processing has a long history. In the 1970s, studies of F0 estimation for solo musical audio signals were started; music was a new target domain of signal processing techniques that were originally developed for speech analysis. Moorer [2] then built a system for transcribing duets. This system, however, was limited, succeeding only with music featuring two instruments of different timbres and frequency ranges, and with strict limitations on the allowable simultaneous musical intervals in the performances. In 1989, Katayose and Inokuchi [43] dealt with the problem of obtaining traditional scores from audio signals of multiple simultaneous-note performances played on a single instrument by combining signal processing and knowledge processing. Martin [3] and other researchers also dealt with similar problems. A new research field, *automatic music transcription*, was thus established.

Automatic Music Transcription

The music transcription system developed by Katayose and Inokuchi [43] interprets the time-frequency representation of a given audio signal by integrating some rule-based processing modules. The audio signal is first transformed to a set of note symbols, consisting of the pitches and the onset and offset times, where the onset and pitch are estimated based on data, stored in advance, of the attack time and the relative powers of partials of the target instrument. The note symbols are then organized as a musical score based on some knowledge of musical structure.

Martin [3] proposed a system for transcribing piano performances of four-voice Bach chorales using the blackboard framework. In this system, the blackboard workspace is arranged in a hierarchy consisting of a log-lag correlogram input (at the lowest level), peaks, periodicities, envelope, onsets, and notes (at the highest level). Several knowledge sources are used to make hypotheses at each level from lower-level information, and the other knowledge source is used to prune away obviously incorrect note hypotheses.

Klapuri *et al.* [4] developed a music transcription system using their onset detection [44], multiple F0 estimation [5], and harmonic sound separation [45] methods. Their F0 estimation method was based on the iterative processing of estimating the predominant F0 and removing its partials.

Marolt [6] proposed a system for transcribing piano music based on a connectionist approach. The system consists of two main parts: a partial tracking module, which calculates a time-frequency representation of an input audio signal, and a note recognition module, which groups the found partials into notes. In the note recognition module, time-delay neural networks are trained. Specifically, each network is trained to recognize a certain piano note in its input. The input to networks consists of the output of all the oscillator networks in a few recent time frames and of the amplitude envelopes at the outputs of the auditory feedback. Supervised learning with a large amount of piano music is used to train the neural networks.

Music Scene Analysis

Kashino *et al.* [46] proposed an architecture, called OPTIMA, for music transcription based on the Bayesian network. They introduced the hierarchical structure of partials (frequency components), notes, and chords, and probabilistically modeled their relationships. Their system determines the most likely interpretation using both bottom-up and

top-down clues. They used the term *music scene analysis*, which will be mentioned again later, in place of automatic music transcription because their work was motivated by *computational auditory scene analysis* [47]. Unlike most of the above-mentioned studies, they dealt with polyphonic music played on multiple instruments and the grouping of notes according to instrument.

Parametric Modeling of Multiple Instrument Sounds

In the late 1990s, the trend of F0 estimation moved from rule-based techniques to parametric modeling. This approach basically designs a parametric model that approximates power spectra containing multiple instrument sounds and searches for the model parameter where the model best fits the given spectra.

Goto [7] proposed a new method for modeling power spectra containing multiple instrument sounds. His PreFEst method models an observed power spectrum as a weighted mixture of tone models $p(x|F)$ of 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

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

where Fl and Fh denote the lower and upper limits, respectively, of the possible F0 range, and $w^{(t)}$ 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. The model parameter can be estimated using the Expectation-Maximization (EM) algorithm.

Kameoka *et al.* [8] also proposed a new method for multiple F0 estimation based on model parameter estimation using the EM algorithm. They modeled each spectral peak as a Gaussian model and a set of spectral peaks from a single note (a harmonic structure) as a Gaussian mixture model (GMM) where the means of the Gaussians corresponding to overtones are constrained to be integer multiples of that corresponding to the fundamental in the linear frequency scale. The model parameter set $\boldsymbol{\theta}$ for a power spectrum containing K harmonic structures is therefore represented as

$$\boldsymbol{\theta} \;=\; \{\boldsymbol{\mu}_k, \boldsymbol{w}_k, \sigma \mid k = 1, \cdots, K\},$$

$$\begin{aligned}\boldsymbol{\mu}_k &= \{\mu_k, \cdots, \mu_k + \log n, \cdots, \mu_k + \log N_k\}, \\ \boldsymbol{w}_k &= \{w_1^k, \cdots, w_n^k, \cdots, w_{N_k}^k\},\end{aligned}$$

where μ_k is F0 of k-th harmonic structure in the log-frequency scale and w_n^k and σ are weights and variance (that is assumed here as a constant) of the respective Gaussian distributions. These model parameters are estimated using the EM algorithm. They named this method *harmonic clustering* (HC). They also introduced the Akaike Information Criterion (AIC) to determine the number of GMMs (in other words, to estimate the number of harmonic structures).

They subsequently extended HC to represent temporal features in an audio stream [9]. Let $p(x, t|\boldsymbol{\Theta}_k)$ be the temporal stream of the k-th harmonic structure. This can then be represented as the product of the harmonic structure model $h_k(x)$ and the envelope function $g_k(t)$ as follows:

$$p(x, t|\boldsymbol{\Theta}_k) = w_k g_k(t) h_k(x),$$

where the weight w_k corresponds to the power of the k-th harmonic structure. The key point of this method is the formulation of the spectrogram of a harmonic structure as the multiplication of two different functions, one of which models the spectral aspect and the other of which models the temporal aspect. The harmonic structure $h_k(x)$ is modeled with a weighted sum of Gaussian kernels given by

$$h_k(x) = \sum_n \frac{r_n^k}{\sqrt{2\pi\sigma_k^2}} \exp\left[-\frac{\{x - (\mu_k + \log n)\}^2}{2\sigma_k^2}\right],$$

where r_n^k is related to the spectral components. This model is basically the same as that used in HC. The power envelope $g_k(t)$ is modeled as

$$g_k(t) = \sum_y \frac{c_y^k}{\sqrt{2\pi\phi_k^2}} \exp\left[-\frac{\{t - (o_k + y\phi_k)\}^2}{2\phi_k^2}\right]$$

where c_y^k is the weights that directly determine the shape of the power envelope and o_k is the center of forefront Gaussian. By setting the standard deviation of each Gaussian and the interval of adjacent Gaussians to the same variance ϕ_k, the power envelope $g_k(t)$ becomes a linear elastic function allowing various time lengths of audio streams. This method is called *harmonic-temporal-structured clustering* (HTC).

Sagayama *et al.* [10] proposed a new method for spectral analysis termed *specmurt*. The specmurt uses the property whereby the positions of the spectral peaks of overtones

shift in parallel to F0 fluctuation in the log-frequency scale. Assuming that all sounds in a given signal have a common harmonic structure, denoted as $h(x)$, which does not depend on the F0, they modeled the power spectrum $v(x)$ as the convolution of the F0 distribution $u(x)$ and the common harmonic structure $h(x)$, that is, $v(x) = h(x) * u(x)$, where x is the log frequency. The F0 distribution $u(x)$ lets us know the power of the harmonic structure with the F0 of x for every frequency x. Given an appropriate harmonic structure model $h(x)$, the F0 distribution can be calculated as $U(y) = V(y)/H(y)$ where $U(y)$, $H(y)$, and $V(y)$ are the Fourier transform of $u(x)$, $h(x)$, and $v(x)$, respectively. They subsequently proposed a method for estimating the quasi-optimal common harmonic structure model with an iterative calculation because the accuracy of the F0 distribution greatly depends on the common harmonic structure [11].

Otherwise, many researchers have proposed different approaches to tackle the multiple F0 estimation problem, for example, approaches based on non-negative matrix factorization [12], non-negative sparse coding [13], generative models [14, 15], instrument models [48], and Bayesian estimation with frequency domain modeling [16]. Vincent and Rodet's method based on independent subspace analysis [42], which will be described later, is also a multiple F0 estimation method based on parametric modeling of spectra. Detailed reviews can be found in [17–19].

Musical Instrument Recognition for Solo Sounds

As can be seen from the above review, the main subjects in the field of musical audio signal processing to date have been F0 estimation and its application to automatic music transcription. In fact, almost none of the above-mentioned studies (except OPTIMA) have dealt with musical instrument recognition. In the late 1990s, however, attempts on musical instrument recognition for solo sounds were started.

Most studies on instrument recognition for solo sounds [20, 23, 25–27, 29–31, 33, 36] dealt comparatively with many kinds of instruments (between 10 and 30). Various acoustic features were used; some were designed based on the knowledge of musical acoustics (*e.g.*, spectral centroid and odd/even energy ratio) [20, 23, 32, 33, 35] and some were used in speech recognition (*e.g.*, MFCCs and LPCs) [21, 22, 28, 30, 31]. Some studies adopted dimensionality reduction or feature selection techniques to avoid the redundancy of high-dimensional feature spaces [20, 24–27, 29, 32, 35]. The commonly used classifiers were the Gaussian [23, 29], GMM [21, 22, 28, 31, 32], k-NN [23–27, 29, 31, 33, 35, 36],

and SVM (support vector machine) [28, 32, 33] classifiers. While almost all of the recent speech recognition studies have employed hidden Markov models (HMMs), they were used by only a few studies on musical instrument recognition [30]. Some introduced hierarchical schemes [20, 23]. A number of studies achieved the recognition rates of 70–80% for more than 10 target instruments [20, 23, 31, 33] and some achieved about 90% [29, 36]; however, these studies cannot be directly compared because different data and different evaluation methods were used.

Martin [20] reported the results of large-scale experiments on musical instrument recognition. He proposed a hierarchical method for musical instrument recognition where the taxonomy of instruments was manually designed based on knowledge on the sounding mechanisms of instruments. He used a variety of acoustic features including spectral features (such as the spectral centroid and average relative spectrum), modulation features (such as the tremolo, centroid modulation, and individual harmonic amplitude modulation), and attack features (such as the relative onset time). This was a pioneering study that investigated the effectiveness of various acoustic features against musical instrument recognition through experiments using a large-scale database of monophonic sounds of actual instruments (1,023 sounds of 14 instruments). He also investigated human ability to recognize musical instruments.

Brown [21] dealt with recognition of oboe and saxophone sounds. She used cepstral coefficients as acoustic features because these instruments can be modeled as a resonator with a periodic excitation similar to that of human voice. She subsequently investigated acoustic features useful for recognizing woodwind instrument sounds because these instruments were difficult to distinguish from each other due to similar attacks, decays, pitch ranges, and modes of excitation [22]. She used frequency-domain features (such as cepstral coefficients, constant-Q coefficients and their bin-to-bin differences, and spectral centroid) and time-domain features (such as autocorrelation coefficients, moments of the residual of the LPC (linear prediction coefficients) filtered signal, the third (skew), fourth (kurtosis), and fifth moments of the raw signal, and the second through fifth moments of the envelope of the signal). She investigated the importance of each feature through testing the differences in the recognition accuracies with different feature sets.

Eronen and Klapuri [23] dealt with large-scale musical instrument recognition with the hierarchical approach using a variety of acoustic features similarly to Martin. They combined cepstral coefficients used by Brown and temporal features like those used by

Martin. Their experimental results from a dataset consisting of 1,498 monophonic sounds of 30 instruments showed the recognition rates of approximately 80% at the individual-instrument level and 94% at the category (instrument family) level.

Fujinaga [24, 25] developed an exemplar-based musical instrument recognition system, which is based on the k-nearest neighbor (k-NN) classifier. He introduced feature selection and feature weighting based on the genetic algorithm (GA) to improve recognition. For feature selection, the set of features is converted to "genes", where each feature is represented by a bit in the binary array. Each gene having a different sequence of bits then represents a subset of features to be used for classification and those having high recognition rates are made to survive in the pseudo-biological environment. The GA is used again to weight features after performing feature selection. His experimental results from a database consisting of 1,338 different sounds of 39 different timbres from 23 instruments showed that the recognition rate was approximately 50% when seven features were selected, while the best recognition rate for a single feature was 20%, where the selected feature was the centroid. He and his colleague subsequently re-implemented this method as a real-time recognition system based on Puckette's fiddle program [49], which is robust real-time F0 estimation software [26, 27].

Marques and Moreno [28] also dealt with recognition of musical instruments for short excerpts of audio signals of solo music. They first divided a given audio signal into segments 0.2 seconds in length. They then extracted three different features sets, linear prediction coefficients (LPC), FFT-based cepstral coefficients, and FFT-based mel cepstral coefficients, from each segment. Next they explored two different classification algorithms, the GMM and SVM. The relationship between the recognition rates and the choices of the feature sets and classification algorithms were investigated in detail.

Livshin *et al.* [29] dealt with musical instrument recognition for monophonic musical instrument sounds. They used the acoustic features proposed by Peeters and Rodet [50] and three different classifiers (Gaussian, learning vector quantization (LVQ), and k-NN) with linear discriminant analysis (LDA). They showed the recognition rate of more than 95% with leave-one-out (LOO) cross validation for 16 instruments. They also investigated the importance of each feature by comparing recognition rates with feature spaces in which some features were eliminated. Furthermore, they showed that the recognition rates with different databases for training and test sets were significantly lower than those with the same database.

Eronen [30] attempted musical instrument recognition using left-to-right HMMs. Given an audio signal, MFCCs and delta MFCCs were extracted every 15 ms. The feature space was then transformed by using independent component analysis (ICA). Next, left-to-right HMMs, which are commonly used in speech recognition studies, were trained or used to recognize the instrument through well-known techniques such as the Baum-Welch and Viterbi algorithms.

Krishna and Sreenivas [31] tried to recognize musical instruments for solo music recordings using only frame-level features in order to omit the process of onset detection, which is not easy in phrase performances. They extracted the spectral features termed *line spectral frequencies*, which can be obtained with LPC analysis, from every short segment. They then determined the instrument based on the average likelihood of GMMs for all of the frames. They also attempted to use the k-NN classifier instead of GMMs, and claimed that they achieved good performances despite the use of simple features (*i.e.*, no temporal features were used).

Essid *et al.* [32] dealt with monophonic musical instrument recognition using pairwise classification strategies. They extracted more than 150 acoustic features and reduced the dimensionality of this large-dimensionality feature space using two different methods of feature selection: genetic algorithms (GAFS) and inertia ratio maximization using feature space projection (IRMFSP). In the former, the approximately optimal lower-dimensional feature space is searched randomly under the guidance of the fitness function given by

$$F(C) = \frac{||\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2||_2}{\sqrt{\left(|\Sigma_1| + |\Sigma_2|\right)/2}},$$

where $\boldsymbol{\mu}_i$ and Σ_i $(i = 1, 2)$ are the mean vectors and the diagonal covariance matrices of the multivariate Gaussian distributions for the two target classes, respectively. In the latter, feature selection is made iteratively with the aim of deriving an optimal subset of d features among D, the total number of features. At each step i, a subset $\boldsymbol{X}_i$ of i features is built by appending an additional feature to the previously selected subset $\boldsymbol{X}_{i-1}$. The feature that should be appended is selected based on the following:

$$r_i = \frac{\sum_{k=1}^{K} \frac{N_k}{N} ||\boldsymbol{m}_{i,k} - \boldsymbol{m}_i||}{\sum_{k=1}^{K} \left(\frac{1}{N_k} \sum_{n_k=1}^{N_k} ||\boldsymbol{x}_{i,n_k} - \boldsymbol{m}_{i,k}|| \right)},$$

where $\boldsymbol{x}_{i,n_k}$ is n_k-th feature space (i-dimensional) from the class k, $\boldsymbol{m}_{i,k}$ and $\boldsymbol{m}_i$ are

respectively the means of the vectors of the class k and all classes, K is the number of classes, and N_k and N are respectively the numbers of the vectors of the class k and all classes. This criterion is known as the Fisher discriminant. This algorithm is also performed pairwise. As classification methods, GMMs, pairwise coupling [51], and SVMs were tried.

Using different approaches, a number of other researchers have tried musical instrument recognition for monophonic sounds in recent years [33–36]. Good reviews on recent musical instrument recognition studies are available, for example, in [52, 53].

Musical Instrument Recognition for Polyphonic Music

Although the targets of musical instrument recognition studies have been monophonic sounds until just recently, the number of studies now dealing with polyphonic music is increasing.

The first main difficulty in instrument recognition in polyphonic music (a mixture of multiple instrument sounds) lies in the fact that the sounds contained in the mixture interfere with each other, and this interference makes it difficult to extract acoustic features from the sounds exactly and robustly. If a clean sound for each instrument could be obtained using sound separation technology, instrument recognition for polyphonic music would become equivalent to the recognition of the monophonic sound of each instrument. In practice, however, a mixture of sounds is difficult to separate without experiencing distortion. When some partials (harmonic components) of some sounds in the mixture overlap in frequency, the separation is very difficult and therefore the acoustic features extracted from the mixture are quite different from those extracted from monophonic sounds. Recently, different researcher groups have tackled this overlapping problem through different approaches.

Kashino and Murase [37] proposed an architecture for sound source identification, called *Ipanema*, based on a multi-agent scheme. An agent is prepared for each target instrument, and each agent tries to detect the sound of the target instrument, and regards the sounds of the other instruments as noise. Each agent maintains a bank of waveforms, each of which is a waveform of a single note of a specific pitch and expression. Each agent examines the input F0 and checks whether the F0 is within the pitch range of the instrument corresponding to the agent. And, if there are a possibility of being included, the agent suggests a waveform, applying a phase tracking method to one of the

waveforms stored in the bank. The agents that suggested waveforms then modify (adapt) them to minimize the squared error of the suggested waveforms. Let $z(k)$ and $r_n(k)$ be the input signal and the waveform of n-th source, respectively, where k represents time. The modification of the waveforms, called template adaptation, is formulated as estimation of FIR filters $h_n(m)$ that minimize the following equation:

$$J = E\left[\left\{z(k) - \sum_n (h_n(m) * r_n(k))\right\}^2\right],$$

where $E[\cdot]$ is the temporal average and $*$ denotes convolution. This was a pioneering study dealing with polyphonic music and its importance lay in that they formulated instrument recognition as the matching of waveforms instead of feature vectors, which were difficult to extract robustly. In addition, they introduced music stream networks, which model sequences of melodies, to take musical context into account.

Kinoshita *et al.* [38] tackled the overlapping problem using a different approach. They manually categorized the acoustic features for recognition into three types (additive, preferential, and fragile) according to how the features varied when partials overlapped. When partials of multiple sounds overlap, the additive features extracted from the partials tend to become close to the sum of the partials in the monophonic case. The preferential features tend to become close to the maximum or minimum of the partials in the monophonic case and the fragile features tend to become meaningless values. According to this categorization, the features are recalculated or invalidated when overlapping partials are observed.

Eggink and Brown [39, 40] introduced the missing feature approach to tackle the overlapping problem. This is a technique for canceling unreliable features using a vector called a mask, which represents whether each feature is reliable or not. Although this technique is known to be effective if the features to be masked are correctly estimated, automatic mask estimation is difficult and still an unsolved problem. They tried two kinds of masks: the a priori mask and the pitch-based mask. The a priori mask is generated by comparing the observed signal and the clean (unmixed) signal, which is impossible to obtain in realistic situations. This mask is therefore not applicable to realistic situations and is useful only for investigating the upperbound of the missing feature approach. Once the F0s of overlapping sounds are estimated, the partials that are possible to overlap with each other can be determined. If a partial from the target sound overlap with partials from any non-target sounds, in the pitch-based mask, the features derived from this partial are

masked. A limitation in the missing feature approach lies in the requirement of locally spectral features, because the features used for recognition should clearly correspond to specific frequency regions, as the main idea is to exclude specific frequency regions.

The second major difficulty in instrument recognition in polyphonic music is that the preprocesses of recognition (*e.g.*, onset detection and F0 estimation) are not sufficiently reliable for polyphonic music. In the frameworks of Kashino and Murase [37] and Kinoshita *et al.* [38], instrument recognition was performed for each note. They needed to estimate the onset time and F0 of each note to extract the segment corresponding to the note before identifying the instrument for the note. Onset detection and F0 estimation for polyphonic music are, however, still challenging problems and their errors can adversely affect instrument recognition. Eggink and Brown's framework [39, 40] does not need to estimate the onset time of each note because it identifies the instruments for each frame. It does, however, need F0 estimation for each frame to generate pitch-based masks. This is why most of the above-mentioned studies [37, 40] gave their systems correct onset times and/or F0s in the experiments. Kinoshita *et al.* [38] reported that, given random note patterns taken from three different instruments, instrument recognition performance was around 72–81% for correct F0s but decreased to around 66–75% for estimated F0s. Vincent and Rodet [42] and Essid *et al.* [32] proposed new instrument recognition techniques that can avoid this difficulty.

Vincent and Rodet [42] formulated both music transcription (onset detection and F0 estimation) and instrument recognition as a single optimization problem. Let $(\mathbf{x}_t)$ be the short-time log-power spectra of a given musical excerpt containing n instruments. Denoting $\mathbf{m}_{jt}$ the power spectrum of instrument j at time t and $\mathbf{\Phi}'_{jht}$ the log-power spectrum of note h from instrument j at time t, they assume:

$$\begin{aligned}
\mathbf{x}_t &= \log\left[\sum_{j=1}^{n}\mathbf{m}_{jt}+\mathbf{n}\right]+\epsilon_t \\
\mathbf{m}_{jt} &= \sum_{h=}^{H_j}\exp(\mathbf{\Phi}'_{jht})\exp(e_{jht}), \\
\mathbf{\Phi}'_{jht} &= \mathbf{\Phi}_{jh}+\sum_{k=1}^{K}v_{jht}^{k}\mathbf{U}_{jh}^{k},
\end{aligned}$$

where $\exp(\cdot)$ and $\log(\cdot)$ are the exponential and logarithm functions applied to each coordinate. The vector $\mathbf{\Phi}_{jh}$ is the unit-power mean log-power spectrum of note h from instrument j and $(\mathbf{U}_{jh}^{k})$ are L_2-normalized "variation spectra" that model variations of

the spectrum of this note around $\mathbf{\Phi}_{jh}$. The scalar e_{jht} is the log-power of note h from instrument j at time t and (v_{jht}^k) are "variation scalars" associated with the "variation spectra." The vector $\mathbf{n}$ is the power spectrum of the stationary background noise. The modeling error vector ϵ_t is assumed to be a white Gaussian noise. The log-power e_{jht} is considered to be determined based on a hidden discrete state $E_{jht} \in \{0, 1\}$ denoting absence or presence; it is constrained to $-\infty$ given $E_{jht} = 0$ and it follows a Gaussian law given $E_{jht} = 1$. The instrument model $\mathcal{M}_j$, for each instrument j, is defined as the collection of the fixed parameters describing instrument specific properties, for example $\mathbf{\Phi}_{jh}$ and (U_{jh}^k), and is trained in advance. Then, maximizing $P_{\text{trans}} = P(\mathcal{O}, (E_{jht}), (\mathbf{p}_{jht})|(\mathbf{x}_t))$, where $\mathcal{O}$ is a list of instrument models, with $\mathbf{p}_{jht} = (e_{jht}, v_{jht}^1, \cdots, v_{jht}^K)$ means finding the best (E_{jht}) and $\mathcal{O}$ explaining $(\mathbf{x}_t)$, which approximate music transcription and instrument recognition, respectively. Both music transcription and instrument recognition is thus achieved as a single optimization process.

Essid *et al.* [41] tackled the problem of the unreliability of preprocessing using a completely different approach. They introduced a multi-instrument recognition scheme processing real-world music that did not require F0 estimation or separation steps. Their approach exploits a taxonomy of musical ensembles to represent every possible combination of instruments likely to be played simultaneously in relation to a given musical genre.

Different from Essid *et al.*'s study [32], other research [37–40, 42] dealt with duo or trio music played on instruments chosen from 3–5 instrument candidates and achieved recognition rates of approximately 50 to 88%. They did not deal with music containing vocal or percussive sounds.

2.1.2 Computational Auditory Scene Analysis

Computational auditory scene analysis (CASA) [47] is the research field aiming at implementation of human auditory functions on a computer. Humans can hear various auditory events that often occur simultaneously and can understand the events seamlessly regardlessly of whether these are human voices, music or other sounds. They can also focus their listening attention on a single talker among a mixture of conversations and background noises, ignoring other conversations, in a very noisy environment (the cocktail party effect). A framework to represent such flexible auditory functions, known as *auditory scene analysis* (ASA), has been proposed by Bregman [54] and has been investigated in detail

from a psychoacoustical viewpoint. Attempts at strictly formulating and implementing auditory scene analysis from an engineering standpoint have been started in the field of artificial intelligence. This work was inspired by Marr's work on vision [55].

Bregman discussed the framework of ASA from the viewpoint of two kinds of grouping [54]. The first is *simultaneous grouping*, which groups frequency components originating from the same auditory event spread along the frequency axis. The second is *sequential grouping*, which groups temporally sequential auditory events originating from the same sound source (the sequence of the grouped events is called a *stream*). Bregman pointed out the phenomenon in which a sequence of alternate auditory events originating from two different sources (*i.e.*, A-B-A-B-...) is perceptually segregated into two streams, and called this phenomenon *auditory streaming*. He investigated various acoustic cues causing auditory streaming through various experiments, for example, proximity of the frequencies, rate of the sequence, and similarity of the timbres.

There are two main features of the work in this field. The first is a unified framework of speech and non-speech understanding, which is in contrast to conventional speech recognition research which aims to recognize only speech (non-speech sounds are always noise). The second is that the inputs are mixtures of multiple auditory events. The CASA can be formulated as the problem of inputting a set of auditory events and outputting a description of each event. This formulation has a close affinity with music recognition because music usually contains multiple auditory events and some representations of auditory events in music have been established, such as traditional scores. In fact, Kashino et al. [37, 46] dealt with music as an example target domain in their CASA research. The concept of the two groupings proposed by Bregman [54] can be applied to music transcription with no modification.

2.1.3 Recognition of Other Sounds

The most popular research field on recognition of non-musical sounds is speech recognition (including spoken content and speakers) [56, 57]. The recent development of large-scale speech corpora and statistic pattern recognition approaches has made speech recognition sufficiently accurate as long as the given signal is clearly spoken (*i.e.*, like the speech of professional announcers) and does not contain noise. The recognition is, however, very difficult when the signals include noise, in particular, when multiple people speak simultaneously. The major approach for dealing with such situations is sound source separation

with many (typically more than two) microphones. If the number of microphones is equal to or greater than the number of sound sources, separation of the sources is possible, for example, using independent component analysis (ICA) [58, 59]. Studies of the separation of multiple sources and recognition of the separated sources have been carried out separately in many cases, but recently there have been some studies of speech recognition under conditions in which other sound sources exist. For example, some researchers have been trying recognition of multiple speech sources or speech under noise using missing feature theory [60, 61]. Furthermore, a number of researchers have recently been attempting speech recognition in noisy (*e.g.*, in-car) environments [62].

The other field in non-musical sound recognition research is recognition of environmental sounds. A relatively limited number of attempts [63, 64] have been reported compared to speech and music recognition.

2.1.4 Perceptual Timbre

What kind of physical properties of sounds correspond to what we perceive as *timbre*? This has been an unsolved problem in acoustics for many years. Of the three basic psychoacoustical parameters of sounds (pitch, intensity, and timbre), pitch and intensity can usually be measured in physical properties, *i.e.*, fundamental frequency and amplitude. Timbre, on the other hand, which is also called *tone color* or *tone quality*, is difficult to put on a physical scale. Timbre is considered multidimensional and more complex than the other parameters and hence has never been fully defined. In fact, the American Standards Association has defined timbre as the following:

> "Timbre is that attribute of auditory sensation in terms of which a listener can judge that two sounds having the same loudness and pitch are dissimilar."

This definition is indirect and has serious limitations. In order for this definition to apply, for example, two sounds need to be able to be presented at the same pitch.

There are two acceptable standpoints in the definition of timbre. One is to consider timbre to be an acoustical characteristic corresponding to all aspects of the impression that humans receive from sounds. In this case, timbre would be described verbally. The other is to consider timbre to be an acoustical characteristic linked to differences between the sounds of different instruments. In this case, the names of the instruments can be used as labels for the timbres [54, 65]. While studies on computational musical instrument

recognition including ours adopt the latter standpoint in general, studies on human timbre perception such as those described here adopt the former standpoint.

Human timbre perception was well studied in the 1970s. In most of these studies, human subjects were asked to rate the timbre similarity between a pair of stimuli and the results were analyzed by multidimensional scaling (MDS). The stimuli used were real instrument sounds in some studies and artificial sounds in other studies.

Wedin and Goude [66] used stimuli of nine real instruments and asked two tasks of their human subjects. The first task was to identify the instrument name of each stimulus. From the results of this task, they discussed the accuracy of correct identification of instruments by humans. The second task was to rate the timbre similarity of every pair of stimuli. After that, they also asked the subjects to state the similarity between the instruments presented only by name. The results of the former similarity rating showed the "perceptual structure" of the acoustic characteristics of musical instruments, while the results of the latter similarity rating showed the "cognitive structure" of instruments, that is, knowledge about classification of the instruments. They concluded that these two structures of instruments were different for both trained and naive listeners.

Bismark [67] investigated the relationship between verbal attributes and spectral patterns in terms of verbal attributes from timbre factor analysis. In the experiments, 30 pairs of verbal attributes and 35 artificial sounds were used.

Grey [68], in a similar manner to Wedin and Goude, evaluated the perceptual similarity of timbres using 16 notes of synthetic sounds emulating 12 orchestral instruments. Perceptual similarity was measured using two methods: one judged the similarities for all pairs of the 16 notes, and the other used the accuracy of listeners in associating specific names with the notes in a learning task with feedback. In this case, the psychological distance of two tones was related to the number of confusions that occurred between them. The results of similarity estimation were then treated with MDS and hierarchical clustering. Three dimensions were found, corresponding to the spectral energy distribution, the presence of synchronicity in the transients of the higher harmonics, and the presence of low-amplitude, high-frequency energy in the attack segment, respectively.

The same author subsequently studied timbre discrimination in musical context while most of the previous studies dealt with discrimination of isolated tones [69]. He prepared two kinds of synthetic tones for each target instrument (*i.e.*, clarinet, basson, and trumpet). One is the "complete" version where the signal is emulated directly from a spectral

analysis, and the other is the "line-segment" version where the power envelope of each overtone is approximated to a time-varying function. He used 12 musical contexts (patterns of note sequences) consisting of isolated tonal comparisons, single-voice patterns, and multi-voice (up to three) patterns. The stimuli of these patterns were played back from computers using synthetic tones of the clarinet, basson, and trumpet. When they were played back, the different versions of synthesis may or may not have been used for the first and second halves. Thirteen musically sophisticated listeners judged whether the presented stimulus changed in timbre, that is, whether the first and second halves used the same version of synthesis. When the instrument was the clarinet or trumpet, the accuracy of judgment for the multi-voice patterns was the lowest of the three kinds of context patterns, and that for the single-voice pattern was lower than that for the isolated tonal comparisons. For the basson, on the other hand, there were no significant differences among the context patterns.

While some studies on timbre perception were conducted in the 1970s [70, 71], relatively little research occurred again until the 1990s, for example, [72–74].

Marozeau *et al.* [75] focused on the dependency of timbre on F0. They conducted three experiments. In Experiment I, subjects compared the timbres of stimuli produced by a set of 12 instruments with equal F0, duration, and loudness. This was repeated three times with different F0s. The results showed that the dissimilarity matrices were similar at different F0s. In Experiment II, the same stimuli were rearranged in pairs, each with the same difference in F0s (*i.e.*, 2 and 11 semitones). The similarity matrices for both 2- and 11-semitone differences resembled those of Experiment I. In Experiment III, subjects rated the timbre similarity between pairs of instruments with and without a difference in F0s, and the similarity matrices were analyzed similarly to the previous experiments. They concluded that timbre differences could be perceived independently from differences of F0s when the F0 differences were smaller than one octave.

2.2 Content-based Music Information Retrieval

Music information retrieval (MIR) has become one of the hottest topics in the field of computing technology. The International Conference on Music Information Retrieval (ISMIR) has been held annually since 2000, and its scale (*e.g.*, the numbers of submitted and published papers and attendees) has grown larger year on year. Although a thorough

review is impossible because of the great variety of topics being studied, we review recent MIR studies here from several aspects.

2.2.1 Styles of Retrieval

Although various styles of MIR have been proposed, the styles that have been particularly well studied are "Query-by-Humming" and "Query-by-Example".

The Query-by-Humming, as the name implies, enables one to retrieve the title of a musical piece by humming or singing its melody using sounds like "*la-la-la...*" [76]. Whereas the Query-by-Humming was the predominant subject in the 1990s, other styles have been more studied in recent years. Query-by-Humming technology has mostly move on to the industrial application phase.

The Query-by-Example is also known as "similarity-based MIR." In this style, users specify their favorite musical piece(s) and the system searches for musical pieces that are similar to the specified piece(s) in some sense. The main issue in this style is the design of the similarity measure.

There have been many other styles used also. For example, retrieval by specifying a certain fragment of a musical audio signal is effective in the situation where one would like to know the title of a musical piece that is currently being heard [77]. Retrieval by specifying adjectives (*e.g.*, happy) enables us to search for musical pieces that present a certain impression [78].

2.2.2 Techniques for MIR

Here, we briefly review music similarity measure, genre classification, and musical content description as techniques required for MIR, although this is not an exhaustive list.

Music Similarity Measure

The design of appropriate music similarity measures is a central subject in the recent MIR field. Aucouturier and Pachet's study [79], which is one of the pioneering studies on music similarity, used cepstrum coefficients and GMMs for calculating music similarity. Paulus and Klapuri [80] proposed a method for measuring the similarity of rhythmic patterns. Pampalk developed a MATLAB toolbox for calculating music similarity using various acoustic features [81]. Casey and Slaney [82] pointed out the importance of temporal modeling of musical features in calculating music similarity and investigated

several methods for modeling temporal features. There have also been many studies related to music similarity, such as classification and visualization of music collections using self-organizing maps (SOMs) based on music similarity [83], evaluation of acoustic and subjective music similarity measures [84], and detailed investigation of timbre similarity [85, 86]. These studies mainly used low-level acoustic features such as cepstrum. Higher-level, musically meaningful features are desired for further improvement of music similarity [87].

Genre Classification

Genre classification is also widely studied (*e.g.*, [88]; see [89] for a review) because it is directly usable for a task such as searching for jazz music. In addition to the music similarity measurement, genre classification is usually performed using low-level acoustic features and pattern recognition techniques such as GMMs and SVMs. This task has a difficult problem stemming from the ambiguity of musical genres. There are no agreed methods for designing a genre taxonomy that is widely acceptable because there are no strict definitions and unambiguous boundaries of genres.

Musical Content Description

The ability to describe musical elements (*e.g.*, melody, rhythm, harmony, and timbre) in a universal format will play an important role in achieving sophisticated MIR. Such descriptions are called *metadata*, and frameworks for widely distributing and exchanging them have been established in recent years.

MPEG-7, formally named "Multimedia Content Description Interface", is a standard for describing multimedia content, including music, in a universal format as metadata established by the International Organization for Standardization (ISO) [90]. MPEG-7 includes a standardized set of descriptors (Ds) and description schemes (DSs) for audio-visual content as well as a formal language for defining new Ds and DSs. In the context of MPEG-7, Ds represent elements of the content (*e.g.*, a representation of a feature and the syntax and semantics of the feature representation) while DSs represent structures of the content (*e.g.*, the relationship between Ds and/or DSs). The language for designing Ds and DSs is called Description Definition Language (DDL) and is derived by extension of XML Schema, which is developed by the W3C consortium using Extensible Markup Language (XML) as the basis.

Several Ds and DSs for audio content (Audio Ds and DSs) are included in the original MPEG-7 standard. These have been designed on the basis of the idea of two-layer description. Audio Ds and DSs can be divided into generic lower-level tools and application-specific high-level tools. The former represent features that can be automatically extracted from signals, such as AudioWaveformType, AudioPowerType, AudioSpectrumCentroidType, AudioFundamentalFrequencyType, LogAttackTimeType, HarmonicSpectralCentroidType, and TemporalCentroidType. The latter consist of general sound recognition and indexing tools, spoken content description tools, musical instrument timbre description tools, and melody description tools. These represent more abstract concepts than those represented by the former and include Ds/DSs assuming manual annotation.

Gomez *et al.* [91] discussed the use and enhancement of the MPEG-7 standard to describe musical content. They pointed out that the current MPEG-7 standard has limitations regarding different musical layers, *e.g.*, melodic, rhythmic, and instrumental, and presented some proposals for overcoming the limitations. They did not, however, deal with automation of musical content description.

Another approach for music description is XML formats based on traditional musical scores. MusicXML [92] and WEDELMUSIC Format [93] are the two major score-based formats, and many techniques for handling documents in these formats have actively been developed. It is however difficult to describe content that is not usually included in traditional scores, such as acoustic characteristics of a certain instrument sound.

Moreover, there have been some research projects aimed at developing techniques for automatically describing musical content from audio signals.

The CUIDADO project [94, 95] aimed to develop a new chain of applications through the use of audio/music content descriptors, in the spirit of the MPEG-7 standard. The project included the design of appropriate description structures, the development of extractors for deriving high-level information from audio signals, and the design and implementation of two applications: the Sound Palette and the Music Browser. These applications include new features, which systematically exploit high-level descriptors and provide users with content-based access to large catalogues of audio/music material.

Goto [96] launched the music scene description project, the goal of which is to build a computer system that can understand musical audio signals in a human-like fashion. Even if people listening to music cannot transcribe it as a traditional score, they can easily hum the melody and notice a phrase being repeated. He claimed, from this fact,

that transcribing music as a traditional score is not essential for music understanding and that the computer system should obtain higher-level descriptions. He therefore developed techniques for extracting five high-level descriptions: hierarchical beat structure, melody line, base line, repeated sections, and chorus sections. However, his system and its details, input/output data formats for example, are not publicly available.

The SIMAC project [97] investigated extraction of the semantic description of musical content from audio signals. The descriptors that they dealt with include rhythm, harmony, timbre, subjective intensity, and structure. The project developed music similarity measures and created a prototype MIR system based on the measures.

2.2.3 Applications of MIR Techniques

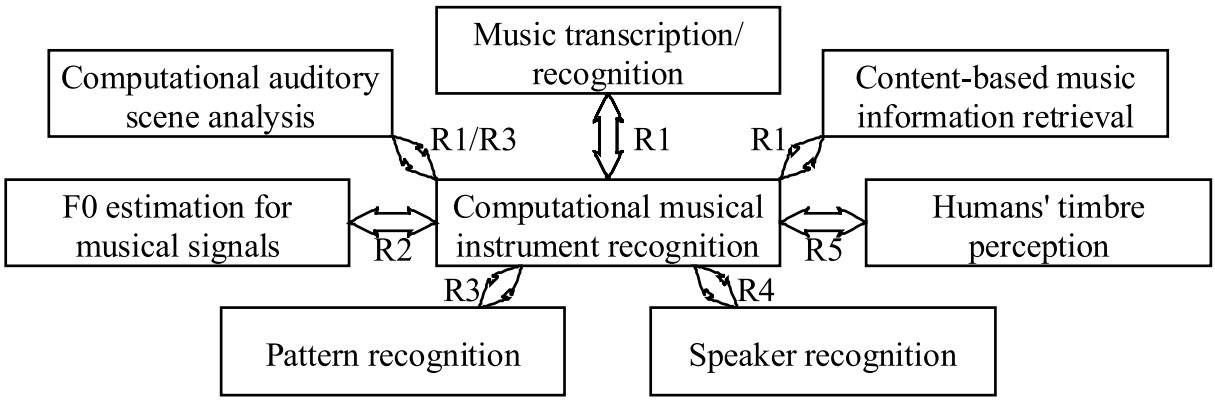
MIR can be useful in itself but it is also used as a subtask of automatic playlist generation [98], content-based music recommendation [99], etc. Furthermore, integrated environments for accessing music, such as PersonalRadio [100], have been proposed. PersonalRadio is a prototype for set-top-box music services with a slider ranging between two extreme values (from conservative to exploratory). The explorativeness of music selection is determined depending on the position of the slider.

2.3 Positioning of This Thesis

In this section, we discuss the positioning of this thesis by comparing it with related studies reviewed above.

2.3.1 Positioning of Musical Instrument Recognition

The relationship between musical instrument recognition and the related fields that we reviewed above are shown in Figure 2.1. Musical instrument recognition is placed in the broader field of music recognition because music recognition aims at recognizing various aspects of music from audio signals. Also, it is a subtask for automatic music transcription, usually referred to as transcribing music in a traditional musical score format, and musical content description, usually referred to as describing what the music is like in an XML-based format. Because CASA aims to provide a unified framework that deals with a variety of sounds, it obviously includes musical instrument recognition as a subtask. Musical instrument recognition is also useful as a subtask for content-based MIR. Pattern



R1: One is a subtask for the other.
R2: The inputs of the analysis are same but the outputs are different.
R3: One is a specific target domain of the other (more general research field).
R4: The target domains are different but the techniques can commonly be used in part.
R5: Recognition of the same aspect of sounds by humans and computers.

Figure 2.1: Relationship between musical instrument recognition and related fields.

recognition is the research field that aims to connect patterns observed through a sensor with their semantic categories. It therefore includes musical instrument recognition as well as character recognition, image recognition, and speech recognition. F0 estimation for musical signals and musical instrument recognition have a commonality in terms of inputs to the systems. Some techniques, especially for the front-end, can therefore be commonly used. Both are subtasks for automatic music transcription. Studies on computational musical instrument recognition and human timbre perception investigate the nature of timbre through different approaches.

Relationship with Automatic Music Transcription and F0 Estimation

We consider both musical instrument recognition and F0 estimation to be indispensable subtasks for automatic music transcription because both are necessary for generating a complete multi-part score with instrument labels. Notes for different instruments, in general, should be described on different staves in a score, and each stave should have the description of the instrument. However, most of the studies aimed at automatic music transcription have dealt with F0 estimation only. This could be because the current F0 estimation technologies can barely deal with complex musical pieces played on multiple instruments accurately. Although F0 estimation for multi-note performances played on a single instrument has been actively studied and is gradually reaching a practical

level, F0 estimation for multi-part performances played on multiple instruments remains a challenging problem.

If musical instrument recognition is applied to automatic music transcription, instrument recognition should be performed *notewise*; in other words, the instrument playing each note contained in the target polyphonic musical signals should be recognized. In Chapter 5, we address such notewise instrument recognition for polyphonic music (up to quartet). We provide a solution to the problem of the overlapping of common-frequency partials played on multiple instruments based on feature weighting using a mixed-sound template. We also introduce musical context to avoid musically unnatural errors. These can be directly applied to automatic music transcription for polyphonic music. In fact, we apply them to transcribing musical audio signals in a simplified MusicXML format. MusicXML is one of the most popular XML formats for transcribing music in a traditional musical score form.

F0 estimation is also used as a preprocess of musical instrument recognition because most acoustic features used for instrument recognition are based on the harmonic structure, which cannot be extracted until the F0 is estimated. In our instrument recognition method described in Chapter 5, F0 estimation is used as a preprocess; however, correct data are given in experiments. On the other hand, the instrogram analysis, described in Chapter 6, does not use F0 estimation as a deterministic preprocess. This new framework instead probabilistically integrates the results of the processes corresponding to F0 estimation and instrument recognition. We consider that we have developed a new relation of F0 estimation and instrument recognition.

Relationship with Content-based Music Information Retrieval

Musical instrument recognition techniques are expected to play an important role in improvement of content-based MIR. For example, most existing studies on music similarity measurement and genre classification have used lower-level features such as MFCCs. Such features can be reliably extracted and are useful for capturing characteristics of music to some extent. However, the correspondence of these features and their musical meaning is unclear. For example, the difference of MFCCs between two pieces may be caused by the difference of instrumentation and may be caused by the difference of chords. Thus it is difficult to achieve elaborate services, such as adapting the weight of each musical element to the preferences of users in measuring music similarity. To solve this problem,

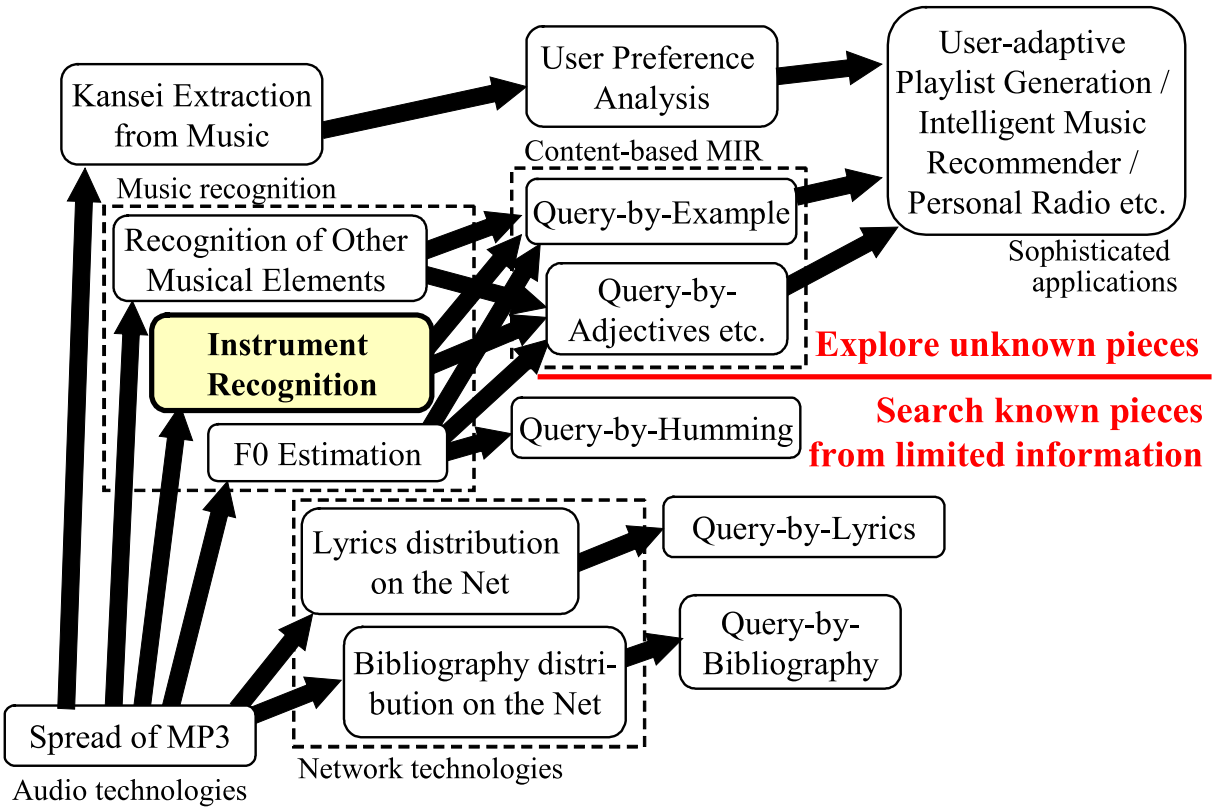


Figure 2.2: Long-span roadmap to MIR applications.

higher-level, musically meaningful features including instrumentation are necessary.

Projects of musical content description such as those reviewed above will solve this problem. Most of these studies, however, did not deal with instrument recognition. Our instrument recognition study and such projects are therefore complementary, and integrating them will achieve total music description.

We next discuss the positioning of musical instrument recognition within a long-span roadmap to sophisticated MIR applications (Figure 2.2). MIR applications can be classified according to their stances into two different categories. The first is retrieval of musical pieces that are known to the user based on limited information. The Query-by-Humming style, for example, aims to identify musical titles only from the melodies hummed by the user. The second is exploration of musical pieces that are unknown to the user. The Query-by-Example style is an example of this. Music recognition including instrument recognition is expected to become a key technology for this type of MIR.

Relationship with Pattern Recognition and Speech/Speaker Recognition

Both musical instrument recognition and speech/speaker recognition are concrete domains of pattern recognition. Theoretical and practical knowledge of general pattern recognition is useful in musical instrument recognition research. In fact, we apply commonly used theories and techniques in pattern recognition research, such as principal component analysis (PCA), linear discriminant analysis (LDA), multivariate normal distribution, and the Bayes decision rule. In addition, musical instrument recognition and speech/speaker recognition have a commonality in terms of the feature of inputs; inputs assumed in both studies are time-series data. We therefore introduce HMMs in Chapter 6 to model time-series data inspired by the successful use of HMMs for speech recognition.

Relationship with Human Timbre Perception

A common interest in studies on computational musical instrument recognition and human timbre perception is what exactly is timbre. Although standpoints for defining timbre differ, our study and some studies on human timbre perception focus on similar aspects of timbres.

First, we and Marozeau *et al*[75] focus on the dependency of timbres on F0s. We both point out that F0 is a factor influencing timbre. Although Marozeau *et al.* concluded that relative perception of timbre similarity could be independent from F0, acoustic features of musical instruments definitely vary according to the F0. We therefore deal with explicitly modeling the dependency on the F0.

Second, we and some studies on human timbre perception deal with hierarchical classification of timbres. The hierarchical taxonomies of timbres that we build by calculating acoustic similarity on a computer and those based on human perception can be compared.

2.3.2 Positioning of This Thesis within Previous Musical Instrument Recognition Studies

In contrast to the comparison of our study with related fields above, we now compare our study with previous musical instrument recognition studies.

Difficulty of Problem

Two main factors affecting the difficulty of instrument recognition are the numbers of simultaneous notes (#SN) and target instruments (#TI). If #SN for recordings to be

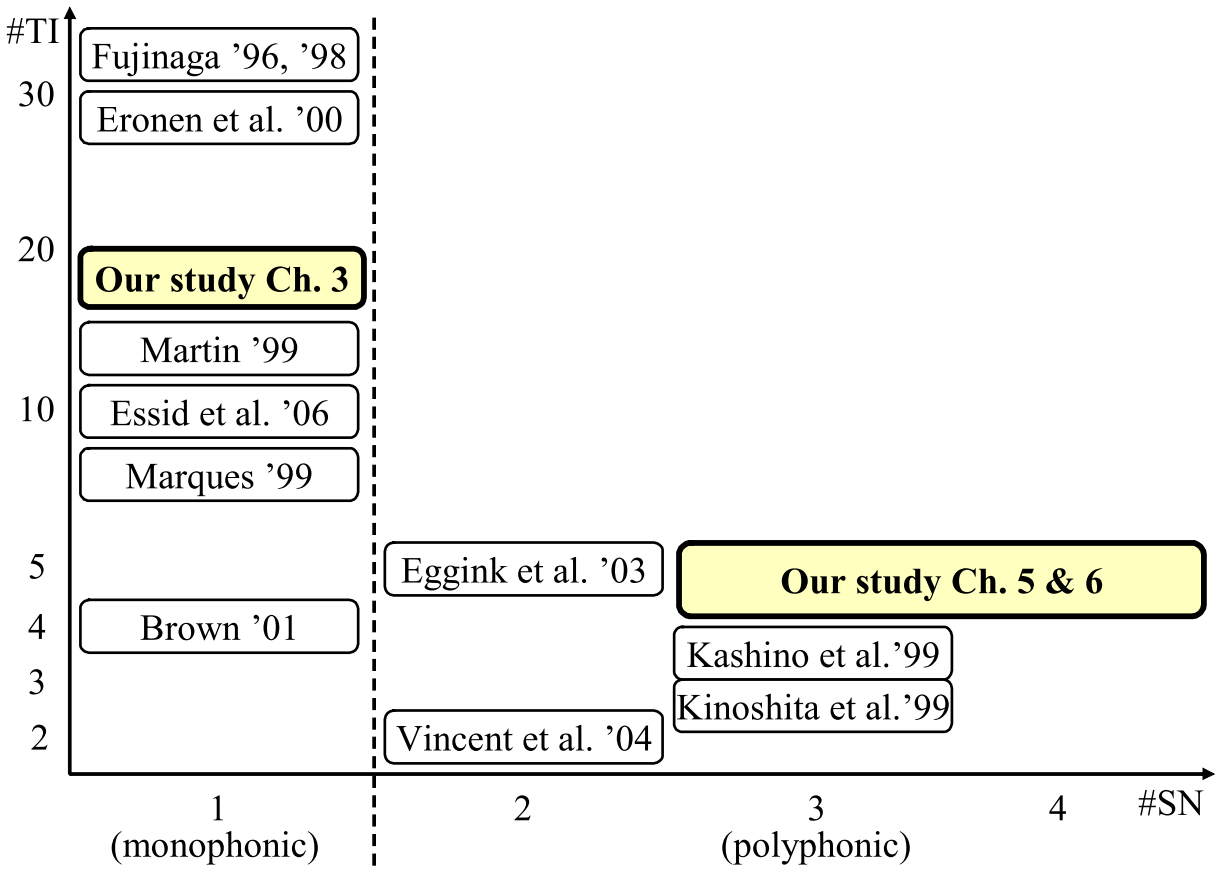


Figure 2.3: Comparison of the numbers of simultaneous notes (#SN) and target instruments (#TI) among our study and previous studies (only several typical studies are mentioned here).

recognized is one, the recordings are called solo or monophonic. If #SN is greater than one, the overlapping of common-frequency partials of different instrument sounds often occurs. Because the overlapping occurs more often as #SN increases, #SN gives an indication of the difficulty of instrument recognition. In addition, instrument recognition is more difficult as #TI is increased in general. The chance rate, which is an expectation of the success rate when instruments are determined at random, becomes lower as #TI is greater. #SI and #TI among our study and previous studies are compared in Figure 2.3. We deal with solo sounds of 19 instruments in Chapter 3 and duo, trio, and quartet music played on instruments chosen from five instrument candidates in Chapter 5. We consider these state-of-the-art.

Another factor affecting the difficulty of instrument recognition is whether the correct data of F0s etc. are given in advance. Especially for polyphonic music, as previously described, F0 estimation is still a challenging problem. Most previous studies dealing

with polyphonic music therefore manually gave the correct data of F0s etc. in their experiments [37, 40]. We also give the correct data of F0s etc. in Chapter 5 to evaluate the performance of instrument recognition alone, because our focus in Chapter 5 is on how to deal with the above-mentioned overlapping problem. We have already described Kinoshita *et al.*'s report [38] that the performance of instrument recognition decreases by introducing automatic F0 estimation. To solve this problem, we need a new framework where errors of F0 estimation do not influence the performance of instrument recognition, in addition to trying to improve the F0 estimation performance. The purpose of Chapter 6 is to develop such a new framework.

Choice between synthesized music and real-performance recordings also affects the difficulty of instrument recognition. Most previous studies used synthesized music as test samples. For example, Kashino *et al.* [46] and Kinoshita *et al.* [38] tested their methods on polyphonic musical audio signals that were synthesized by mixing isolated monophonic sounds of every target instrument on a MIDI sampler. This was because information on the instrument for every note that was used as correct references in the evaluation was then easy to prepare. We also use synthesized music in Chapter 5. On the other hand, the framework proposed in Chapter 6 performs instrument recognition non-notewise and hence no notewise transcription is needed for evaluation. We therefore use real-performance recordings in the evaluation.

The problem of non-registered instruments described in Chapter 4 has not been pointed out or dealt with in previous studies on instrument recognition although a similar problem is known as the out-of-vocabulary problem in speech recognition studies. The purpose of this chapter is to point out and tackle this new problem rather than to improve the recognition rate. Our solution of category-level recognition is tested only on solo sounds in our experiments, but its concept can be applied to polyphonic music with no modification.

Chapter 3

F0-dependent Timbre Modeling

In this chapter, we propose an F0-dependent multivariate normal distribution as a solution to the pitch dependency of timbres. We then discuss acoustic features for musical instrument recognition. After that, we report the results of experiments on the F0-dependent multivariate normal distribution using a solo musical instrument sound database.

3.1 Introduction

One of the main difficulties in musical instrument recognition for both monophonic and polyphonic music is the fact that acoustic features depend on the pitch. Compared to other sounds including human voices, timbres of musical instruments are obviously affected by the pitch due to their wide range of pitch. For example, the pitch range of the piano covers over seven octaves. This is why it is indispensable to cope with this *pitch dependency* of timbres to attain accurate musical instrument recognition (Figure 3.1).

The pitch dependency of timbres, however, has not been explicitly dealt with in previous studies. Most previous studies used sounds of every pitch within the pitch range of each target instrument for training. Also, Eronen and Klapuri [23] and Kashino *et al.* [46] treated F0 as an element of the feature vector. However, the quantitative modeling of how acoustic features vary according to the pitch has not been investigated.

In this chapter, we extend a multivariate normal distribution to represent the pitch-dependent distributions of musical instrument sounds in a feature space. The key idea behind this is to approximate the pitch dependency of each feature representing timbres of musical instrument sounds as a function of fundamental frequency (F0). The approximate function represents the relationship between the F0 and each feature and hence can be considered to represent the position (mean) of the distribution of the feature at each F0.

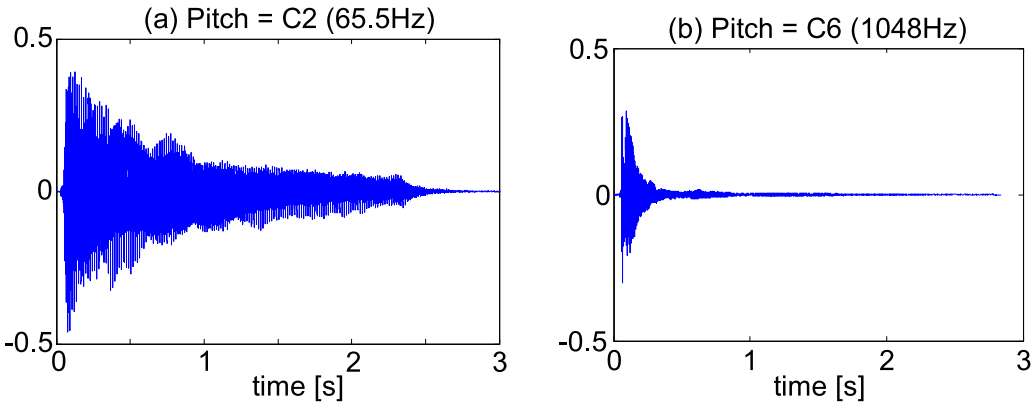


Figure 3.1: Example of the timbre of musical instruments (piano) varying according to the pitch. Whereas the power of the piano sound slowly decay at a lower pitch as shown in (a), the speed of the decay is very high at a high pitch as shown in (b). Such a tendency is not limited to the decay of the power, and every acoustic features of musical instruments depend on the pitch in general.

This function is therefore introduced as the mean vector of the extended distribution. This extended distribution is called *F0-dependent multivariate normal distribution*.

3.2 F0-dependent Multivariate Normal Distribution

Musical instrument identification in this chapter is basically performed based on the Bayes decision theory under the assumption that the given audio signal has only a single isolated monophonic tone. Let $\Omega = \{\omega_1, \cdots, \omega_m\}$ be the set of target instruments and let $\boldsymbol{x} = (x_1, \cdots, x_d)'$ be the vector consisting of acoustic features, $x_1, \cdots, x_d$, extracted from the given audio signal, where $'$ denotes the transposition operator. The problem is then formulated as maximization of $p(\omega_i|\boldsymbol{x})$, that is,

$$\hat{\omega} = \underset{\omega_i \in \Omega}{\operatorname{argmax}}\, p(\omega_i|\boldsymbol{x}) = \underset{\omega_i \in \Omega}{\operatorname{argmax}}\, \frac{p(\boldsymbol{x}|\omega_i)p(\omega_i)}{\sum_{\omega_j \in \Omega} p(\boldsymbol{x}|\omega_j)p(\omega_j)} = \underset{\omega_i \in \Omega}{\operatorname{argmax}}\, p(\boldsymbol{x}|\omega_i)p(\omega_i),$$

where $p(\boldsymbol{x}|\omega_i)$ is a probability density function (PDF) and $p(\omega_i)$ is the a priori probability with respect to the instrument ω_i. The PDF $p(\boldsymbol{x}|\omega_i)$ is calculated by analyzing the distribution of a large number of audio data of the instrument ω_i prepared in advance (called *training data*). However, tone features at different pitches, in general, have different positions (means) of distributions in the feature space. The F0-dependent multivariate normal distribution is proposed to represent this dependency of the distribution on the pitch. It has two parameters: an *F0-dependent mean function* and an *F0-normalized covariance*.

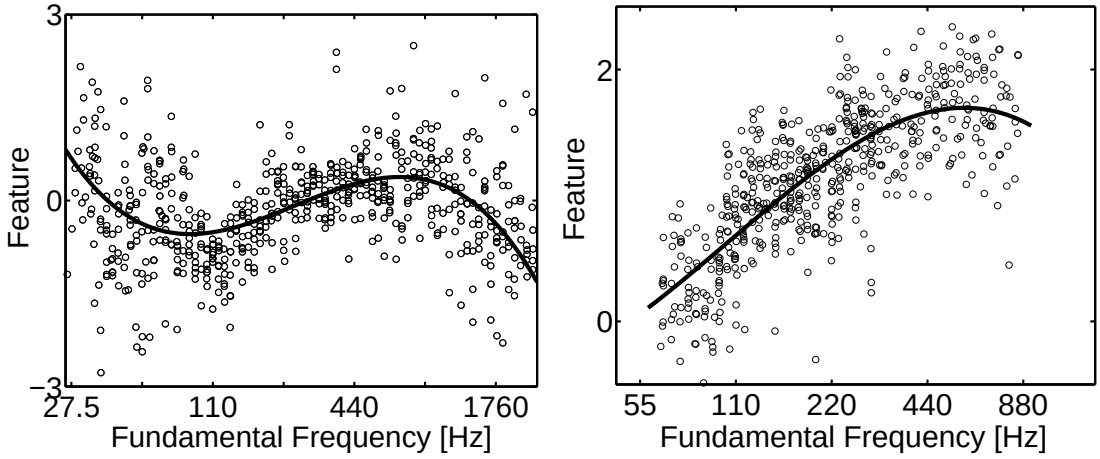


Figure 3.2: Examples of F0-dependent mean functions. Left: piano's 4th basic vector. Right: Cello's 1st basic vector. The basic vectors are obtained by applying PCA and LDA.

The former represents the pitch dependency of features and the latter represents the non-pitch dependency. Approximating the mean of the distribution as a function of F0 makes it possible to model how the features will vary according to the pitch with a small set of parameters.

3.2.1 Parameters of F0-dependent multivariate normal distribution

The following two parameters of the F0-dependent multivariate normal distribution $\mathcal{N}_{\mathrm{F0}}(\boldsymbol{\mu}_i(f), \Sigma_i)$ are estimated for each instrument ω_i.

- **F0-dependent mean function $\boldsymbol{\mu}_i(f)$**

  For each element of the feature vector, the pitch dependency of the distribution is approximated as a function of F0 using the least square method. In this paper, a cubic polynomial is used.

- **F0-normalized covariance Σ_i**

  The F0-normalized covariance is calculated with the following equation:

$$\Sigma_i = \frac{1}{n_i} \sum_{\boldsymbol{x} \in \chi_i} (\boldsymbol{x} - \boldsymbol{\mu}_i(f_{\boldsymbol{x}}))(\boldsymbol{x} - \boldsymbol{\mu}_i(f_{\boldsymbol{x}}))',$$

  where χ_i is the set of the training data of the instrument ω_i and n_i is the total number. $f_{\boldsymbol{x}}$ denotes the F0 of the feature vector $\boldsymbol{x}$. Because the F0-dependent

mean function represents the pitch dependency of features, the F0-normalized covariance, obtained by subtracting the mean from each feature, eliminates the pitch dependency of features.

3.2.2 Bayes decision rule for F0-dependent multivariate normal distribution

Once the parameters of the F0-dependent multivariate normal distribution have been estimated, the Bayes decision rule is applied to identify the name of the instrument. The discriminant function is defined as a maximization problem of a posteriori probabilities $p(\omega_i|\boldsymbol{x}; f)$ such as the following equation:

$$\begin{aligned}\hat{\omega} &= \underset{\omega_i}{\operatorname{argmax}}\, p(\omega_i|\boldsymbol{x}; f) \\ &= \underset{\omega_i}{\operatorname{argmax}}\, \frac{p(\boldsymbol{x}|\omega_i; f)p(\omega_i)}{p(\boldsymbol{x})} \\ &= \underset{\omega_i}{\operatorname{argmax}}\, p(\boldsymbol{x}|\omega_i; f)p(\omega_i) \\ &= \underset{\omega_i}{\operatorname{argmax}}\, \{\log p(\boldsymbol{x}|\omega_i; f) + \log p(\omega_i)\}, \end{aligned} \tag{3.1}$$

where $\boldsymbol{x}$ is a given feature vector, $p(\boldsymbol{x}|\omega_i; f)$ is a probability density function (PDF) of the F0-dependent multivariate normal distribution and $p(\omega_i; f)$ is the a priori probability of the instrument ω_i.

The PDF of the F0-dependent multivariate normal distribution is defined by

$$p(\boldsymbol{x}|\omega_i; f) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} \exp\left\{-\frac{1}{2} D^2_{\mathrm{M}}(\boldsymbol{x}; \boldsymbol{\mu}_i(f), \Sigma_i)\right\}, \tag{3.2}$$

where d is the number of dimensions of the feature space and D^2_{M} is the squared Mahalanobis distance defined by

$$D^2_{\mathrm{M}}(\boldsymbol{x}; \boldsymbol{\mu}_i(f), \Sigma_i) = (\boldsymbol{x} - \boldsymbol{\mu}_i(f))'\Sigma_i^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_i(f)).$$

Substituting Eq. (3.2) into Eq. (3.1), thus, generates the following discriminant function:

$$\hat{\omega} = \underset{\omega_i}{\operatorname{argmax}} \left\{-\frac{1}{2} D^2_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{\mu}_i(f)) - \frac{1}{2}\log|\Sigma_i| + \log p(\omega_i; f)\right\}.$$

The a priori probability $p(\omega_i; f)$ represents whether the pitch range of the instrument ω_i includes f, that is,

$$p(\omega_i; f) = \begin{cases} 1/c & (\text{if } f \in R_i) \\ 0 & (\text{if } f \notin R_i) \end{cases}$$

where R_i is the pitch range of the instrument ω_i, and c is the normalizing factor for satisfying $\sum_i p(\omega_i; f) = 1$.

3.3 Acoustic Features

This section describes acoustic features used for recognizing musical instruments. We first extract the harmonic structure from the given audio signal and then extract 129 features from the harmonic structure. These features can be categorized into four groups: spectral, temporal, modulation, and peak kurtosis features. Whereas the spectral, temporal, and modulation features are designed based on previous studies, the peak kurtosis features are originally designed and have not been used in previous studies. In the phase of applying the Bayes decision rule, we use an 18-dimensional feature space obtained by applying dimensionality reduction techniques to the original 129-dimensional feature space, instead of directly using the original 129-dimensional feature space, because the original high-dimensional space is redundant and may cause the curse of dimensionality.

3.3.1 Preprocessing

Given a musical instrument signal, it is first analyzed by the short-time Fourier transform (STFT) with a 4096-point Hanning window for every 10 ms, and spectral peaks are extracted from the power spectrum. Then the harmonic structure $\mathcal{H}$ is obtained from these peaks. The harmonic structure $\mathcal{H}$ is given by

$$\mathcal{H} = \{(F_i(t), A_i(t)) \mid i = 1, 2, \cdots, h,\ 0 \leq t \leq T\},$$

where $F_i(t)$ and $A_i(t)$ are the frequency and amplitude of the i-th partial at time t. Frequency is represented by relative frequency where the temporal median of the F0 is 1. Above, h is the number of harmonics, and T is the note duration. In the current implementation, h is 30.

3.3.2 Feature Extraction

The following features are extracted.

Spectral Features

1 *Spectral centroid (SC)*

The SC is given by the following equation:

$$\mathrm{SC} = \sum_{i=1}^{h} \overline{A_i} \cdot \overline{F_i} \Big/ \sum_{i=1}^{h} \overline{F_i},$$

where $\overline{A_i} = \mathrm{median}_t\, A_i(t)$ and $\overline{F_i} = \mathrm{median}_t\, F_i(t)$.

This feature is also known as *brightness* and is commonly used in most existing studies [20, 23, 29, 32, 33, 35, 36, 38, 41, 46, 50] as well as studies on genre classification [88] and mood detection [101]. It has also been adopted in the MPEG-7 Audio Description Framework [90].

2–30 *Relative cumulative powers (RCP)*

The RCPs are given by the following equation:

$$\mathrm{RCP}(k) = \sum_{i=1}^{k} \overline{A_i} \Big/ \sum_{i=1}^{h} \overline{A_i} \qquad (k = 1, 2, \cdots, h-1)$$

These features are related to the spectral rolloff [32, 53, 88, 101]. The spectral rolloff is the index of the frequency at which the relative cumulative powers reach to a given value whereas the RCPs are relative cumulative powers themselves.

31 *Odd/even power ratio (OER)*

The OER is calculated by the following equation:

$$\mathrm{OER} = p\left(|X| \leq \log \sum_{i=1}^{h'} \overline{A_{2i}} - \log \sum_{i=1}^{h'} \overline{A_{2i-1}}\right),$$

$$p(|X| \leq z) = \int_{-z}^{z} (1/\pi\sqrt{2s^2}) \exp(-x^2/2s^2) dx,$$

where $h' = \lfloor h/2 \rfloor$ ($\lfloor \cdot \rfloor$ represents the floor function) and $s = 100$ in the current implementation. Using the cumulative distribution function for a normal distribution, which is described as $p(\cdot)$ above, is because the feature should be finite even if the amplitudes of all odd or even components are zero.

This feature has been commonly used in instrument recognition [20, 29, 38, 46, 50] because it is well known that some instruments such as the clarinet have low power in even partials. Since the F0 information is necessary to calculate this feature, this has not been used in some studies that do not use F0 estimation.

32–40 *The number of stably existing partials (NEP)*

The NEP is calculated as the number of partials the duration of which is $p\%$ longer than the longest duration ($p = 10, 20, \cdots, 90$) as follows:

$$\mathrm{NEP}(p) = n\left(\left\{ i \in \{1, \cdots, h\} \;\middle|\; \mathrm{len}(A_i) > \frac{p}{100} \max_j (\mathrm{len}(A_j)) \right\}\right),$$

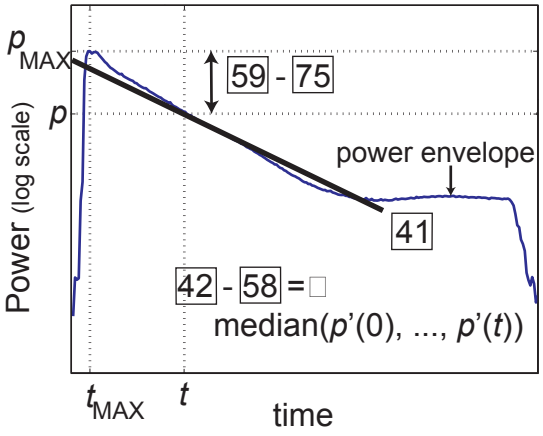


Figure 3.3: Example of temporal features (piano, C4, forte), where $p'(t)$ is a differential of the power at time t.

$$\mathrm{len}(A_i) = n\left(\{t \in [0, T] \mid A_i(t) > A_\theta\}\right),$$

where $n(X)$ is the number of the elements in the set X, A_θ is an experimentally determined threshold. The NEP was not commonly used in previous studies.

Temporal Features

41 *The power decay speed (PDS)*

The PDS is calculated as the gradient of the straight line approximating the power envelope using the least square method (Figure 3.3).

This feature is useful for distinguishing decayed instruments such as the piano and sustained instruments such as the flute and violin. A similar feature named "post-onset slope of amplitude decay" or "slope of line fitted into rms-energy curve after attack" has therefore been used in [20, 23]. Another example of features useful for distinguishing decayed and sustained instruments is *temporal centroid* [29, 50], which is also part of the MPEG-7 Audio Description Framework [90].

42–58 *Average differential of the power envelope (ADP)*

The ADP is calculated after extracting initial t-sec bits ($t = 0.15, 0.20, \cdots, 0.95$) as follows:

$$\mathrm{ADP}(t) = \underset{0 \le \tau \le t}{\mathrm{mean}}(A(\tau + dt) - A(\tau)), \qquad A(t) = \sum_{i=1}^{h} A_i(t).$$

This feature represents the strength of the attack given t is a small value, while it represents the tendency of temporal variations after the attack part given t is a large value. To represent the strength of the attack, *attack time* (or *onset duration*) has commonly been used [20, 23, 29, 50] (the MPEG-7 Audio Description Framework also includes *LogAttackTime*). We however does not use it because strictly determining the boundary of the attack and sustain/release parts is not so easy. A similar feature to the ADP has been used in [38].

59–75 *Relative power (RP)*

The RP is given as the ratios of the powers at the t-sec after the onset time ($t = 0.15, 0.20, \cdots, 0.95$) to the maximum power as follows:

$$\mathrm{RP}(t) = A(t) \Big/ \max_{0 \le \tau \le T} A(\tau)$$

Modulation Features

76, 77 *The amplitude and frequency of AM*

78, 79 *The amplitude and frequency of FM*

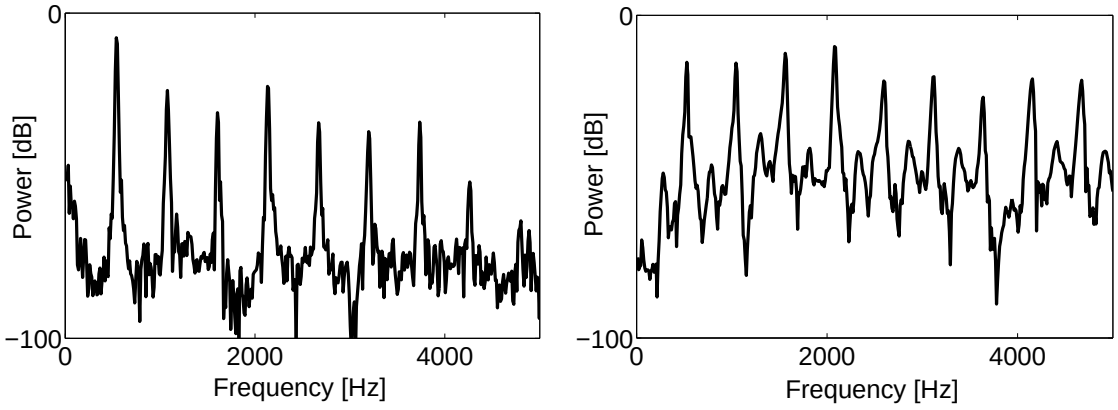
The amplitude of each modulation is calculated as the inter-quartile range (IQR) of the difference between the original envelope and the sufficiently smoothed envelope. The smoothing method proposed by Savitzky and Golay [102] is used. The frequency of each modulation is calculated as the zerocrossing rate of the temporal differential of the envelope. Those of other modulations are calculated in the same way.

Features related to AM and FM have been used in some existing studies [20, 23, 32, 41], but the methods for calculating them are completely different among different studies.

80, 81 *The amplitude and frequency of the spectral centroid modulation*

82–107 *The amplitude and frequency of the k-th MFCC modulation*

These modulations are not as common as AM or FM but the centroid modulation has also been used in [20]. Whereas MFCCs or ceptral coefficients themselves have commonly been used [21, 22, 28, 29, 41, 50], but the modulations of MFCCs have not been used.



(a) Recorder (C5, Forte) (b) Alto Sax (C5, Forte)

Figure 3.4: Power right after the onset. The kurtosis of spectral peaks of sounds containing large non-harmonic components such as (b) tends to be low compared to (a).

Peak Kurtosis Features

108–118 *Temporal average of the kurtosis of spectral peaks in each partial*

119–129 *The amplitude of the temporal modulation of the kurtosis of spectral peaks in each partial*

The kurtosis of spectral peaks is related to how large non-harmonic components are included. If a sound has small non-harmonic components, spectral peaks will have large kurtosis. If it has large non-harmonic components, they will have small kurtosis (see Figure 3.4). The kurtosis of spectral peaks can therefore be used for modeling the degree of incorporation of non-harmonic components, which has not been considered in previous studies.

3.3.3 Dimensionality Reduction

After the feature extraction, the feature space is standardized and then the dimensionality of it is reduced by two methods: the 129-dimensional feature space is reduced to a 79 dimensional one by principal component analysis (PCA) and then it is further reduced to an $(m-1)$-dimensional one by (Fisher's) linear discriminant analysis (LDA). PCA generates new feature axes by linear combinations of the original feature axes so that the new feature axes are uncorrelated, and therefore it transforms a feature space into a lower-dimensional one reducing the redundancy incorporated in the original feature space. LDA, on the other hand, is a dimensionality reduction method based on the ratio of the

between-class scatter to the within-class scatter, called Fisher's criterion, and thus take the separability of the classes into account. Using LDA can therefore be expected to improve the performance of instrument recognition. Using PCA before LDA is because it is better that features as inputs of LDA have lower correlation. The feature space generated by LDA is $(m-1)$-dimensional for m-class data set. Since we deal with 19 instruments here, the feature space becomes 18-dimensional.

Dimensionality reduction has commonly been used also in existing studies. Kashino *et al.* [46] and Kaminskyj and Czaszejko [36] used PCA. Livshin [29] used LDA. Agostini *et al.* [33] used quadratic discriminant analysis (QDA) and canonical discriminant analysis (CDA). Eronen [30] used independent component analysis (ICA). Another approach for reducing the redundancy is feature selection. This approach, as the name implies, selects some important features and removes the other redundant features instead of making new feature axes by combinations. Gradual Descriptor Elimination (GDE), proposed by Livshin*et al.* [29], determines the feature to be removed based on the result of LDA and removes it iteratively until the number of the remaining features is n (manually given). Inertia ratio maximization using feature space projection (IRMFSP), proposed by Essid *et al.* [32, 41], appends the additional feature maximizing Fisher's criterion iteratively, as the details have been described in Section 2.1.1. Fujinaga [24, 25] determined the feature set that shows the best recognition rate using the genetic algorithm (GA), the details of which have also been described in Section 2.1.1.

3.4 Experiments

3.4.1 Experimental Conditions

We conducted experiments on musical instrument recognition for investigating improvement of the performance by the proposed method. We obtained the recognition rates by the commonly used multivariate normal distribution (called *baseline*) and by the proposed F0-dependent multivariate normal distribution, and compared them.

The benchmark used for evaluation is a subset of the *"RWC Music Database: Musical Instrument Sound"* (RWC-MDB-I-2001) [103], which is a large musical instrument sound database available to researchers around the world. This subset summarized in Table 3.1 was selected by the quality of recorded sounds and consists of 6,247 solo tones of 19 orchestral instruments. All data are sampled at 44.1 kHz with 16 bits. We first divided

Table 3.1: Contents of the database used in this paper.

Instrument name (Abbrev.)	pitch range	# of tones	# of individuals	Intensity	Articulation
Piano (PF)	A0–C8	508	3	Forte, normal & piano	normal only
Classical Guitar (CG)	E2–E5	696			
Ukulele (UK)	F3–A5	295			
Acoustic Guitar (AG)	E2–E5	666			
Violin (VN)	G3–E7	528			
Viola (VL)	C3–F6	472			
Cello (VC)	C2–F5	558			
Trumpet (TR)	E3–A♯6	151	2		
Trombone (TB)	A♯1–F♯5	262	3		
Soprano Sax (SS)	G♯3–E6	169			
Alto Sax (AS)	C♯3–A5	282			
Tenor Sax (TS)	G♯2–E5	153			
Baritone Sax (BS)	C2–A4	215			
Oboe (OB)	A♯3–G6	151	2		
Faggoto (FG)	A♯1–D♯5	312	3		
Clarinet (CL)	D3–F6	263			
Piccolo (PC)	D5–C8	245			
Flute (FL)	C4–C7	134	2		
Recorder (RC)	C4–B6	160	3		

the whole data into 10 groups, and then repeated the following step 10 times: each time, we left out one of the 10 groups for training and used the omitted one for testing. That means that nine tenths of the data listed in Table 3.1 were used for calculating F0-dependent mean functions and F0-normalized covariances. This experiment technique is called 10-fold cross validation.

We evaluated the category-level performance of our method, because the category of instruments is useful for some applications including music retrieval. For example, when

Table 3.2: Categorization of 19 instruments.

Category	Instruments (abbreviation)
Piano	Piano (PF)
Guitars	Classical Guitar (CG), Ukulele (UK), Acoustic Guitar (AG)
Strings	Violin (VN), Viola (VL), Cello (VC)
Brasses	Trumpet (TR), Trombone (TB)
Saxophones	Soprano Sax (SS), Alto Sax (AS), Tenor Sax (TS), Baritone Sax (BS)
Double Reeds	Oboe (OB), Faggoto (FG)
Clarinet	Clarinet (CL)
Air Reeds	Piccolo (PC), Flute (FL), Recorder (RC)

a user wants to find a piece of piano solo on a music retrieval system, the system can reject pieces containing instruments of different categories, which can be judged without identifying individual instrument names. We adopted the categories of musical instruments summarized in Table 3.2, which are determined based on the sounding mechanisms of instruments and existing studies [20, 23].

3.4.2 Experimental Results

Table 3.3 summarizes recognition rates by both the *baseline* and *proposed* methods. The proposed F0-dependent method improved the recognition rates at the individual-instrument level from 75.73% to 79.73% and at the category level from 88.20% to 90.65% on average. It also reduced the recognition errors by 16.48% and 20.67% on average at the individual-instrument and category levels, respectively.

We confirmed the significance of the results using t-test (one-tailed). Let the difference of the recognition rates of the two methods for each instrument be d_i $(i = 1, \cdots, m)$. The test statistic is then given by

$$t_0 = \frac{|\bar{d}|}{\sqrt{\sum_i (d_i - \bar{d})^2 \Big/ m(m-1)}},$$

where $\bar{d}$ is the average of $d_1, \cdots, d_m$. The test statistics calculated for the individual-instrument and category levels were 5.4781 and 3.9482, respectively, and both were sta-

Table 3.3: Accuracy by usual distribution (baseline) and F0-dependent distribution (proposed).

	Individual-instrument level			Category level		
	Baseline	*Proposed*	Improv.	*Baseline*	*Proposed*	Improv.
PF	74.21%	83.27%	+9.06%	74.21%	83.27%	+9.06%
CG	90.23%	90.23%	±0.00%	97.27%	97.13%	−0.14%
UK	97.97%	97.97%	±0.00%	97.97%	98.31%	+0.34%
AG	81.23%	83.93%	+2.70%	94.89%	95.65%	+0.76%
VN	69.70%	73.67%	+3.97%	98.86%	99.05%	+0.19%
VL	73.94%	76.27%	+2.33%	93.22%	94.92%	+1.70%
VC	73.48%	78.67%	+5.19%	95.16%	96.24%	+1.08%
TR	73.51%	82.12%	+8.61%	76.82%	85.43%	+8.61%
TB	76.72%	84.35%	+7.63%	85.50%	89.69%	+4.19%
SS	56.80%	65.89%	+9.09%	73.96%	80.47%	+6.51%
AS	41.49%	47.87%	+6.38%	73.76%	77.66%	+3.90%
TS	64.71%	66.01%	+1.30%	90.20%	92.16%	+1.96%
BS	66.05%	73.95%	+7.90%	81.40%	86.05%	+4.65%
OB	71.52%	72.19%	+0.67%	75.50%	74.83%	−0.67%
FG	59.61%	68.59%	+8.98%	64.74%	71.15%	+6.41%
CL	90.69%	92.07%	+1.38%	90.69%	92.07%	+1.38%
PC	77.56%	81.63%	+4.07%	89.39%	90.20%	+0.81%
FL	81.34%	85.07%	+3.73%	82.09%	85.82%	+3.73%
RC	91.88%	91.25%	−0.63%	92.50%	91.25%	−1.25%
Av.	75.73%	79.73%	+4.00%	88.20%	90.65%	+2.45%

Baseline: Usual (F0-independent) distribution

Proposed: F0-dependent distribution

tistically significant with the significance level of 0.05%, where the critical region was $(3.9217, \infty)$.

3.4.3 Discussions for PCA

The factor loadings for principal components are shown in Figure 3.5. From Figure 3.5 (a) and (b), we can see that the first principal component is highly related to the NEPs (32–40) and the temporal features (41–75) while the second principal component is highly related to harmonics-related features (2–30). It is well known that the two main factors of musical instrument timbres are harmonics and temporal variations of power.This knowledge has been confirmed by our results. The 3rd principal component is related to the amplitudes of the temporal modulations of MFCCs and peak kurtosis features (82–94, 119–129). The 4th principal component is related to the frequencies of MFCC modulations (95–107). The 5th principal component is related to MFCCs and peak kurtosis features (95–129). The 6th principal component is related to the amplitudes of temporal modulations of MFCCs and peak kurtosis features (95–107, 119–129). The factor loadings for the peak kurtosis features, which have been design to capture the degree of incorporation of non-harmonic components, are high for a number of principal components. This result implies the importance of dealing with non-harmonic components, which has not been dealt with even though its importance has been pointed out

3.4.4 Discussions for LDA

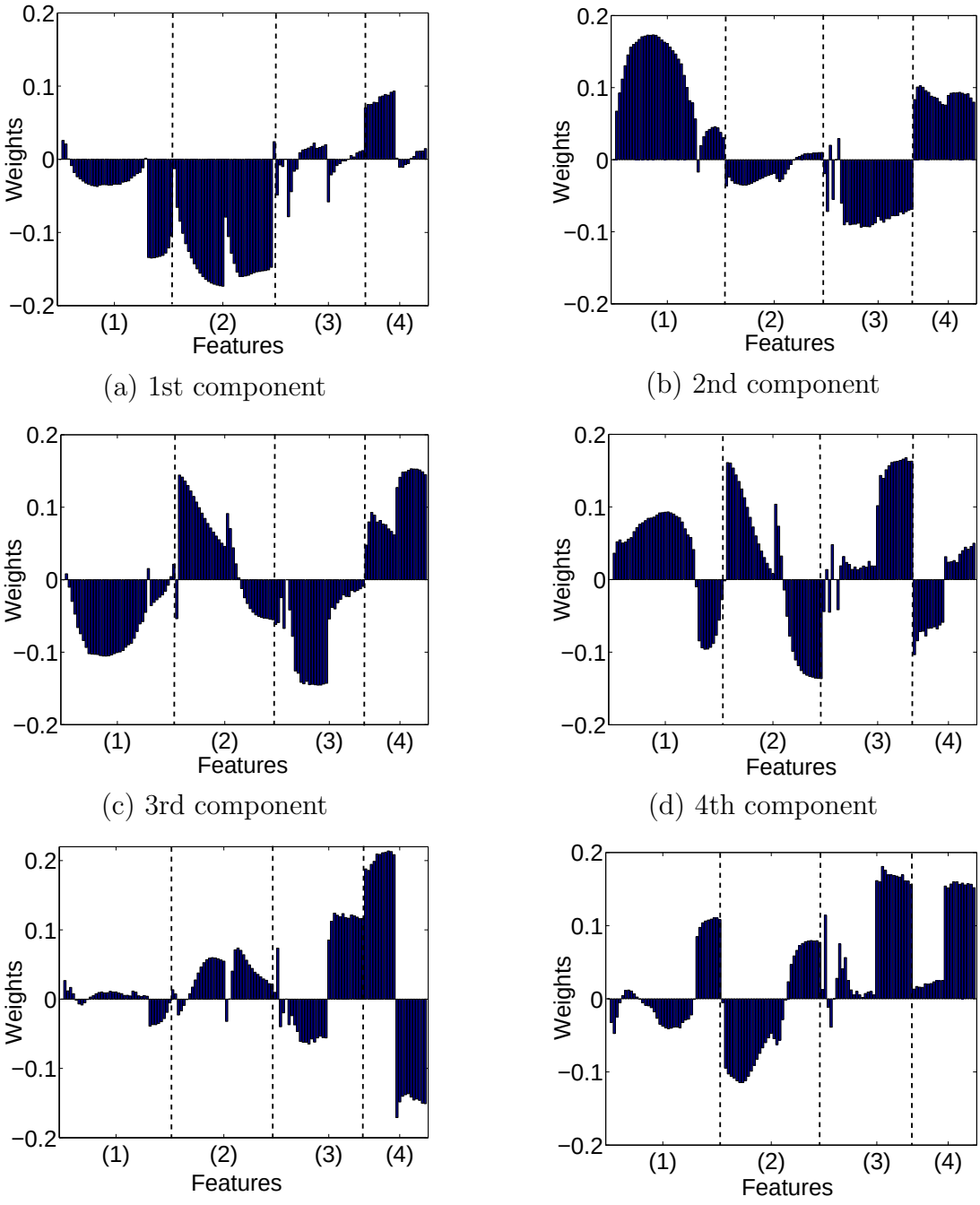
The weights of the features of each category for new feature axes in the transformation matrix obtained by combining PCA and LDA are shown in Table 3.4. Observations about these can be summarized as follows:

1. **Spectral features**

   In addition that the weight of RCP(1) (2) for the 9th axes were large (0.3586), the weights of the NEPs (32–40) for the 5th, 7th, 9th, and 10–13th axes were large (0.2721–0.4363). This result matches the old knowledge that the spectra of musical instrument sounds characterize their timbres.

2. **Temporal and modulation features**

   The weights of the PDS (41) for the 3rd and 4th axes were large (0.5977 and -0.2578, respectively). The weight of ADP(0.15) (42) for the 10-th axes was



(a) 1st component

(b) 2nd component

(c) 3rd component

(d) 4th component

(e) 5th component

(f) 6th component

Figure 3.5: Factor loadings of PCA

Table 3.4: Excerpt of weights of features in transformation matrix

	Features and their weights
1st axes	73 (0.2701), 74 (0.3220), 75 (0.3926), 79 (−0.3204), 81 (0.2559)
2nd	40 (−0.2721), 76 (0.4425), 78 (0.3554), 82 (−0.2771),
3rd	41 (0.5977), 109 (0.2607)
4th	41 (−0.2578), 79 (−0.2917), 109 (0.2944)
5th	40 (0.4286), 78 (0.3219), 108 (0.5400)
6th	76 (−0.2755), 108 (−0.4529)
7th	40 (0.3974), 108 (−0.4576)
8th	76 (0.3378), 85 (0.2614), 108 (−0.4541)
9th	2 (0.3586), 40 (−0.2783), 84 (0.4525)
10th	40 (0.2887), 42 (−0.3200), 108 (−0.3292), 109 (0.4508)
11th	32 (0.4363), 36 (−0.2837), 109 (−0.2732)
12th	39 (0.2794), 78 (0.3174), 81 (0.2704)
13th	40 (0.3521), 120 (−0.2522)
14th	76 (−0.3484), 77 (0.4201)

−0.3200, and those of RP(0.85), RP(0.90), and RP(0.95) (73–75) for the 1st axes were between 0.2701 and 0.3926. In addition, the weights of AM- and FM-related features (76–79) for many axes were large (0.4755–0.4425). These results mean that the temporal variations are important in instrument recognition. It is known that the temporal variations are also important for humans' timbre perception. For example, it is difficult even for humans to identify instruments in the case of reverse playback even though their stationary spectra are the same [104].

3. **Peak kurtosis features**

   The weights of peak kurtosis features, especially 108 and 109, were high for many axes. This implies the importance of dealing with non-harmonic components in recognizing instruments as described in the previous section.

3.4.5 Discussions for Experimental Results

Observations about the experimental results are summarized below:

1. The recognition rate for the piano was improved by 9.06%, and its recognition errors were reduced by 35.13%. This big improvement was attained since their pitch dependency is salient due to their wide range of pitch.

2. The recognition rates for the classical guitar, ukulele, and recorder were not improved. This would be because there were no room to improve the recognition rates for these instruments due to their sufficiently high recognition rates with the baseline method.

3. The category-level recognition rates for the guitar and strings were better (94.92–99.05%) than other instruments. This is because similar instruments did not exist in the other categories whereas reed instruments were divided into several categories, which makes the timbres between these categories similar.

4. The recognition rates for the four types of saxophones at the individual-instrument level (47–73%) were lower than those at the category level (77–92%). This is because sounds of these saxophones were quite similar. In fact, Martin reported that sounds of various saxophones are very difficult even for humans (music experts) to discriminate [20].

3.5 Comparison with k-NN Classifier

The effect of the Bayes decision rule in musical instrument recognition was evaluated by comparing with the k-NN rule (k-nearest neighbor rule; $k = 3$ in this paper) with/without LDA. Three variations of the dimensionality reduction are examined:

(a) Reduction to 79 dimension by PCA,

(b) reduction to 18 dimension by PCA, and

(c) reduction to 18 dimension by PCA and LDA.

The last one is adopted in the proposed method.

The experimental results listed in Table 3.5 showed that the proposed Bayes decision rule performed better in average than the 3-NN rule. Some observations are as follows:

- The Bayes decision rule with 79-dimension showed poor performance for Acoustic Guitar (AG), Trumpet (TR), Soprano Sax (SS), Tenor Sax (TS), Oboe (OB), and

Table 3.5: Accuracy by k-NN rule and the Bayes decision rule.

	k-NN rule ($k = 3$)			Bayes decision rule		
	79-Dim.	18-Dim.		79-Dim.	18-Dim.	
	PCA		PCA&LDA	PCA		PCA&LDA
PF	53.94%	46.46%	63.39%	55.91%	59.06%	83.27%
CG	79.74%	77.16%	75.72%	98.28%	97.27%	90.23%
UK	94.58%	92.54%	97.63%	67.12%	80.00%	97.97%
AG	95.05%	92.79%	97.00%	19.97%	44.14%	83.93%
VN	47.73%	46.02%	45.83%	89.58%	84.47%	73.67%
VL	55.93%	54.24%	61.86%	71.19%	79.24%	76.27%
VC	86.20%	85.84%	84.23%	45.16%	30.82%	78.67%
TR	36.42%	38.41%	47.02%	41.72%	72.85%	82.12%
TB	70.99%	54.58%	77.86%	75.19%	78.24%	84.35%
SS	23.08%	14.20%	24.85%	48.52%	66.86%	65.89%
AS	37.59%	29.79%	40.43%	72.70%	41.84%	47.84%
TS	62.09%	66.01%	68.63%	30.07%	61.44%	66.01%
BS	68.84%	67.91%	66.98%	55.35%	54.42%	73.95%
OB	47.68%	48.34%	49.01%	43.71%	81.46%	72.19%
FG	64.10%	65.06%	74.36%	40.38%	30.12%	68.59%
CL	93.45%	87.93%	93.10%	95.51%	93.45%	92.07%
PC	84.08%	84.90%	84.08%	63.27%	58.37%	81.63%
FL	88.06%	72.39%	94.03%	35.82%	84.33%	85.07%
RC	97.50%	93.75%	97.50%	85.00%	96.25%	91.25%
Av.	70.27%	66.98%	72.53%	62.11%	66.50%	79.73%

Flute (FL), since there are insufficient training data to estimate parameters of a 79-dimensional normal distribution. For small training sets with 79-dimension, k-NN is superior to the Bayes decision rule.

- LDA with the Bayes decision rule improved the accuracy of musical instrument recognition from 66.50% to 79.73% on average. Although it seemed that PCA with 79-dimension performed better than LDA for Classical Guitar (CG), Violin (VN), and Alto Sax (AS), the cumulative performance of LDA for the categories of strings and saxophones is better than that of PCA.

- We did not conduct the experiment using only LDA. This is because LDA cannot be applied to features that are highly correlative: the inverse matrix of the with-in covariance, which is used by LDA, is not accurately calculated when the feature space includes some highly correlative dimensions. Because PCA not only reduces the dimensionality but also orthogonalizes the feature space, for our features, some of which are highly correlative, using PCA before LDA is effective.

3.6 Comparison with Approach of Appending F0 to Feature Vector

An alternative approach to dealing with the pitch dependency of timbres can be to append F0s to feature vectors. We compared this approach with the proposed method in this section. We conducted experiments on musical instrument recognition using the following methods:

(a) Using usual (F0-independent) multivariate normal distributions without appending F0s to feature vectors (baseline),

(b) Using usual (F0-independent) multivariate normal distributions with the 18-dimensional feature space obtained by applying PCA and LDA to the 130-dimensional feature space where the F0 is appended,

(c) Using 4-mixture Gaussian mixture models (GMMs) with the same feature space,

(d) Using 4-mixture GMMs with the 20-dimensional feature space to which the F0 is appended after dimensionality reduction based on PCA and LDA,

(e) Using 8-mixture GMMs with the same feature space,

(f) Using the proposed method.

The other experimental conditions were the same as those of Section 3.4. The experimental results are shown in Table 3.6. The results have the following tendencies:

- The average recognition rate for (f) was highest.
- Whereas there were no instruments for which the recognition rates were lowered more than 1% in Case (f), the recognition rates for six instruments were lowered more than 2% and those for two instruments were lowered more than 6% in Case (e). The instruments for which the recognition rates were lowered tended to have a small number of training data.
- When we used dimensionality reduction after appending the F0, the performance was not improved.

3.7 Conclusion

We conclude this chapter as follows:

- We proposed a musical instrument recognition method that deals with the pitch dependency of timbres by approximating it as a function of F0. Although the pitch is an important factor of feature variations and is peculiar to musical instrument sounds due to their wide pitch ranges, investigation and modeling of the relationship between the pitch and feature variations have not been attempted in previous studies.
- We explained 129 acoustic features used for musical instrument recognition. Some features have also been commonly used in previous studies such as the spectral centroid, but other features have not been used such as the peak kurtosis features. Through analysis of the results of PCA and LDA, we also discussed contributions of these features to instrument recognition.
- We reported our experimental results that attained the recognition rate of about 80% for 6,247 solo sounds of 19 instruments. The effectiveness of the F0-dependent

Table 3.6: Results of experiments in Section 3.6 (DR: dimensionality reduction).

	(a) Baseline	(b) Append F0 befor DR Single Gauss	(c) Append F0 before DR GMM(4mix)	(d) Append F0 after DR GMM(4mix)	(e) Append F0 after DR GMM(8mix)	(f) Proposed
PF	74.41%	72.44%	81.10% $^{(++)}$	82.48% $^{(++)}$	86.22% $^{(++)}$	83.85% $^{(++)}$
CG	90.23%	90.23%	84.91% $^{(-)}$	85.06% $^{(-)}$	86.06% $^{(-)}$	90.23%
UK	97.97%	98.31%	95.59% $^{(-)}$	97.63%	97.63%	97.97%
AG	82.43%	81.98%	81.08%	80.48%	84.98% $^{(+)}$	83.33%
VN	69.89%	67.61% $^{(-)}$	69.69%	70.83%	71.97% $^{(+)}$	72.92% $^{(+)}$
VL	74.15%	74.79%	69.70% $^{(-)}$	71.82% $^{(-)}$	70.97% $^{(-)}$	76.69% $^{(+)}$
VC	73.30%	73.78%	71.33%	72.76%	74.19%	78.67% $^{(+)}$
TR	75.50%	74.83%	69.54% $^{(-)}$	65.56% $^{(--)}$	70.20% $^{(-)}$	82.12% $^{(++)}$
TB	77.10%	79.39% $^{(+)}$	85.50% $^{(++)}$	80.15% $^{(+)}$	86.64% $^{(++)}$	85.50% $^{(++)}$
SS	57.99%	58.58%	56.21%	63.91% $^{(+)}$	65.09% $^{(++)}$	64.50% $^{(++)}$
AS	41.49%	43.97% $^{(+)}$	59.22% $^{(++)}$	53.90% $^{(++)}$	58.51% $^{(++)}$	47.52% $^{(++)}$
TS	64.71%	64.71%	69.28% $^{(+)}$	71.90% $^{(++)}$	68.63% $^{(+)}$	65.36%
BS	66.05%	65.12%	71.16% $^{(+)}$	71.16% $^{(+)}$	73.02% $^{(++)}$	73.95% $^{(++)}$
OB	72.19%	73.51%	63.58% $^{(--)}$	62.25%	64.24% $^{(--)}$	72.19%
FG	59.62%	60.90%	59.29%	66.02% $^{(--)}$	69.87% $^{(++)}$	68.27% $^{(++)}$
CL	90.34%	90.34%	87.59% $^{(-)}$	84.48% $^{(-)}$	86.55% $^{(-)}$	91.38%
PC	77.96%	77.96%	81.22% $^{(+)}$	80.82% $^{(+)}$	86.94% $^{(++)}$	81.63% $^{(+)}$
FL	80.60%	81.34%	73.13% $^{(--)}$	73.13% $^{(--)}$	68.66% $^{(--)}$	85.07% $^{(+)}$
RC	91.88%	89.38% $^{(-)}$	88.13% $^{(-)}$	90.63%	91.88%	91.25%
Av.	75.98%	75.88%	75.91%	76.38%	78.57% $^{(+)}$	79.73% $^{(+)}$

$^{(++)}$ Improved by 6% or more from the baseline.

$^{(+)}$ Improved by 2% or more from the baseline.

$^{(--)}$ Decreased by 6% or more from the baseline.

$^{(-)}$ Decreased by 2% or more from the baseline.

multivariate normal distribution was also tested by comparing it with the usual (F0-independent) multivariate normal distribution, the k-NN classifier, and the GMM with appending the F0 to the feature vector.

Chapter 4

Category-level Recognition of Non-registered Musical Instruments

This chapter points out a new problem in musical instrument recognition, called the *non-registered instrument* problem, and provides a solution based on category-level recognition. First, it describes construction of a musical instrument taxonomy for the category-level recognition based on acoustic features using a large-scale musical instrument sound database. Next, it reports experimental results of category-level recognition of non-registered instruments based on the constructed musical instrument taxonomy.

4.1 Introduction

In this chapter, we focus on the problem of recognizing *non-registered musical instruments*, that is, recognizing musical instruments that are not contained in training data. Almost all of the previous studies have used training data containing a limited number of musical instruments and have assumed that all the instruments used in the input were contained in the training data. Because there are numerous kinds of musical instruments in the world, it is impossible to prepare training data containing all of them. In addition, recent development of digital audio technology has made it possible to create novel and infinite kinds of original musical sounds (from sounds similar to natural instruments to sounds of instruments that do not actually exist). It is therefore essential to deal with non-registered musical instruments when recognizing musical instrument sounds.

To solve this problem, we propose category-level recognition of the non-registered musical instruments. For example, a musical instrument sound that is similar to a violin and a viola but not the same (for example, a sound made from the two instruments using a synthesizer) is recognized as "strings." When humans listen to this sound for the first

Table 4.1: Conventional taxonomy of musical instruments.

Higher level	Middle level	Lower level	Musical instruments*
Strings	—	Struck strings	PF
		Plucked strings	CG, UK, AG
		Bowed strings	VN, VL, VC
Winds	Wood winds	Air reeds	PC, FL, RC
		Single reeds	SS, AS, TS, BS, CL
		Double reeds	OB, FG
	Brasses	(Rip reeds)	TR, TB
Percuss.	(omitted)	(omitted)	(omitted)

*Notation of musical instruments is defined in Table 3.1.

time, they would think "I do not know this instrument, but it must be a kind of strings." This study aims to achieve such human-like recognition on a computer.

We also discuss a *musical instrument taxonomy* for this category-level recognition. The most important requirement for the musical instrument taxonomy in category-level recognition is that it should reflect the similarity of timbres (acoustical features). However, musical instrument taxonomies satisfying this requirement have not been reported in the literature. We present a method for automatic acquisition of a musical instrument taxonomy based on the acoustical similarity of musical instruments. We call this *TimbreTree*.

4.2 TimbreTree: Musical Instrument Taxonomy based on Acoustical Similarity

The musical instrument taxonomy for category-level recognition should reflect the timbre similarity. In other words, two instruments that are close on the taxonomy should have similar timbres. Most of the commonly used taxonomies, however, do not satisfy this requirement. For example, in the taxonomy shown in Table 4.1 [105] which is designed based on sounding mechanisms and playing methods of musical instruments, both pianos (PF) and violins (VN) belong to the same group of "strings," but their timbres are quite different.

We therefore present a method for automatic acquisition of TimbreTree, which satisfies the above requirement, using a large musical instrument sound database. The rest of this section discusses problems, solutions and results of automatic acquisition of TimbreTree.

4.2.1 Problems and Our Solutions

One of the most commonly used methods for acquiring a hierarchy from feature vectors is *hierarchical clustering*. Hierarchical clustering first calculates distances between feature vectors in a feature space and then merges the closest pair of feature vectors (or clusters) into a single cluster recursively until all the feature vectors are merged into a single cluster. This method can be applied to acquiring TimbreTree, but the following two problems make it difficult to obtain reasonable results:

Problem 1 Clustering results depend on a feature space.

Problem 2 If one sound is used as a representative of each musical instrument, the clustering results also depend on the choice of the representative. This is because features of musical instrument sounds depend on various factors including pitch and differences of individuals.

To solve **Problem 1**, we use the same feature space for both recognition and clustering. Since different musical instrument recognition methods would have different feature spaces, the taxonomies appropriate for the identification methods would also be different. Our approach makes it possible to obtain the taxonomy optimized for each recognizer. To solve **Problem 2**, we apply hierarchical clustering to a multivariate normal distribution of each instrument, which is obtained from a large musical instrument sound database. By using a multivariate normal distribution, instead of a single sound, for each instrument, we can obtain the appropriate representative position of the instrument in the feature space.

4.2.2 Details of the method

TimbreTree is acquired by the following three steps:

(a) **Feature Extraction**

The features that are the same as those used for recognition are extracted. Since we use a musical instrument recognition method presented in Chapter 3, we extract

the 129 features described in Chapter 3 and then reduce dimensionality from 129 to 18 using PCA and LDA.

(b) **Calculation of the Mahalanobis Distances**

Once the distribution of each instrument ω_i in the feature space is approximated by an multivariate normal distribution, the mean vector $\boldsymbol{\mu}_i$ and the covariance matrix Σ_i of this distribution are calculated. The Mahalanobis distance $D_{\mathrm{M}}(\omega_i, \omega_j)$ of each instrument pair (ω_i, ω_j) $(\omega_i \neq \omega_j)$ is calculated by the following equation:

$$D_{\mathrm{M}}(\omega_i, \omega_j) = (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)' \Sigma_{i,j}^{-1} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j),$$

where, $\Sigma_{i,j} = (\Sigma_i + \Sigma_j)/2$, and $'$ represents the transposition operator.

(c) **Hierarchical Clustering**

Hierarchical clustering is performed using the above Mahalanobis distances. In this paper, we adopted the average-link clustering, which considers the distance between two clusters to be equal to the average distance from any member of one cluster to any member of the other.

4.2.3 Experiments on Acquisition of TimbreTree

We conducted experiments on automatic acquisition of TimbreTree using the same data as those used in Chapter 3 (Table 3.1). The TimbreTree acquired by the proposed method is shown in Figure 4.1. We obtained musical instrument categorization by merging musical instrument of which distances from each other in Figure 4.1 are less than a threshold into one cluster. Higher, middle, and lower levels in Table 4.2 show the categorization obtained when the threshold is 30, 20, and 10, respectively. Next, we acquired TimbreTree (Figure 4.2) using only a randomly chosen half of the data in Table 3.1, which will be used as training data in the next section. The musical instrument categorization shown in Table 4.3 was obtained from this tree similarly to the above.

4.2.4 Preliminary Experiment on Category-level Recognition of Registered Instruments

We conducted experiments on recognizing *registered* instruments at the category level. We compared the category-level recognition rates for the traditional categorization (Table

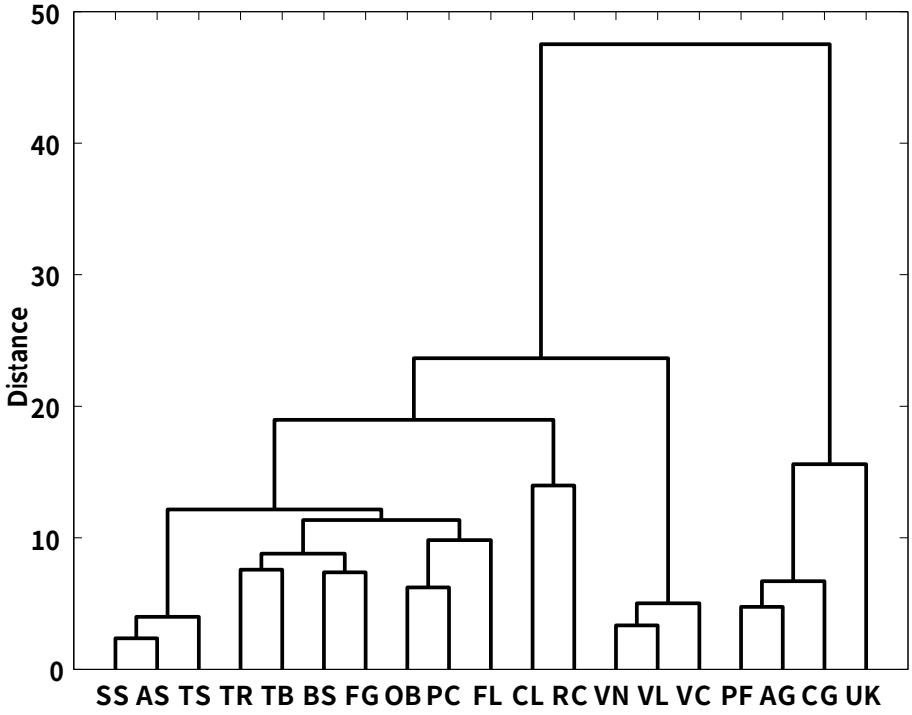


Figure 4.1: TimbreTree obtained using the proposed method (Case of using all the data in Table 3.1)

Table 4.2: Musical instrument categorization at three different levels obtained from Figure 4.1.

Higher level	Middle level	Lower level	Musical Instruments
Decayed	—	Ukulele	UK
		Others	PF, CG, AG
Sustained	Strings	—	VN, VL, VC
	Woods	Saxophones	SS, AS, TS
		Clarinet	CL
		Recorder	RC
		Brasses, etc.	TR, TB, BS, FG
		Others	OB, PC, FL

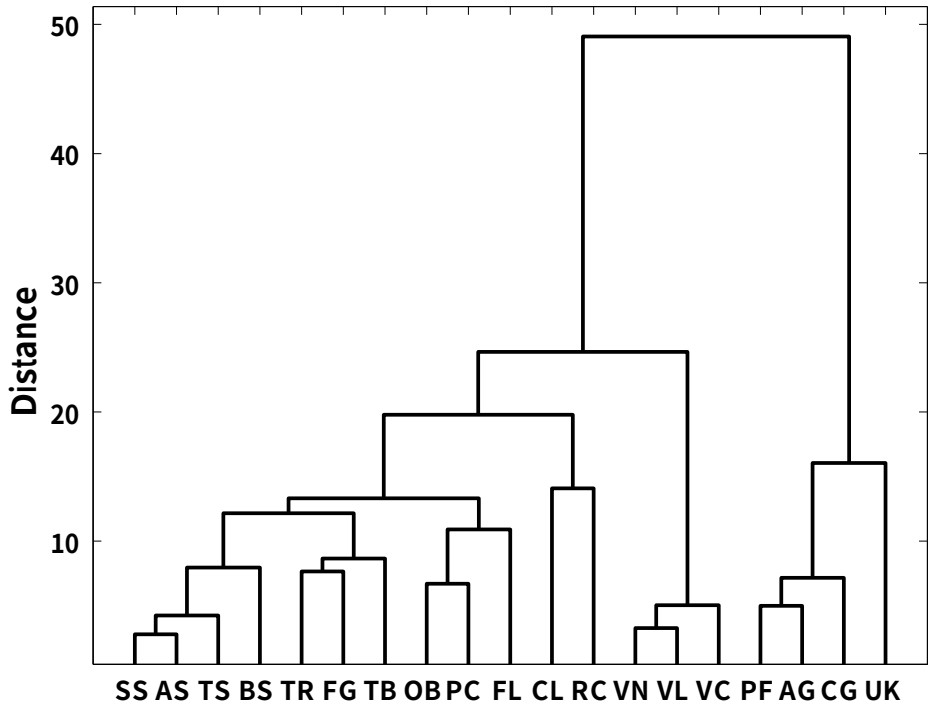


Figure 4.2: TimbreTree obtained using the proposed method (Case of using a half of the data in Table 3.1)

Table 4.3: Musical instrument categorization at three different levels obtained from Figure 4.2.

Higher level	Middle level	Lower level	Musical Instruments
Decayed	—	Ukulele	UK
		Others	PF, CG, AG
Sustained	Strings	—	VN, VL, VC
	Woods	Saxophones	SS, AS, TS, BS
		Clarinet	CL
		Recorder	RC
		Brasses, etc.	TR, TB, FG
		Others	OB, PC, FL

4.1, lower-level) and our categorization (Table 4.2, Table 4.3, both lower-level). To make the number of categories equal, the 'single reeds' category in Table 4.1 is divided to the 'saxophones' category and the 'clarinet' category. We used the musical instrument recognition method described in Chapter 3, assigning a half of the data in Table 3.1 to training and the rest to testing. The experimental results, listed in Table 4.4, show that the recognition rates with our categorization were higher than those with the traditional categorization.

4.2.5 Discussions

The results of acquiring TimbreTree are summarized below:

- **Division into decayed and sustained instruments**
  Our taxonomy divided all instruments into two categories: *decayed* and *sustained* instruments. This division matches reports on psychological acoustics [75] and manually constructed timbre-based hierarchies [20, 23]. This shows that our taxonomy approximately reflects the timbre similarity. This is one of the major differences between our taxonomy and a conventional one (Table 4.1).

- **Categories that consist of only one instrument**
  Three instruments, the ukulele, the clarinet, and the recorder, each formed a category singly at the lower level. The reason why the ukulele and the clarinet did so is that the Mahalanobis distances between them and others are large due to their peculiar characteristics. Ukuleles decay the fastest of the four decayed instruments. Clarinets have small powers of even-ordered harmonic components, especially 2nd one. On the other hand, the reason why the recorder did so is that the variance of the recorder's distribution is small. Recorders' flows are fixed by the forms of the narrow windways while flutes' flows are fixed by the forms of the players' lips. The sounds of recorders, therefore, do not vary much from player to player. This is why the variance of the recorder's distribution was small.

- **Influence of pitch range**
  In classifying wind instruments, instruments that have a similar pitch range tended to be placed into the same category. This result means that the features of musical instrument sounds depend on not only the sounding mechanisms but also the pitch. This matches the literature on psychological acoustics [75].

Table 4.4: Results of category-level identification of registered instruments.

Instr.	Individual-level	Category-level		
		Conv.	*Prop. (1)*	*Prop. (2)*
PF	80.45%	80.45%	98.12%	98.12%
CG	92.66%	96.64%	99.39%	99.39%
UK	96.73%	96.73%	96.73%	96.73%
AG	78.40%	95.73%	98.13%	98.13%
VN	71.63%	98.94%	98.94%	98.94%
VL	73.20%	92.00%	92.00%	92.00%
VC	75.18%	96.72%	96.72%	96.72%
TR	71.62%	74.32%	91.89%	82.43%
TB	74.05%	83.97%	92.37%	85.50%
SS	53.93%	78.65%	74.16%	78.65%
AS	49.17%	73.33%	69.17%	73.33%
TS	49.04%	87.50%	72.12%	87.50%
BS	67.86%	85.71%	78.57%	85.71%
OB	63.41%	70.73%	68.29%	68.29%
FG	71.23%	74.66%	75.34%	78.08%
CL	90.98%	90.98%	90.98%	90.98%
PC	80.74%	88.99%	88.99%	88.99%
FL	63.63%	66.23%	70.13%	70.13%
RC	88.88%	88.88%	88.88%	88.88%
Av.	75.98%	88.85%	90.81%	91.25%

Conv. Using the conventional categorization (Table 4.1)

Prop. (1) Using our categorization (Table 4.2, lower-level)

Prop. (2) Using our categorization (Table 4.3, lower-level)

- **Saxophones and Clarinets**

  Although saxophones and clarinets have single reeds, our results show that their sounds are not similar. This is because clarinets are cylindrical while saxophones are conical. This shape difference causes spectral differences, especially powers of even-ordered harmonic components. While conventional taxonomies such as Table 4.1 do not take these timbre differences into consideration, our taxonomy does.

- **Consistent Training Data**

  We confirmed from the preliminary experiments the importance of using the same data for constructing TimbreTree and for training an instrument recognizer. Because our method for constructing TimbreTree is completely automatically performed, the taxonomies can easily be switched according to training data for instrument recognizers.

4.2.6 Comparison with Related Work

The automatic acquisition of a musical instrument taxonomy has not been dealt with in previous studies.

In the field of acoustic psychology, various experiments on timbre similarities from the viewpoint of humans' perception have been conducted [66, 68, 73]. Such studies are important but do not necessarily match with our purpose because appropriate timbre similarity measures and musical instrument taxonomies for humans' perception and those for computational musical instrument recognition will be different.

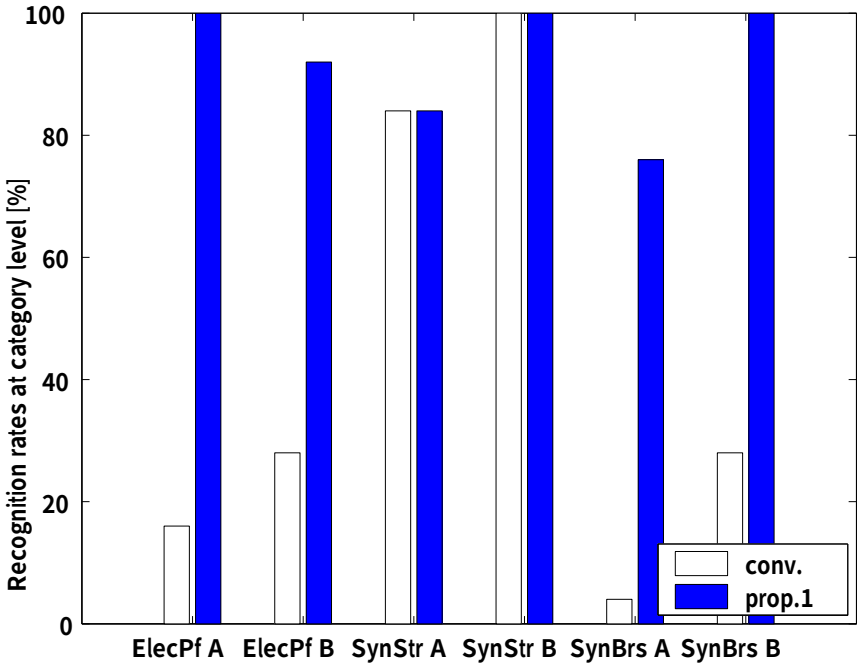
Martin [20], Eronen and Klapuri [23], and Peeters and Rodet [50] introduced a hierarchical scheme to musical instrument identification. Although the taxonomies used there partly match with ours, their taxonomies were manually designed.

Casey [106] introduced MPEG-7's framework for describing various relations of general sounds including musical instrument sounds as a tree structure. They, however, did not deal with the problem of how to automatically obtain such a tree structure.

Dubnov and Tishby [107] applied hierarchical clustering to 31 sound samples extracted from an electric instrument. They, however, used a single tone for each instrument and thus did not take into account feature variations caused by factors such as the pitch and the difference of individuals.

Table 4.5: Musical instrument sounds used as non-registered instruments

Instruments	Electric Piano (ElecPf), Synth Strings (SynStr), Synth Brass (SynBrs)
Variations	Two for each instrument
Velocity	100
Pitch range	C3–C5 (A4=440Hz)



conv. Using the conventional categorization (Table 4.1)
prop.1 Using our categorization (Table 4.2, lower-level).

Figure 4.3: Results of category-level recognition of non-registered instruments.

4.3 Category-level Recognition of Non-registered Musical Instruments

In this section, we report experiments on category-level recognition of non-registered musical instruments using the musical instrument categorization obtained by TimbreTree. We used the sounds listed in Table 3.1 as training data and electric sounds played by a MIDI tone generator (MU2000, Yamaha), listed in Table 4.5, as non-registered musical instrument sounds. Because the sounds listed in Table 4.5 are not sounds of actual

instruments, identification of the names of the instruments is impossible. The categories of these sounds, however, should be recognized. When people who know piano sounds listen to electric piano sounds for the first time, for example, they may judge that the sounds belong to the same category as the piano sounds even if they cannot identify the instrument name. The aim of the experiments here is to implement such judgment on a computer.

4.3.1 Category-level Recognition of Non-registered Instruments

We assigned all data in Table 3.1 to training and all data in Table 4.5 to testing, and conducted the experiment of recognizing the test data at the category level. Two kinds of categorization, *i.e.*, conventional categorization based on the sounding mechanisms (Table 4.1) and automatically generated categorization based on TimbreTree (Table 4.2, lower-level), were used.

The results are shown in Table 4.3. The recognition rates using TimbreTree were between 75 and 100% while the recognition rates using the conventional taxonomy were very low except for synth strings. The recognition rates using TimbreTree for all instruments were better than or equal to those using the conventional taxonomy. These results suggest that the sounding-mechanism-based categorization is unsuitable for electric sounds, since they do not have sounding mechanisms.

4.3.2 Determination of Whether Instruments Are Registered or Not

While non-registered instruments should be recognized at the category level, registered instruments should be recognized at the individual-instrument level. It means that the level of recognition should be switched and therefore it is required to determine whether the instruments of given signals are registered or not. This is equivalent to rejection of recognition results at the individual-instrument level and is performed through the following steps:

(1) Identify the instrument of a given sound at the individual-instrument level.

(2) Calculate the Mahalanobis distance from the given sound to the distribution of the above result.

(3) Judge it to be *registered* if the distance is less than a threshold, or *non-registered* if the distance is not.

To calculate the Mahalanobis distance, (a) the 23-dimensional feature space obtained by PCA (proportion value: 90%), (b) the 18-dimensional feature space obtained by PCA (proportion value: 88%), and (c) the 18-dimensional feature space obtained by both PCA and LDA (same as that used in Chapter 3) were used. We assigned a half of the data in Table 3.1 to training data, the rest to test data of registered instruments, and all of the data in Table 4.5 to test data of non-registered instruments.

Experimental results are listed in Table 4.6. In Case (a), the rate of correct determination was 85% when the threshold was 40. The rate of correctly determining registered instruments and the rate of correctly determining non-registered instruments have a trade-off relation in general, and the average of the two rates were mostly between 80 and 85% in Case (a).

When we focus on the threshold where the rate of correctly determining registered instruments was 85 or 86%, the rate of correctly determining non-registered instruments was 85% (highest) in Case (a) and 71% (lowest) in Case (c). Even though LDA is effective for classification of registered instruments, as shown in Chapter 3, it is not necessarily effective for classification of registered and non-registered instruments.

The sounds of ElecPf A were often mistakenly determined, *i.e.*, determined as a registered instrument. This is because these sounds were comparatively similar to real piano sounds. In fact, they were difficult even for humans to distinguish.

4.3.3 Flexible Musical Instrument Recognition

We finally present results of flexible musical instrument recognition, that is, individual-instrument-level recognition for registered instruments and category-level recognition for non-registered instruments. Similarly to the previous experiment, we assigned a half of the data in Table 3.1 to training data, the rest to test data of registered instruments, and all of the data in Table 4.5 to test data of non-registered instruments. We used the 23-dimensional feature space (Case (a)) for registered/non-registered determination. The threshold was 40. The experimental results listed in Table 4.7 show that our method correctly recognized 66.62% of registered instrument sounds at the individual-instrument level, 13.16% at the category level, and 77.33% of non-registered instrument sounds at the category level while distinguishing them from registered instrument sounds. The average

Table 4.6: Results of determination of registered/non-registered instruments.

Dim. Reduction*		(a) PCA(23)				(b) PCA(18)			(c) PCA+LDA(18)		
	Threshold	50	40	30	25	40	30	25	40	30	25
Registered	PF	92%	86%	79%	71%	93%	84%	79%	88%	82%	71%
	CG	94%	90%	83%	77%	95%	89%	86%	97%	92%	85%
	UK	86%	82%	68%	63%	87%	81%	73%	88%	82%	73%
	AG	91%	86%	80%	75%	90%	83%	80%	92%	86%	78%
	VN	91%	86%	73%	61%	94%	85%	76%	94%	84%	73%
	VL	95%	94%	79%	70%	97%	95%	85%	97%	93%	86%
	VC	96%	93%	89%	79%	97%	93%	92%	99%	94%	87%
	TR	94%	87%	70%	60%	96%	89%	79%	96%	92%	50%
	TB	92%	86%	75%	66%	95%	89%	84%	97%	91%	84%
	SS	96%	88%	73%	54%	96%	85%	71%	96%	94%	85%
	AS	88%	81%	58%	50%	92%	86%	76%	88%	80%	71%
	TS	80%	62%	46%	34%	80%	78%	70%	88%	70%	58%
	BS	88%	73%	63%	51%	92%	77%	69%	88%	77%	82%
	OB	87%	75%	65%	54%	87%	79%	71%	98%	85%	61%
	FG	85%	78%	68%	64%	87%	78%	74%	89%	78%	67%
	CL	92%	77%	67%	52%	90%	85%	80%	98%	90%	76%
	PC	90%	82%	67%	55%	92%	83%	77%	82%	73%	35%
	FL	88%	71%	47%	37%	96%	80%	50%	100%	88%	40%
	RC	91%	81%	69%	53%	94%	81%	72%	95%	90%	59%
	Av.	91%	85%	74%	65%	93%	86%	79%	94%	86%	72%
Non registered	ElecPf A	36%	44%	64%	76%	32%	36%	36%	24%	44%	48%
	ElecPf B	52%	84%	88%	92%	36&	52%	55%	36%	60%	76%
	SynStr A	100%	100%	100%	100%	100%	100%	100%	56%	88%	92%
	SynStr B	100%	100%	100%	100%	100%	100%	100%	40%	60%	100%
	SynBrs A	76%	80%	88%	92%	72%	84%	88%	72%	80%	84%
	SynBrs B	100%	100%	100%	100%	100%	100%	100%	76%	96%	100%
	Av.	77%	85%	90%	93%	73%	79%	81%	51%	71%	83%

*The values in the parentheses following the names of dimensionality reduction methods are the number of dimensions after the reduction.

error rates for registered and non-registered instruments were therefore 20.22 and 22.67%, respectively.

4.3.4 Discussions

Contributions and remaining issues are summarized as follows:

- We achieved recognition like "I do not the instrument name but it must be a kind of strings." This approach is effective for not only annotation of a variety of musical sounds but also other applications such as music transcription. When an input signal contains both piano sounds and piano-like but non-registered instrument sounds (*e.g.*, electric piano), previous music transcription systems could not distinguish them. Our method has made it possible to distinguish them by recognizing non-registered instruments at the category level.

- This approach will also be effective for multimedia integration. Suppose that a system recognized a musical instrument for annotating a motion picture of a music performance and its result through auditory features was "the instrument name is unknown but it is a kind of strings." If the system recognizes the name of this instrument through visual features, our approach is applicable to re-train the sounds of this instrument as a new instrument belonging to the strings category.

- Although the lower-level categories were used in all experiments, the granularity of categorization should be appropriately determined according to applications. Future work will include development of a method for determining the threshold used for obtaining categories from TimbreTree according to applications.

- Whereas the aim of this study was acquisition of a musical instrument taxonomy optimized for computational recognition, taxonomies for specifying instrument categories by humans should match with humans' intuition. We therefore plan to construct a musical instrument taxonomy based on humans' intuition using the results of acoustic psychology studies and develop a method for converting them. The problem of the conversion between different taxonomies representing the same concept is known as the *ontology problem* or *semantic mapping problem* [108].

Table 4.7: Results of handling both registered and non-registered instruments.

	Correct (a)	Correct (b)	Incorrect
PF	68.80%	17.29%	13.91%
CG	83.49%	11.62%	4.89%
UK	96.73%	—	3.27%
AG	68.27%	14.40%	17.33%
VN	61.70%	13.82%	24.48%
VL	68.80%	11.20%	20.00%
VC	69.71%	9.85%	20.44%
TR	63.51%	14.86%	21.63%
TB	63.36%	16.79%	19.85%
SS	47.19%	11.24%	41.57%
AS	40.00%	16.67%	43.33%
TS	29.81%	25.96%	44.23%
BS	49.11%	19.64%	31.25%
OB	47.56%	19.51%	32.93%
FG	56.16%	16.44%	27.40%
CL	90.98%	—	9.02%
PC	66.06%	17.43%	16.51%
FL	45.45%	19.48%	35.07%
RC	88.88%	—	11.12%
Av.	66.62%	13.16%	20.22%
ElecPf A	—	44.00%	56.00%
ElecPf B	—	76.00%	24.00%
SynStr A	—	88.00%	12.00%
SynStr B	—	100.00%	0.00%
SynBrs A	—	60.00%	40.00%
SynBrs B	—	96.00%	4.00%
Av.	—	77.33%	22.67%

Correct (a) correct at the individual-instrument level

Correct (b) correct at the category-level while rejecting the individual-instrument-level result

4.4 Conclusion

In this paper, we described a new approach for dealing with non-registered instruments based on category-level recognition. The conclusions of this chapter are summarized as follows:

- We pointed out a new problem in musical instrument recognition, that is, non-registered instruments. Although the problem of non-registered instruments is inevitable for annotating real-world musical audio signals, it has not been dealt with in previous studies.

- We proposed category-level recognition as a solution to the above-mentioned problem. This approach aims at implementing human-like flexible instrument recognition such as "I have not heard the sounds of this instrument but it may be a kind of strings." It therefore has a wide range of potentials, as discussed in Section 4.3.4, as well as musical instrument annotation.

- We proposed a method for automatically acquiring a musical instrument taxonomy optimized for computational category-level instrument recognition. Although category-level recognition of (registered) musical instruments based on a musical instrument taxonomy has been attempted in previous studies [20, 23, 50], the taxonomy used was manually designed.

Chapter 5

Feature Weighting based on Mixed-sound Template for Polyphonic Music

This chapter aims to discuss how to achieve robust instrument recognition with respect to polyphonic music. We first describe that extraction of harmonic structures suppresses the influence of interfering notes. We then point out that the suppression by harmonic structure extraction is insufficient and propose a method of feature weighting to minimize the influence. We also present a method of using musical context for further improvement.

5.1 Introduction

In this chapter, we deal with instrument recognition in polyphonic music. The main difficulty in recognizing instruments in polyphonic music is the fact that acoustical features of each instrument cannot be extracted without blurring because of the overlapping of partials (harmonic components). Here, we approach this *overlapping problem* by weighting each feature based on how much the feature is affected by the overlapping. If we can give higher weights to features suffering less from this problem and lower weights to features suffering more, it will facilitate robust instrument recognition in polyphonic music. To do this, we quantitatively evaluate the influence of the overlapping on each feature as the ratio of the *within-class variance* to the *between-class variance* in the distribution of training data obtained from polyphonic sounds because greatly suffering from the overlapping means having large variation when polyphonic sounds are analyzed. This evaluation makes the feature weighting described above equivalent to dimensionality reduction using *linear discriminant analysis* (LDA) on training data obtained from polyphonic sounds.

Because LDA generates feature axes using a weighted mixture where the weights minimize the ratio of the within-class variance to the between-class variance, using LDA on training data obtained from polyphonic sounds generates a subspace where the influence of the overlapping problem is minimized. We call this method *DAMS* (discriminant analysis with mixed sounds). In previous studies, techniques such as time-domain waveform template matching [37], feature adaptation with manual feature classification [38], and the missing feature theory [39, 40] have been tried to cope with the overlapping problem, but no attempts have been made to give features appropriate weights based on their robustness to the overlapping.

In addition, we propose a method for improving instrument recognition using musical context. This method is aimed at avoiding musically unnatural errors by considering the temporal continuity of melodies; for example, if the identified instrument names of a sequential note sequence are all "flute" except for one "clarinet," this exception can be considered an error and corrected.

5.2 Notewise Musical Instrument Recognition for Polyphonic Music

In instrument recognition in this chapter, the instrument for each note is identified. Suppose that a given audio signal contains K notes, $n_1, n_2, \cdots, n_k, \cdots, n_K$. The recognition process has two basic subprocesses: feature extraction and a posteriori probability calculation. In the former process, a feature vector consisting of some acoustic features is extracted from the given audio signal for each note. Let $\boldsymbol{x}_k$ be the feature vector extracted for note n_k. In the latter process, for each of the target instruments, $\omega_1, \cdots, \omega_m$, the probability $p(\omega_i|\boldsymbol{x}_k)$ that the feature vector $\boldsymbol{x}_k$ is extracted from a sound of the instrument ω_i is calculated. Based on the Bayes theorem, $p(\omega_i|\boldsymbol{x}_k)$ can be expanded as follows:

$$p(\omega_i|\boldsymbol{x}_k) = \frac{p(\boldsymbol{x}_k|\omega_i)p(\omega_i)}{\sum_{j=1}^{m} p(\boldsymbol{x}_k|\omega_j)p(\omega_j)},$$

where $p(\boldsymbol{x}_k|\omega_i)$ is a probability density function (PDF) and $p(\omega_i)$ is the a priori probability with respect to the instrument ω_i. The PDF $p(\boldsymbol{x}_k|\omega_i)$ is trained using data prepared in advance. Finally, the name of the instrument maximizing $p(\omega_i|\boldsymbol{x}_k)$ is determined for each note n_k.

5.3 Feature Weighting based on Mixed-sound Template

In this section, we discuss how to design an instrument recognition method that is robust to the overlapping of sounds. First, we explain that extracting harmonic structures effectively suppresses the influence of other simultaneously played notes. Next, we point out that harmonic structure extraction is insufficient and propose a method for feature weighting to improve the robustness.

5.3.1 Use of Harmonic Structure Model

In speech recognition and speaker recognition studies, features of spectral envelopes such as mel-frequency cepstrum coefficients are commonly used. Although they can reasonably represent the general shapes of observed spectra, when a signal of multiple instruments simultaneously playing is analyzed, focusing on the component corresponding to each instrument from the observed spectral envelope is difficult. Because most musical sounds except percussive ones have harmonic structures, previous studies on instrument recognition [38, 46, 109] have commonly extracted the harmonic structure of each note and then extracted acoustic features from the structures.

We also extract the harmonic structure of each note and then extract acoustic features from the structure. The harmonic structure model $\mathcal{H}(n_k)$ of the note n_k can be represented as the following equation:

$$\mathcal{H}(n_k) = \{(F_i(t), A_i(t)) \mid i = 1,2,\cdots,h, 0 \leq t \leq T\},$$

where $F_i(t)$ and $A_i(t)$ are the frequency and amplitude of the i-th partial at time t. Frequency is represented by relative frequency where the temporal median of the fundamental frequency, $F_1(t)$, is 1. Above, h is the number of harmonics, and T is the note duration. This modeling of musical instrument sounds based on harmonic structures can restrict the influence of the overlapping of sounds of multiple instruments to the overlapping of partials. Although actual musical instrument sounds contain non-harmonic components, which can be factors characterizing sounds, we focus only on harmonic ones because non-harmonic ones are difficult to reliably extract from a mixture of sounds.

5.3.2 Feature Weighting based on Robustness to Overlapping of Sounds

As described in the previous section, the influence of the overlapping of sounds of multiple instruments is restricted to the overlapping of the partials by extracting the harmonic structures. If two notes have no partials with common frequencies, the influence of one on the other when the two notes are simultaneously played may be ignorably small. In practice, however, partials often overlap. When two notes with the pitches of C4 (about 262 Hz) and G4 (about 394 Hz) are simultaneously played, for example, the $3i$-th partials of the C4 note and the $2i$-th partials of the G4 note overlap for every natural number i. Because note combinations that can generate harmonious sounds cause overlaps in many partials in general, coping with the overlapping of partials is a serious problem.

One effective approach for coping with this overlapping problem is feature weighting based on the robustness to the overlapping problem. If we can give higher weights to features suffering less from this problem and lower weights to features suffering more, it will facilitate robust instrument recognition in polyphonic music. Concepts similar to this feature weighting, in fact, have been proposed, such as the missing feature theory [39, 40] and feature adaptation [38].

- Eggink and Brown [39, 40] applied the missing feature theory to the problem of recognizing instruments in polyphonic music. This is a technique for cancelling unreliable features using a vector called a mask, which represents whether each feature is reliable or not. Because masking a feature is equivalent to giving a weight of zero to it, this technique can be considered an implementation of the feature weighting concept. Although this technique is known to be effective if the features to be masked are given, automatic mask estimation is very difficult in general and has not yet been established.

- Kinoshita *et al.* [38] proposed a feature adaptation method. They manually classified their features for recognition into three types (additive, preferential, and fragile) according to how the features varied when partials overlapped. Their method recalculates or cancels the features extracted from overlapping components according to the three types. Similarly to Eggink's work, cancelling features can be considered an implementation of the feature weighting concept. Because this method requires manually classifying features in advance, however, using a variety of fea-

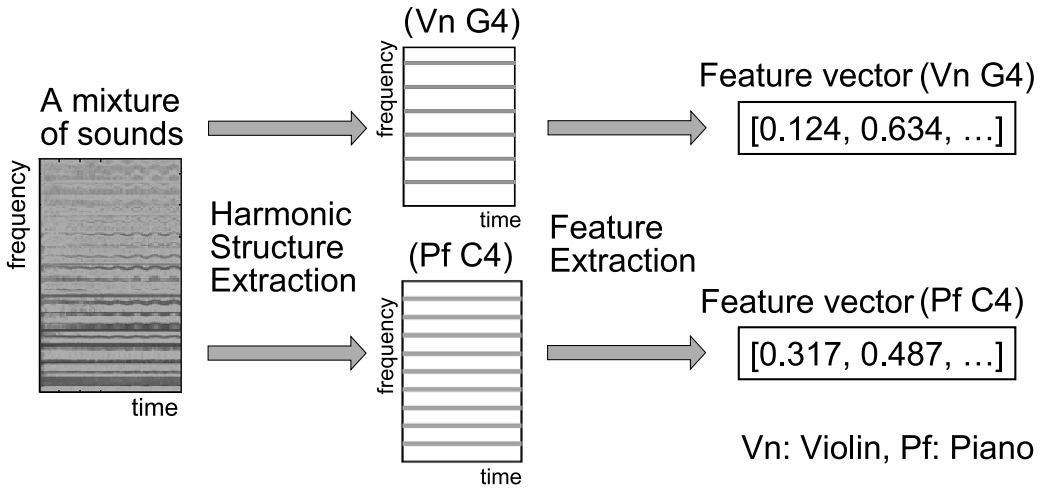


Figure 5.1: Overview of process of constructing mixed-sound template.

tures is difficult. They introduced a feature weighting technique, but this technique was performed on monophonic sounds and hence did not cope with the overlapping problem.

- Otherwise, there has been Kashino's work based on a time-domain waveform template-matching technique with adaptive template filtering [37]. The aim was the robust matching of an observed waveform and a mixture of waveform templates by adaptively filtering the templates. This study, therefore, did not deal with feature weighting based on the influence of the overlapping problem.

The issue in the feature weighting described above is how to quantitatively design the influence of the overlapping problem. Because training data were obtained only from monophonic sounds in previous studies, this influence could not be evaluated by analyzing the training data. Our DAMS method quantitatively models the influence of the overlapping problem on each feature as the ratio of the within-class variance to the between-class variance in the distribution of training data obtained from polyphonic sounds. This modeling makes weighting features to minimize the influence of the overlapping problem equivalent to applying LDA to training data obtained from polyphonic sounds.

Training data are obtained from polyphonic sounds through the process shown in Figure 5.1. The sound of each note in the training data is labeled in advance with the instrument name, the F0, the onset time, and the duration. By using these labels, we extract the harmonic structure corresponding to each note from the spectrogram. We then extract acoustic features from the harmonic structure. We thus obtain a set of many

feature vectors, called a *mixed-sound template*, from polyphonic sound mixtures.

The main issue in constructing a mixed-sound template is to design an appropriate subset of polyphonic sound mixtures. This is a serious issue because there are an infinite number of possible combinations of musical sounds due to the large pitch range of each instrument[1]. The musical feature that is the key to resolving this issue is a tendency of intervals of simultaneous notes. In Western tonal music, some intervals such as minor 2nds are more rarely used than other intervals such as major 3rds and perfect 5ths because minor 2nds generate dissonant sounds in general. By generating polyphonic sounds for template construction from the scores of actual (existing) musical pieces, we can obtain a data set that reflects the tendency mentioned above[2]. We believe that this approach improves instrument identification even if the pieces used for template construction are different from the piece to be identified for the following two reasons:

- There are different distributions of intervals found in simultaneous sounding notes in tonal music. For example, three simultaneous notes with the pitches of C4, C#4, and D4 are rarely used except for special effects.

- Because we extract the harmonic structure from each note, as previously mentioned, the influence of multiple instruments simultaneously playing is restricted to the overlapping of partials. The overlapping of partials can be explained by two main factors: which partials are affected by other sounds, related to *note combinations*, and how much each partial is affected, mainly related to *instrument combinations*. Note combinations can be reduced because our method considers only relative-pitch relationships, and the lack of instrument combinations is not critical to recognition as we find in an experiment described below. If the intervals of note combinations in a training data set reflect those in actual music, therefore, the training data set will be effective despite a lack of other combinations.

[1]Because our data set of musical instrument sounds consists of 2,651 notes of five instruments, $C(2651, 3) \approx 3.1$ billion different combinations are possible even if the number of simultaneous voices is restricted to three. About 98 years would be needed to train all the combinations, assuming that one second is needed for each combination.

[2]Although this discussion is based on tonal music, this may be applicable to atonal music by preparing the scores of pieces of atonal music.

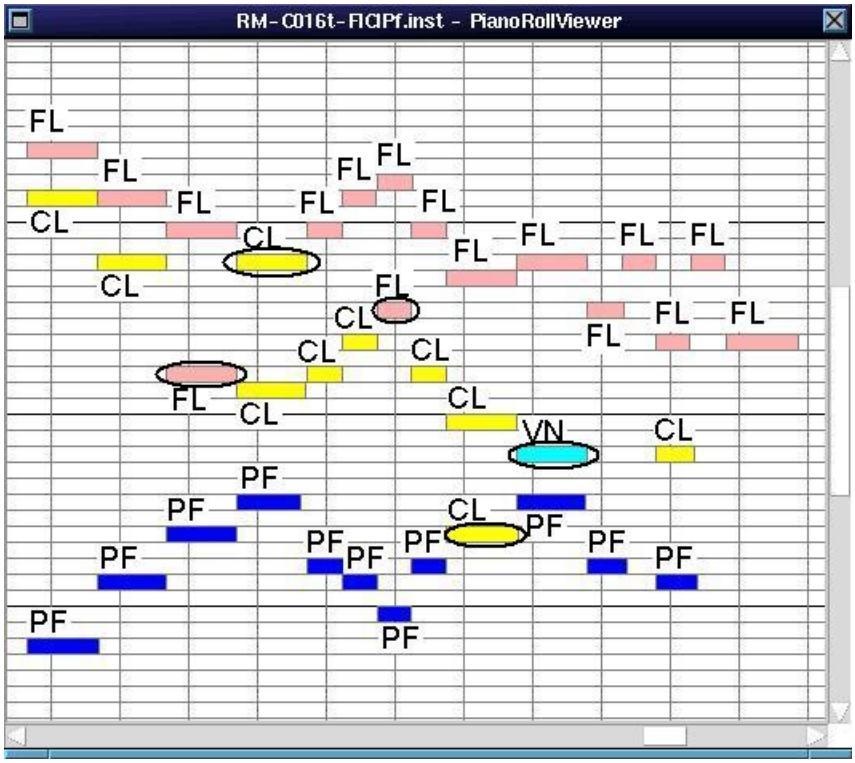


Figure 5.2: Example of musically unnatural errors. This example is excerpt from results of recognizing each note individually in piece of trio music. Marked notes are musically unnatural errors, which can be avoided by using musical context. PF, VN, CL and FL represent piano, violin, clarinet and flute.

5.4 Use of Musical Context

In this section, we propose a method for improving instrument recognition by considering musical context. The aim of this method is to avoid unusual events in tonal music, for example, only one clarinet note appearing in a sequence of notes (a melody) played on a flute, as shown in Figure 5.2. As mentioned in Section 5.2, the a posteriori probability $p(\omega_i|\boldsymbol{x}_k)$ is given by $p(\omega_i|\boldsymbol{x}_k) = p(\boldsymbol{x}_k|\omega_i)p(\omega_i)/\sum_j p(\boldsymbol{x}_k|\omega_j)p(\omega_j)$. The key idea behind using musical context is to apply the a posteriori probabilities of n_k's temporally neighboring notes to the a priori probability, $p(\omega_i)$, of the note n_k (Figure 5.3). This is based on the idea that, if almost all notes around the note n_k are recognized as the instrument ω_i, n_k is also probably played on ω_i. To achieve this, we have to resolve the following issue:

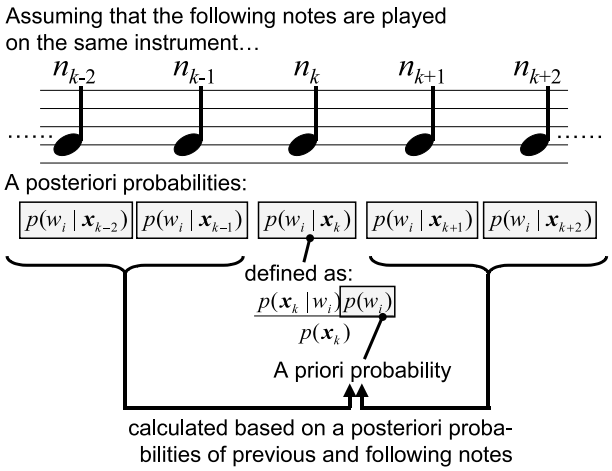


Figure 5.3: Key idea for using musical context. To calculate a posteriori probability of note n_k, a posteriori probabilities of temporally neighboring notes of n_k are used.

Issue *Distinguishing notes played on the same instrument as n_k from neighboring notes.*

Because various instruments are played at the same time, an recognition system has to distinguish notes that are played on the same instrument as the note n_k from notes played on other instruments. This is not easy because it is mutually dependent on musical instrument recognition.

We resolve this issue as follows:

Solution *Take advantage of the parallel movement of simultaneous parts.*

In Western tonal music, voices rarely cross. This may be explained due to the human's ability to recognize multiple voices easier if they do not cross each other in pitch [54]. When they listen, for example, to two simultaneous note sequences that cross, one of which is descending and the other of which is ascending, they cognize them as if the sequences approach each other but never cross. Huron also explains that the pitch-crossing rule (parts should not cross with respect to pitch) is a traditional voice-leading rule and can be derived from perceptual principles [110]. We therefore judge whether two notes, n_k and n_j, are in the same part (i.e., played on the same instrument) as follows: Let $s_{\mathrm{h}}(n_k)$ and $s_{\mathrm{l}}(n_k)$ be the maximum number of simultaneously played notes in the higher and lower pitch ranges when the note n_k is being played. Then, the two notes, n_k and n_j, are considered to be in the same part if and only if $s_{\mathrm{h}}(n_k) = s_{\mathrm{h}}(n_j)$ and $s_{\mathrm{l}}(n_k) = s_{\mathrm{l}}(n_j)$ (Figure 5.4). Kashino *et al.* [37] have introduced musical role consistency to generate music streams. They have designed

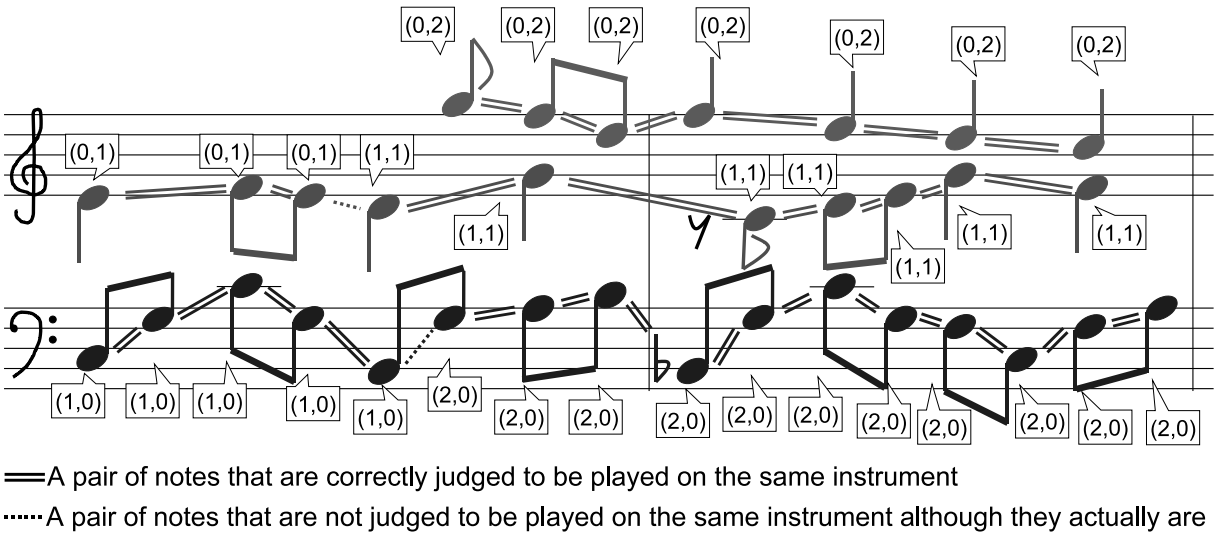


Figure 5.4: Example of judgment of whether notes are played on same instrument. Each tuple (a, b) represents $s_{\mathrm{h}}(n_k) = \mathsf{a}$ and $s_{\mathrm{l}}(n_k) = \mathsf{b}$.

two kinds of musical roles: the highest and lowest notes (usually corresponding to the principal melody and bass lines). Our method can be considered an extension of their musical role consistency.

[1st pass] Pre-calculation of a posteriori probabilities

For each note n_k, the a posteriori probability, $p(\omega_i|\boldsymbol{x}_k)$, is calculated by considering the a priori probability, $p(\omega_i)$, to be a constant because the a priori probability, which depends on the a posteriori probabilities of temporally neighboring notes, cannot be determined in this step.

[2nd pass] Re-calculation of a posteriori probabilities

This pass consists of three steps:

(1) *Finding notes played on same instrument*
Notes that satisfy $\{n_j \mid s_{\mathrm{h}}(n_k) = s_{\mathrm{h}}(n_j) \cap s_{\mathrm{l}}(n_k) = s_{\mathrm{l}}(n_j)\}$ are extracted from notes temporally neighboring n_k. This extraction is performed from the nearest note to farther notes and stops when c notes has been extracted (c is a positive integral constant). Let $\mathcal{N}$ be the set of the extracted notes.

(2) *Calculating a priori probability*
The a priori probability of the note n_k is calculated based on the a posteriori prob-

abilities of the notes extracted in the previous step. Let $p_1(\omega_i)$ and $p_2(\omega_i)$ be the a priori probabilities calculated from musical context and other cues, respectively. Then, we define the a priori probability, $p(\omega_i)$, to be calculated here as follows:

$$p(\omega_i) = \lambda p_1(\omega_i) + (1 - \lambda) p_2(\omega_i),$$

where λ is a confidence measure of musical context. Although this measure can be calculated through statistical analysis as the probability that the note n_k will be played on instrument ω_i when all the extracted neighboring notes of n_k are played on ω_i, we use $\lambda = 1 - (1/2)^c$ for simplicity, where c is the number of notes in $\mathcal{N}$. This is based on the heuristics that as more notes are used to represent a context, the context information is more reliable. We define $p_1(\omega_i)$ as follows:

$$p_1(\omega_i) = \frac{1}{\alpha} \prod_{n_j \in \mathcal{N}} p(\omega_i | \boldsymbol{x}_j),$$

where $\boldsymbol{x}_j$ is the feature vector for the note n_j and α is the normalizing factor given by $\alpha = \sum_{\omega_i} \prod_{n_j} p(\omega_i | \boldsymbol{x}_j)$. We use $p_2(\omega_i) = 1/m$ for simplicity.

(3) *Updating a posteriori probability*
The a posteriori probability is re-calculated using the a priori probability calculated in the previous step.

5.5 Details of Our Instrument Recognition Method

The details of our instrument recognition method are given below. An overview is shown in Figure 5.5. First, the spectrogram of a given audio signal is generated. Next, the harmonic structure of each note is extracted based on data on the F0, the onset time, and the duration of each note, which are estimated in advance using an existing method (e.g., [9, 46, 109]). Then, feature extraction, dimensionality reduction, a posteriori probability calculation, and instrument determination are performed in that order.

5.5.1 Short-time Fourier Transform

The spectrogram of the given audio signal is calculated using the short-time Fourier transform (STFT) shifted by 10 ms (441 points at 44.1 kHz sampling) with an 8192-point Hamming window.

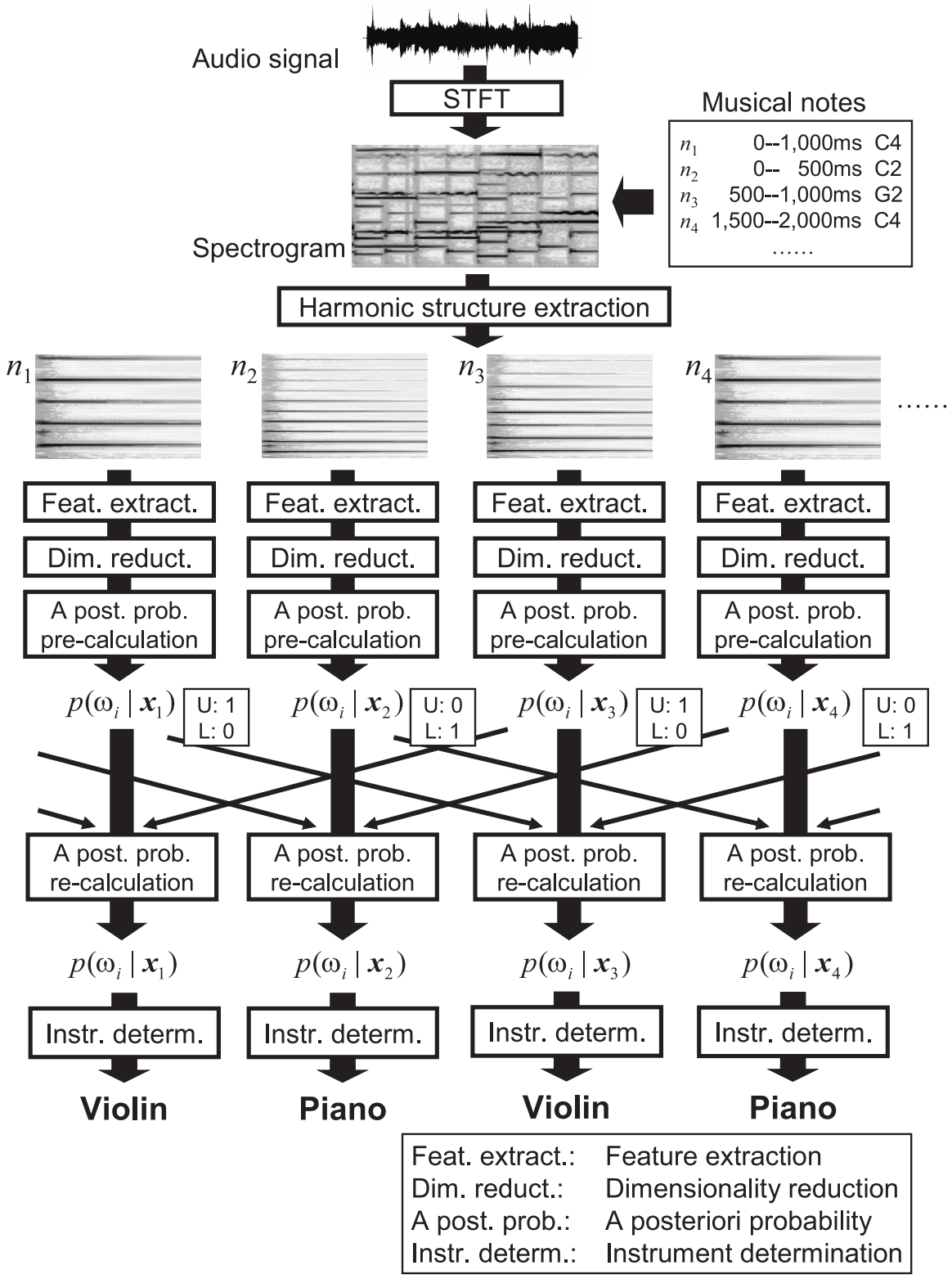


Figure 5.5: Flow of our instrument recognition method.

5.5.2 Harmonic Structure Extraction

The harmonic structure of each note is extracted according to note data estimated in advance. Spectral peaks corresponding to the first 10 harmonics are extracted from the onset time to the offset time. The offset time is calculated by adding the duration to the onset time. Then, the frequency of the spectral peaks is normalized so that the temporal mean of F0 is 1.

Next, the harmonic structure is trimmed because training and recognition require notes with fixed durations. Because a mixed-sound template with a long duration is more stable and robust than a template with a short one, trimming a note to keep it as long as possible is best. We therefore prepare three templates with different durations (300, 450, and 600 ms), and the longest usable, as determined by the actual duration of each note, is automatically selected and used for training and recognition[3]. For example, the 450-ms template is selected for a 500-ms note. In this paper, the 300-, 450-, and 600-ms templates are called *Template Types I, II,* and *III*. Notes shorter than 300 ms are not recognized.

5.5.3 Feature Extraction

Features that are useful for recognition are extracted from the harmonic structure of each note. We basically use the same feature set as that used in Chapters 3 and 4, but remove some features due to the difficulty of extraction from a mixture of sounds. We therefore use the 43 features (for Template Type III) summarized in Table 5.1, which we expected to be robust with respect to a mixture of sounds. We use 37 features for Template Type II and 31 for I because of the limitations of the note durations.

5.5.4 Dimensionality Reduction

Using the DAMS method, the subspace minimizing the influence of the overlapping problem is obtained. Because a feature space should not be correlated to robustly perform the LDA calculation, before using the DAMS method, we obtain a non-correlative space by using principal component analysis (PCA). The dimensions of the feature space obtained with PCA is determined so that the cumulative proportion value is 99% (20 dimensions

[3]The template is selected based on the fixed durations instead of the tempo because temporal variations of spectra, which influence the dependency of features on the duration, occur on the absolute time scale rather than in the tempo.

Table 5.1: Overview of 43 features. Please see Section 3.3.2 for the exact definitions.

Spectral features	
1	Spectral centroid (SC)
2–10	Relative cumulative powers (RCP) (up to the 9th partials)
11	Odd/even power ratio (OER)
12–20	Number of stably existing partials (NEP)
Temporal features	
21	Power decay speed (PDS)
22–30*	Average differential of power envelope (ADP)
31–39*	Relative powers (RP)
Modulation features	
40, 41	Amplitude and frequency of AM
42, 43	Amplitude and frequency of FM

Note: The feature numbers are unrelated to the feature numbers in Chapter 3.
*While the original definition in Section 3.3.2 includes t with the values from 0.15 to 0.95, The values of t are here up to 0.25, 0.40, and 0.55 for Template Types I, II, and III, respectively, due to the limitations of the note durations.

in most cases). By using the DAMS method in this subspace, we obtain an $(m-1)$-dimensional space (m: the number of instruments in the training data).

5.5.5 A Posteriori Probability Calculation

For each note n_k, the a posteriori probability, $p(\omega_i|\boldsymbol{x}_k)$, is calculated. As previously described, this probability can be calculated using the following equation:

$$p(\omega_i|\boldsymbol{x}_k) = \frac{p(\boldsymbol{x}_k|\omega_i)p(\omega_i)}{\sum_j p(\boldsymbol{x}_k|\omega_j)p(\omega_j)}.$$

The PDF $p(\boldsymbol{x}_k|\omega_i)$ is calculated from training data prepared in advance by using an F0-dependent multivariate normal distribution, which is proposed in Chapter 3. For each element of the feature vector, the pitch dependency of The a priori probability, $p(\omega_i)$, is calculated on the basis of the musical context, that is, the a posteriori probabilities of neighboring notes, as described in Section 3.

5.5.6 Instrument Determination

Finally, the instrument maximizing the a posteriori probability $p(\omega_i|\boldsymbol{x}_k)$ is determined as the recognition result for the note n_k.

5.6 Experiments

5.6.1 Data for Experiments

We used audio signals generated by mixing audio data taken from a solo musical instrument sound database according to standard MIDI files (SMFs) so that we would have correct data on F0s, onset times, and durations of all notes because the focus of our experiments was solely on evaluating the performance of our instrument recognition method by itself.

The SMFs we used in the experiments were three pieces taken from RWC-MDB-C-2001 (Piece Nos. 13, 16, and 17) [111]. These are classical musical pieces consisting of four or five simultaneous voices. We created SMFs of duo, trio, and quartet music by choosing two, three, and four simultaneous voices from each piece. We also prepared solo-melody SMFs for template construction.

As audio sources for generating audio signals of duo, trio, and quartet music, an excerpt of RWC-MDB-I-2001 [103], listed in Table 5.2, was used. To avoid using the same audio data for training and testing, we used 011PFNOM, 151VNNOM, 311CLNOM, and 331FLNOM for the test data and the others in Table 5.2 for the training data. We prepared audio signals of all possible instrument combinations within the restrictions in Table 5.3, which was defined by taking the pitch ranges of instruments into account. For example, 48 different combinations were made for quartet music.

5.6.2 Experiment 1: Leave-one-out

The experiment was conducted using the leave-one-out cross-validation method. When evaluating a musical piece, a mixed-sound template was constructed using the remaining two pieces. Because we evaluated three pieces, we constructed three different mixed-sound templates by dropping the piece used for testing. The mixed-sound templates were constructed from audio signals of solo and duo music (S+D) and solo, duo, and trio music (S+D+T). For comparison, we also constructed a template, called a solo-sound template, only from solo musical sounds. The number of notes in each template is listed in Table

Table 5.2: Audio data on solo instruments

Instr. no.	Name	Pitch range	Variation	Dynamics	Articulation	# of data
01	Piano (PF)	A0–C8	1, 2, 3	Forte,		792
09	Classical Guitar (CG)	E2–E5	〃	mezzo	Normal	702
15	Violin (VN)	G3–E7	〃	&	only	576
31	Clarinet (CL)	D3–F6	〃	piano		360
33	Flute (FL)	C4–C7	1, 2			221

Table 5.3: Instrument candidates for each part.

Part 1	PF, VN, FL
Part 2	PF, CG, VN, CL
Part 3	PF, CG
Part 4	PF, CG

5.4. To evaluate the effectiveness of F0-dependent multivariate normal distributions and using musical context, we tested both cases with and without each technique. We fed the correct data on the F0s, onset times, and durations of all notes because our focus was on the performance of the instrument recognition method alone.

The results are shown in Table 5.5. Each number in the table is the average of the recognition rates for the three pieces. Using the DAMS method, the F0-dependent multivariate normal distribution, and the musical context, we improved the recognition rates from 50.9 to 84.1% for duo, from 46.1 to 77.6% for trio, and from 43.1 to 72.3% for quartet music on average.

We confirmed the effect of each of the DAMS method (mixed-sound template), the F0-dependent multivariate normal distribution, and the musical context using McNemar's test. McNemar's test is usable for testing whether the proportions of A-labeled ("correct" in this case) data to B-labeled ("incorrect") data under two different conditions are significantly different. Because the numbers of notes are different among instruments, we sampled 100 notes at random for each instrument to avoid the bias. The results of McNemar's test for the quartet music are listed in Table 5.6 (those for the trio and duo music are omitted but are basically same as those for the quartet), where the χ_0^2 are test

Table 5.4: Number of notes in mixed-sound templates (Type I). Templates of Types II and III have about 1/2 and 1/3–1/4 times the notes of Type I (details are omitted due to a lack of space). S+D and S+D+T stand for the templates constructed from audio signals of solo and duo music, and from those of solo, duo, and trio music, respectively.

		S+D	S+D+T	Subset*
No. 13	PF	31,334	83,491	24,784
	CG	23,446	56,184	10,718
	VN	14,760	47,087	9,804
	CL	7,332	20,031	4,888
	FL	4,581	16,732	3,043
No. 16	PF	26,738	71,203	21,104
	CG	19,760	46,924	8,893
	VN	12,342	39,461	8,230
	CL	5,916	16,043	3,944
	FL	3,970	14,287	2,632
No. 17	PF	23,836	63,932	18,880
	CG	17,618	42,552	8,053
	VN	11,706	36,984	7,806
	CL	5,928	16,208	3,952
	FL	3,613	13,059	2,407

*Template used in Experiment III.

statistics. Because the criterion region at $\alpha = 0.001$ (which is the level of significance) is $(10.83, +\infty)$, the differences except S+D vs. S+D+T are significant at $\alpha = 0.001$.

Other observations are summarized as follows:

- The results of the S+D and S+D+T templates were not significantly different even if the test data were from quartet music. This means that constructing a template from polyphonic sounds is effective even if the sounds used for the template construction do not have the same complexity as the piece to be recognized.

- For PF and CG, the F0-dependent multivariate normal distribution was particularly effective. This is because these instruments have large pitch dependencies due to their wide pitch ranges.

Table 5.5: Results of Experiment 1.

Template		Solo-sound				S+D				S+D+T			
F0-dpt.		×	×	◯	◯	×	×	◯	◯	×	×	◯	◯
Context		×	◯	×	◯	×	◯	×	◯	×	◯	×	◯
Duo	PF	53.7%	63.0%	70.7%	**84.7%**	61.5%	63.8%	69.8%	**78.9%**	69.1%	70.8%	71.0%	**82.7%**
	CG	46.0%	44.6%	50.8%	42.8%	50.9%	67.5%	70.2%	**85.1%**	44.0%	57.7%	71.0%	**82.9%**
	VN	63.7%	**81.3%**	63.1%	**75.6%**	68.1%	**85.5%**	70.6%	**87.7%**	65.4%	**84.2%**	67.7%	**88.1%**
	CL	62.9%	70.3%	53.4%	56.1%	**81.8%**	**92.1%**	**81.9%**	**89.9%**	**84.6%**	**95.1%**	**82.9%**	**92.6%**
	FL	28.1%	33.5%	29.1%	38.7%	67.6%	**84.9%**	67.6%	**78.8%**	56.8%	70.5%	61.5%	74.3%
	Av.	50.9%	58.5%	53.4%	59.6%	66.0%	**78.8%**	72.0%	**84.1%**	64.0%	**75.7%**	70.8%	**84.1%**
Trio	PF	42.8%	49.3%	63.0%	**75.4%**	44.1%	43.8%	57.0%	61.4%	52.4%	53.6%	61.5%	68.3%
	CG	39.8%	39.1%	40.0%	31.7%	52.1%	66.8%	68.3%	**82.0%**	47.2%	62.8%	68.3%	**82.8%**
	VN	61.4%	**76.8%**	62.2%	72.5%	67.0%	**81.8%**	70.8%	**83.5%**	60.5%	**80.6%**	68.1%	**82.5%**
	CL	53.4%	55.7%	46.0%	43.9%	69.5%	**77.1%**	72.2%	**78.3%**	71.0%	**82.8%**	**76.2%**	**82.8%**
	FL	33.0%	42.6%	36.7%	46.5%	68.4%	**77.9%**	68.1%	**76.9%**	59.1%	69.3%	64.0%	71.5%
	Av.	46.1%	52.7%	49.6%	54.0%	60.2%	69.5%	67.3%	**76.4%**	58.0%	69.8%	67.6%	**77.6%**
Quar-tet	PF	38.9%	46.0%	54.2%	64.9%	38.7%	38.6%	50.3%	53.1%	46.1%	46.6%	53.3%	57.2%
	CG	34.3%	33.2%	35.3%	29.1%	51.2%	62.7%	64.8%	**75.3%**	51.2%	64.5%	65.0%	**79.1%**
	VN	60.2%	74.3%	62.8%	73.1%	70.0%	**81.2%**	72.7%	**82.3%**	67.4%	**79.2%**	69.7%	**79.9%**
	CL	45.8%	44.8%	39.5%	35.8%	62.6%	66.8%	65.4%	69.3%	68.6%	74.4%	70.9%	74.5%
	FL	36.0%	50.8%	40.8%	52.0%	69.8%	**76.1%**	69.9%	**76.2%**	61.7%	69.4%	64.5%	70.9%
	Av.	43.1%	49.8%	46.5%	51.0%	58.5%	65.1%	64.6%	71.2%	59.0%	66.8%	64.7%	72.3%

◯: used, ×: not used. Bold font denotes recognition rates of higher than 75%.

- Using musical context improved recognition rates, on average, by approximately 10%. This is because, in the musical pieces used in our experiments, pitches in the melodies of simultaneous voices rarely crossed.

- When the solo-sound template was used, the use of musical context lowered recognition rates, especially for CL. Because our method of using musical context calculates the a priori probability of each note on the basis of the a posteriori probabilities of temporally neighboring notes, it requires an accuracy sufficient for pre-calculating the a posteriori probabilities of the temporally neighboring notes. The lowered recognition rates are because of the insufficient accuracy of this pre-calculation. In fact, this phenomenon did not occur when the mixed-sound templates, which improved the accuracies of the pre-calculations, were used. Therefore, musical context should be used together with some technique of improving the pre-calculation accuracies, such as a mixed-sound template.

Table 5.6: Results of McNemar's test for quartet music (Corr.=correct, Inc.=incorrect).

(a) Template comparison (with both F0-dpt. and context)

		Solo-sound	
		Corr.	Inc.
S+D	Corr.	233	133
	Inc.	25	109

$$\chi_0^2 = (133-25)^2/(133+25) = 73.82$$

		Solo-sound	
		Corr.	Inc.
S+D+T	Corr.	224	148
	Inc.	34	94

$$\chi_0^2 = (148-34)^2/(148+34) = 71.40$$

		S+D	
		Corr.	Inc.
S+D+T	Corr.	347	25
	Inc.	19	109

$$\chi_0^2 = (25-19)^2/(25+19) = 1.5$$

(b) With vs. without F0-dpt
(with S+D+T template and context)

		w/o F0-dpt	
		Corr.	Inc.
w/ F0-dpt	Corr.	314	58
	Inc.	25	103

$$\chi_0^2 = (58-25)^2/(58+25) = 13.12$$

(c) With vs. without context
(with S+D+T template and F0-dpt model)

		w/o Context	
		Corr.	Inc.
w/ Context	Corr.	308	64
	Inc.	27	101

$$\chi_0^2 = (64-27)^2/(64+27) = 15.04$$

- The recognition rate for PF was not high enough in some cases. This is because the timbre of PF is similar to that of CG. In fact, even humans had difficulty distinguishing them in listening tests of sounds resynthesized from harmonic structures extracted from PF and CG tones.

5.6.3 Experiment 2: Template Construction from Only One Piece

Next, to compare template construction from only one piece with that from two pieces (i.e., leave-one-out), we conducted an experiment on template construction from only one piece. The results are shown in Table 5.7. Even when using a template made from only one piece, we obtained comparatively high recognition rates for CG, VN, and CL. For FL, the results of constructing a template from only one piece were not high (e.g., 30–40%), but those from two pieces were close to the results of the case where the same piece was used for both template construction and testing. This means that a variety of influences of sounds overlapping were trained from only two pieces.

5.6.4 Experiment 3: Insufficient Instrument Combinations

We investigated the relationship between the coverage of instrument combinations in a template and the recognition rate. When a template that does not cover instrument combinations is used, the recognition rate might decrease. If this decrease is large, the number of target instruments of the template will be difficult to increase because $O(m^n)$ data are needed for a full-combination template, where m and n are the number of target instruments and simultaneous voices. The purpose of this experiment is to check whether such a decrease occurs in the use of a reduced-combination template. As the reduced-combination template, we used one that contains the combinations listed in Table 5.8 only. These combinations were chosen so that the order of the combinations was $O(m)$. Similarly to Experiment 1, we used the leave-one-out cross validation method. As we can see from Table 5.9, we did not find significant differences between using the full instrument combinations and the reduced combinations. This was confirmed, as shown in Table 5.10, through McNemar's test, similarly to Experiment 1. Therefore, we expect that the number of target instruments can be increased without the problem of combinational explosion.

Table 5.7: Template construction from only one piece (Experiment 2). Quartet only due to lack of space. [Unit: %]

	S+D				S+D+T			
	13	16	17	⋆	13	16	17	⋆
PF	(57.8)	32.3	38.4	36.6	(67.2)	33.2	45.1	39.7
CG	(73.3)	**78.1**	**76.2**	**76.7**	(**76.8**)	**84.3**	**80.3**	**82.1**
13 VN	(**89.5**)	59.4	**87.5**	**86.2**	(**87.2**)	58.0	**85.2**	**83.1**
CL	(68.5)	70.8	62.2	73.8	(72.3)	72.3	68.6	**75.9**
FL	(**85.5**)	40.2	74.9	**82.7**	(**86.0**)	38.9	68.8	**80.8**
PF	74.1	(64.8)	61.1	71.2	**79.6**	(67.1)	73.0	**78.3**
CG	**79.2**	(**77.9**)	**78.9**	74.3	70.4	(**82.6**)	74.0	**75.2**
16 VN	**89.2**	(**85.5**)	**87.0**	**87.0**	**86.0**	(**83.5**)	**84.7**	**85.0**
CL	68.1	(**78.9**)	68.9	**76.1**	72.4	(**82.8**)	**76.3**	**82.1**
FL	**82.0**	(**75.9**)	72.5	**77.3**	**77.9**	(72.3)	35.7	69.2
PF	53.0	39.4	(51.2)	51.6	52.2	40.6	(55.7)	53.7
CG	73.7	69.0	(**75.8**)	**75.0**	**76.0**	74.3	(**78.4**)	**80.0**
17 VN	**79.5**	61.2	(**78.3**)	73.6	**77.4**	58.0	(**78.7**)	71.7
CL	51.3	60.5	(57.1)	57.9	61.1	62.6	(66.9)	65.4
FL	65.0	35.0	(73.1)	68.7	58.6	34.7	(70.9)	62.6

⋆Leave-one-out. Numbers in left column denote piece numbers for test. Those in top row denote piece numbers for template construction.

5.6.5 Experiment 4: Effectiveness of LDA

Finally, we compared the dimensionality reduction using both PCA and LDA with that using only PCA to evaluate the effectiveness of LDA. The experimental method was leave-one-out cross validation. The results are shown in Figure 5.6. The difference between the recognition rates of the solo-sound template and the S+D or S+D+T template was 20–24% using PCA+LDA and 6–14% using PCA only. These results mean that LDA (or DAMS) successfully obtained a subspace where the influence of the overlapping of sounds of multiple instruments was minimal by minimizing the ratio of the within-class variance to the between-class variance. Under all conditions, using LDA was superior to not using LDA.

Table 5.8: Instrument combinations in Experiment 3.

Solo	PF, CG, VN, CL, FL
Duo	PF–PF, CG–CG, VN–PF, CL–PF, FL–PF
Trio	Not used
Quartet	Not used

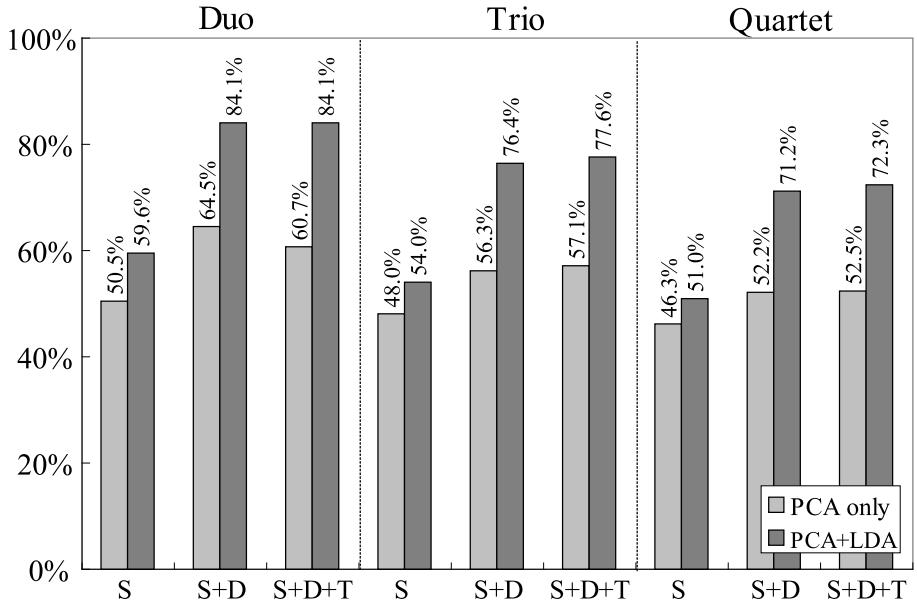


Figure 5.6: Comparison of using both PCA and LDA with using only PCA (Experiment 4). "Duo", "Trio", and "Quartet" represent pieces for test (recognition). "S", "S+D", and "S+D+T" represent types of templates.

We confirmed that combining LDA and the mixed-sound template is effective using two-way factorial analysis of variance (ANOVA) where the two factors are dimensionality reduction methods (PCA only and PCA+LDA) and templates (S, S+D, and S+D+T). Because we tested each condition using duo, trio, and quartet versions of Piece Nos. 13, 16, and 17, there are nine results for each cell of the two-factor matrix. The table of ANOVA is given in Table 5.11. From the table, we can see that the interaction effect as well as the effects of dimensionality reduction methods and templates are significant at $\alpha = 0.001$. This result means that mixed-sound templates are particularly effective when combined with LDA.

Table 5.9: Comparison of templates whose instrument combinations were reduced (subset) and not reduced (full set).

		Subset	Full set
Duo	PF	**85.4%**	**78.9%**
	CG	70.8%	**85.1%**
	VN	**88.2%**	**87.7%**
	CL	**90.4%**	**89.9%**
	FL	**79.7%**	**78.8%**
	Average	**82.9%**	**84.1%**
Trio	PF	73.9%	61.4%
	CG	62.0%	**82.0%**
	VN	85.7%	**83.5%**
	CL	**79.7%**	**78.3%**
	FL	**76.5%**	**76.9%**
	Average	**75.6%**	**76.4%**
Quartet	PF	68.9%	53.1%
	CG	52.4%	**75.3%**
	VN	**85.0%**	**82.3%**
	CL	71.1%	69.3%
	FL	74.5%	**76.2%**
	Average	70.4%	71.2%

5.6.6 Application to XML Annotation

In this section, we show an example of XML annotation of musical audio signals using our instrument recognition method. We used a simplified version of MusicXML instead of the original MusicXML format because our method does not include rhythm recognition and hence cannot determine note values or measures. The document type definition (DTD) of our simplified MusicXML is shown in Figure 5.7. The main differences between it and the original one are that elements related to notation, which cannot be estimated from audio signals, are reduced and that time is represented in seconds. The result of XML annotation of a piece of polyphonic music is shown in Figure 5.8. By using our instrument recognition method, we classified notes according to part and described the instrument

Table 5.10: Results of McNemar's test for full-set and subset templates.

		Subset	
		Corr.	Inc.
Full set	Corr.	341	25
	Inc.	19	115

$$\begin{aligned}\chi_0^2 &= (25-19)^2/(25+19)\\ &= 1.5\end{aligned}$$

Table 5.11: ANOVA.

Src. of var.	S.S.	d.f.	F-value	p-value
D.R.	0.336	1	102.08	1.806×10^{-13}
Template	0.302	2	45.75	7.57×10^{-12}
Interaction	0.057	2	8.75	5.73×10^{-4}
Residual	0.158	48	—	—
Total	0.855	53	—	—

S.S.=sum of squares, d.f.=degrees of freedom,
D.R.=dimensionality reduction.

for each part.

5.6.7 Discussion

We achieved average recognition rates of 84.1% for duo, 77.6% for trio, and 72.3% for quartet music chosen from five different instruments. We think that this performance is state-of-the-art, but we cannot directly compare these rates with experimental results published by other researchers because different researchers used different test data in general. We also find the following two limitations in our evaluation:

(a) The correct F0s are given.

(b) Non-realistic music (i.e., music synthesized by mixing isolated monophonic sound samples) is used.

First, we gave the correct F0 for every note in the evaluation because F0 estimation for a mixture of sounds is still a challenging problem. We have to integrate our method with a multiple-F0 estimation method to evaluate the whole performance, but the integration is

```
<!ENTITY % score-header
    "(work?, movement-number?, movement-title?,
      identification?, defaults?, credit*,
      part-list)">

<!ELEMENT part-list (score-part+)>
<!ELEMENT score-part
    (identification?, part-name,
    part-abbreviation?, score-instrument)>
<!ATTLIST score-part
    id ID #REQUIRED
>
<!ELEMENT score-instrument
    (instrument-name, instrument-abbreviation?)>
<!ELEMENT instrument-name (#PCDATA)>
<!ELEMENT instrument-abbreviation (#PCDATA)>

<!ELEMENT score-partwise-simple>
    (%score-header;, part+)>
<!ATTLIST score-partwise-simple
    version CDATA "1.0"
>
<!ELEMENT part (note+)>
<!ATTLIST part
    id IDREF #REQUIRED
>

<!ELEMENT note (pitch, onset, offset)>
<!ELEMENT pitch (step, alter?, octave)>
<!ELEMENT step (#PCDATA)>
<!ELEMENT alter (#PCDATA)>
<!ELEMENT octave (#PCDATA)>
<!ELEMENT onset (#PCDATA)>
<!ATTLIST onset
    unit CDATA "sec"
>
<!ELEMENT offset (#PCDATA)>
<!ATTLIST offset
    unit CDATA "sec"
>
```

Figure 5.7: DTD of our simplified MusicXML.

not easy because errors of F0 estimation seriously influence the performance of instrument recognition if cascade combination. In the next chapter, therefore, we will propose a new framework where both are probabilistically integrated.

Second, we used non-realistic music because information on the instruments for every note that was used as correct references in the evaluation was easy to prepare. Strictly speaking, however, the acoustical characteristics of real music are different from those of such synthesized music. The performance of our method would decrease for real music because successive notes in a melody sometimes overlap due to legato, unclear note onsets,

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE score-partwise-simple SYSTEM "partwisesimple.dtd">
<score-partwise-simple>
  <part-list>
    <score-part id="P1">
     <part-name>Part 1</part-name>
     <score-instrument>Piano</score-instrument>
    </score-part>
    <score-part id="P2">
     <part-name>Part 3</part-name>
     <score-instrument>Violin</score-instrument>
    </score-part>
                ......
  </part-list>
  <part id="P1">
    <note>
      <pitch>
        <step>G</step>
        <alter>+1</alter>
        <octave>3</octave>
      </pitch>
      <onset>1.0</onset>
      <offset>2.0</offset>
    </note>
    <note>
      <pitch>
        <step>G</step>
        <octave>3</octave>
      </pitch>
      <onset>2.0</onset>
      <offset>2.5</offset>
    </note>
    <note>
      <pitch>
        <step>D</step>
        <octave>4</octave>
      </pitch>
      <onset>2.5</onset>
      <offset>3.0</offset>
    </note>
                ......
  </part>
  <part id="P2">
    <note>
      <pitch>
        <step>D</step>
        <alter>+1</alter>
        <octave>4</octave>
      </pitch>
      <onset>1.5</onset>
      <offset>2.488541</offset>
    </note>
    <note>
      <pitch>
        <step>C</step>
        <alter>+1</alter>
        <octave>4</octave>
      </pitch>
      <onset>3.0</onset>
      <offset>3.5</offset>
    </note>
                ......
  </part>
                ......
</score-partwise-simple>
```

Figure 5.8: Example of MusicXML annotation.

and sound mixtures often involving reverberations. Because the framework proposed in the next chapter does not require such notewise references for evaluation, the difference of the difficulties between synthesized music and real performances will be discussed in the next chapter.

5.7 Conclusion

We proposed new methods for improving instrument recognition in polyphonic music. The conclusions of this chapter are summarized as follows:

- We presented a new solution to the problem of the overlapping of common-frequency partials. The key idea behind our solution is to collect training data from polyphonic sounds. Analyzing training data obtained from polyphonic sounds has made it possible to measure the influence of the partial overlapping on the variation in each feature and to weight the features to minimize the influence. The results of experiments with various conditions showed that this solution was effective even if the training data did not cover the thorough polyphonic combinations.

- We also presented a method for using musical context to avoid musically unnatural errors. If the instrument for a certain note is identified as a certain instrument, those for temporally neighboring notes are also probably the same instrument. Based on this idea, we introduced re-calculation of the a priori probability using the pre-calculated a posteriori probabilities for temporally neighboring notes. We confirmed that this method was effective as long as the musical piece to be recognized rarely has simultaneous melodies that cross each other in pitch.

Chapter 6

Note-estimation-free Instrument Recognition for Polyphonic Music

In this chapter, we first propose a new probabilistic representation of instrument existence called an *instrogram*. Next, we formulate the instrogram as a set of *instrument existence probabilities*, which is defined as the product of two kinds of probabilities. We then present a method for calculating the two kinds of probabilities.

6.1 Introduction

Although the number of studies dealing with instrument recognition in polyphonic music has been increasing in recent years (see Section 2.1.1 for the details), most of them require preceding note (or F0) estimation process. For example, OPTIMA [46], Ipanema [37], Kinoshita *et al.*'s method [38], and our previous method (see Chapter 5) identify the instrument for each note (called *notewise processing*) and hence have to estimate the onset time and fundamental frequency (F0) of each note. Eggink and Brown's methods [39, 40, 112] identify instruments for each frame. Although they do not require onset detection, they still require the estimation of F0s of notes played at each frame. Because onset detection and F0 estimation are difficult in polyphonic music in general, the performance of instrument recognition in these studies are greatly suffered from their errors. In the experiments of most studies mentioned above, therefore, correct data on onset times and F0s were manually given.

In this chapter, 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 [7] 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.

6.2 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 corresponds 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 Figure 6.1. 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 Figure 6.1 is given in Figure 6.2. 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.

6.3 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*

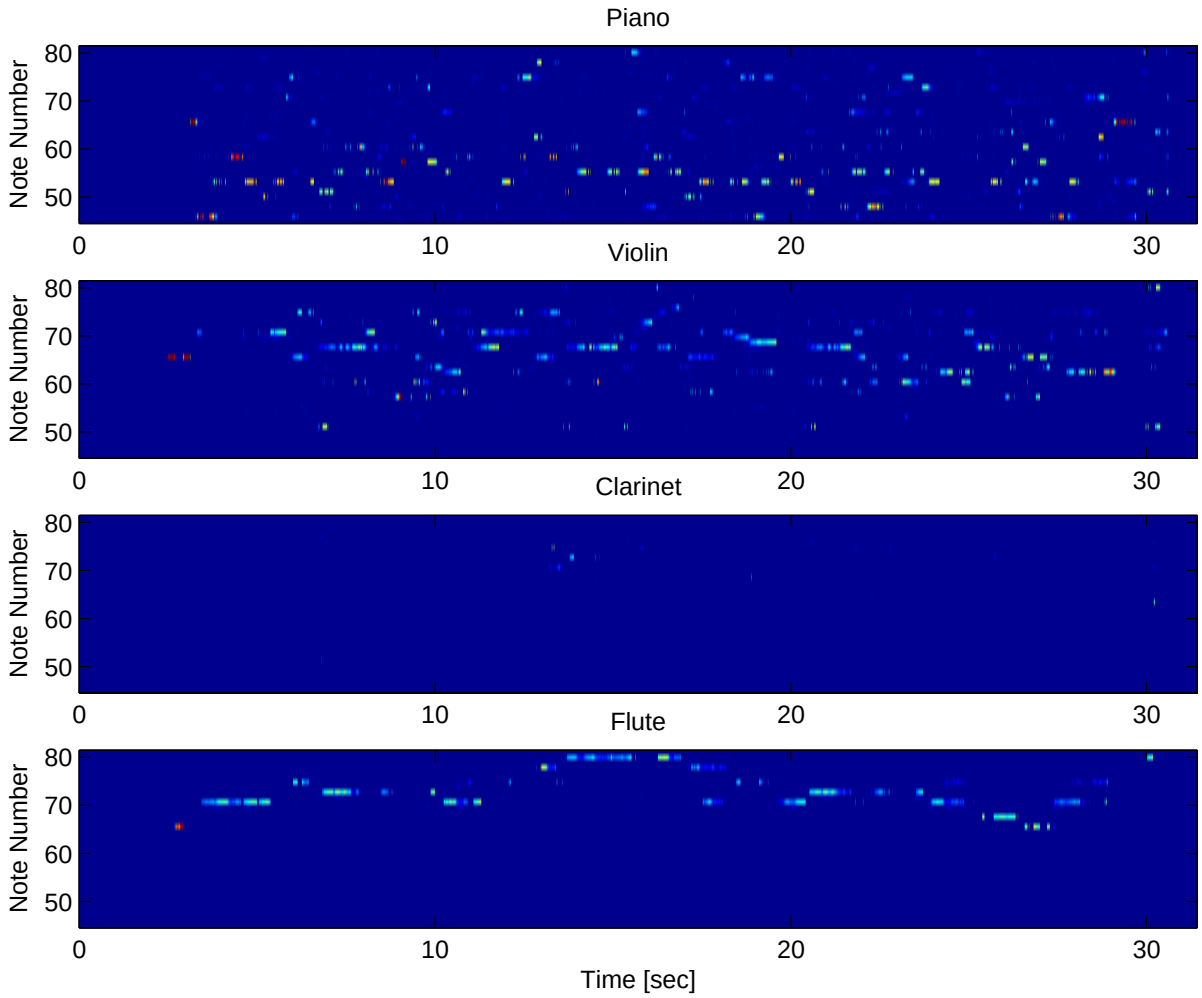


Figure 6.1: Example of the instrogram.

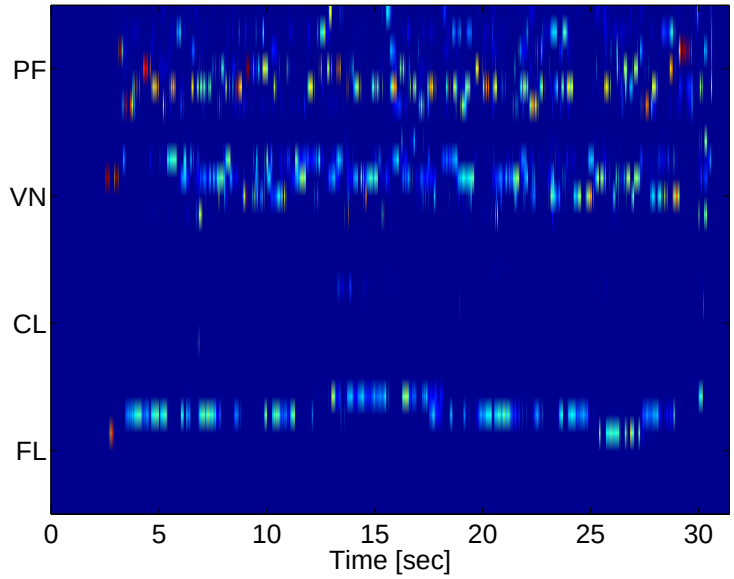


Figure 6.2: Simplified (summarized) instrogram for Figure 6.1.

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 $p(\omega_0; t, f) = 1 - \sum_{\omega_i \in \Omega} p(\omega_i; t, f)$.

6.3.1 Overview

Figure 6.3 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 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 by analyzing the power spectrum at each frame (*timewise processing*) using the PreFEst[7]. 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. The CIEPs are, on the other hand, calculated by analyzing the temporal trajectory of the harmonic structure with every F0 (*pitchwise processing*). The trajectory is analyzed with a framework similar to speech recognition, based on left-to-right hidden Markov models (HMMs) [113]. This HMM-based temporal modeling of harmonic structures is important because temporal variations in spectra characterize timbres well. This is the main difference from framewise recognition methodologies [40, 112]. Finally, the NIEPs and CIEPs are multiplied.

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 prece-

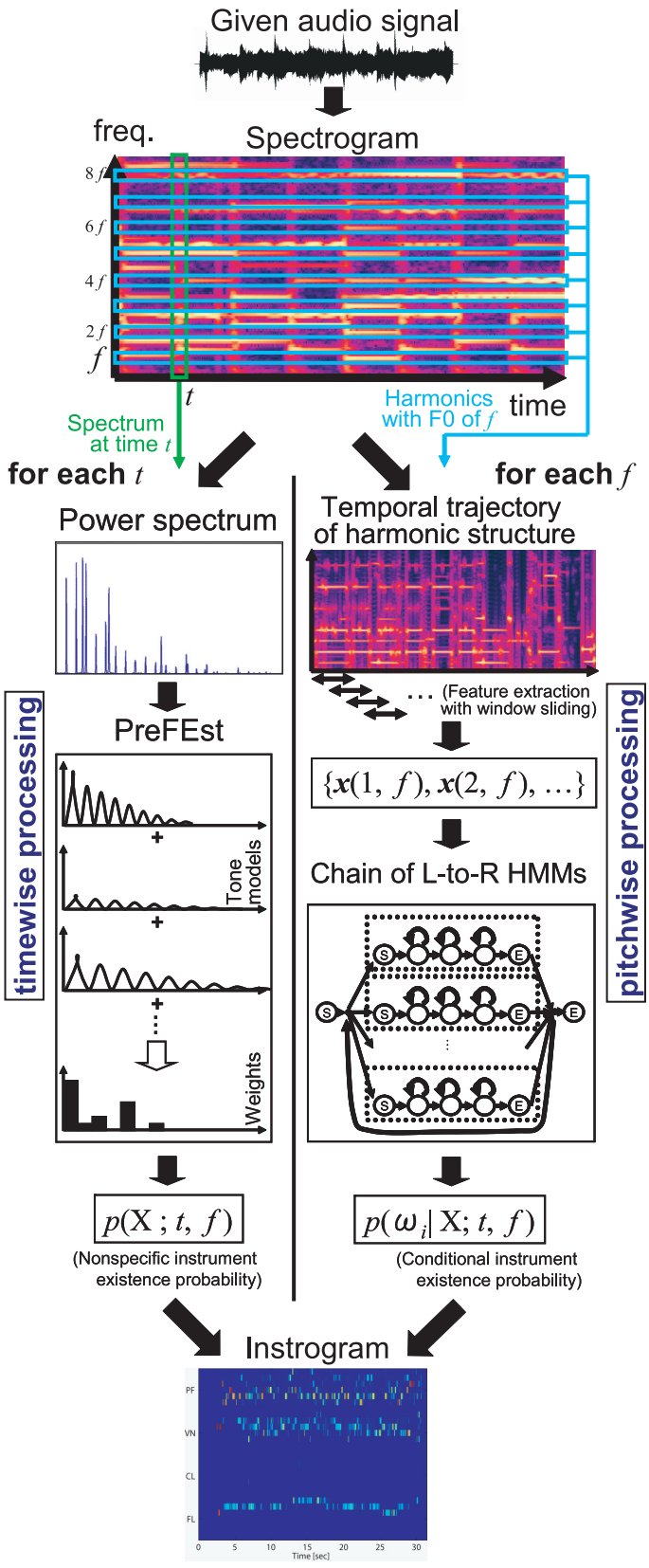


Figure 6.3: Overview of our technique for calculating the instrogram.

dent note estimation, which is generally difficult and not robust for polyphonic music[1]. Unlike such a notewise symbolic approach, our non-symbolic and non-sequential approach is more robust for polyphonic music.

6.3.2 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)

[1]We tested robustness with respect to onset errors in identifying an instrument for every note using our previous method[114]. 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%

algorithm [7]. 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, N = 16, W = 17 cent, and $c(h) = G(h; 1, 5.5)$. This tone model was also used in the earliest version of the PreFEst [115].

6.3.3 Conditional Instrument Existence Probability

The following steps are performed for every frequency f.

[Step 1] Harmonic structure extraction

The temporal trajectory of the harmonic structure with F0 of f is extracted. This is represented as

$$\mathcal{H}(t, f) = \{(F_i(t, f), A_i(t, f)) \mid i = 1, \cdots, h\},$$

where $F_i(t, f)$ and $A_i(t, f)$ are the frequency of amplitude of i-th partial of the sound with F0 of f at time t. $F_i(t, f)$ is basically equal to $i \cdot f$ but they are not exactly equal due to vibrato etc. We set h to 10.

[Step 2] Feature extraction

For every time t (every 10 ms in the implementation), we first excerpt a T-length bit of the harmonic-structure trajectory $\mathcal{H}_t(\tau, f)$ $(t \leq \tau < t + T)$ from the whole trajectory $\mathcal{H}(t, f)$ and then extract a feature vector $\boldsymbol{x}(t, f)$ consisting of 28 features listed in Table 6.1 from $\mathcal{H}_t(\tau, f)$. These features have been designed based on our investigations described in Chapters 3 to 5. Then, the dimensionality is reduced to 12 dimensions using the principal component analysis with the proportion value of 95%. T is 500 ms in the current implementation.

[Step 3] Probability calculation

We train L-to-R HMMs, each consisting of 15 states[2], 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

[2]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.

Table 6.1: Overview of 28 features. Please see Section 3.3.2 for the exact definition.

Spectral features	
1	Spectral centroid (SC)
2–10	Relative cumulative power (RCP) (up to 9th partials)
11	Odd/even power ratio (OER)
12–20	Number of stably existing partials (NEP)
Temporal features	
21	Power decay speed (PDS)
22–24	Average differential of power envelope (ADP) calculated as 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

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 $p(M_i) = 1/(m+1)$.

The above method may cause a problem when some partials overlap partials from other simultaneous sounds. When partials overlap partials from other simultaneous sounds, overlapping partials interfere with each other, and therefore acoustic features extracted from the partials become different from those without the overlapping. To avoid this problem, we use a mixed-sound template, described in the previous chapter, which is a set of training data obtained from polyphonic music. Similarly to the previous chapter, acoustic signals for a mixed-sound template are synthesized based on the scores (standard MIDI files (SMFs), to be exact) of actual musical pieces. The synthesized signals have the labels of the onset times, offset times, and F0s for all notes. After the time series of the feature vectors is extracted from each note, each HMM is trained using a set of the feature vector serieses extracted from the notes played on the corresponding instrument. Training is thus performed notewise whereas recognition is not notewise.

6.3.4 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.

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

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

6.3.5 Symbolization: Conversion to Event-oriented Representation

Although the main feature of the instrogram technique is to represent instrumentation not as symbols but as probabilities, some applications may require a non-probabilistic (*i.e.*, deterministic) representation. We therefore describe a method for transforming an instrogram to an event-oriented representation such as an event whereby a piano sound occurs at time t_0 and continues until t_1. Our transformation method obtains such a

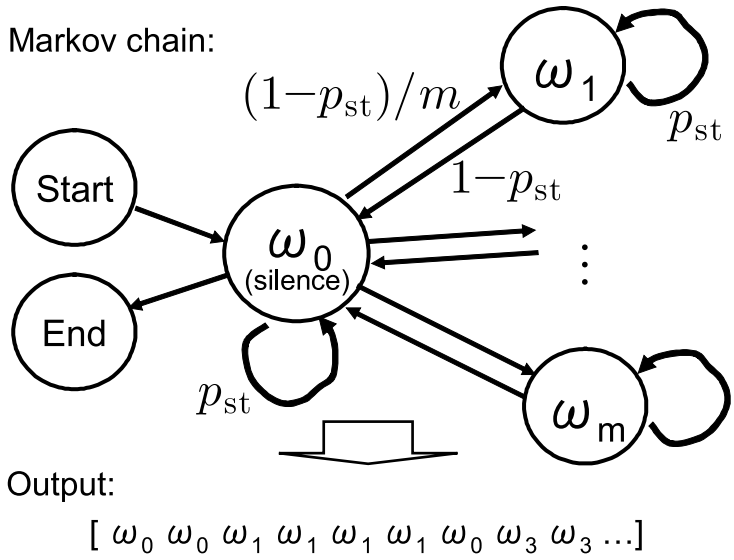
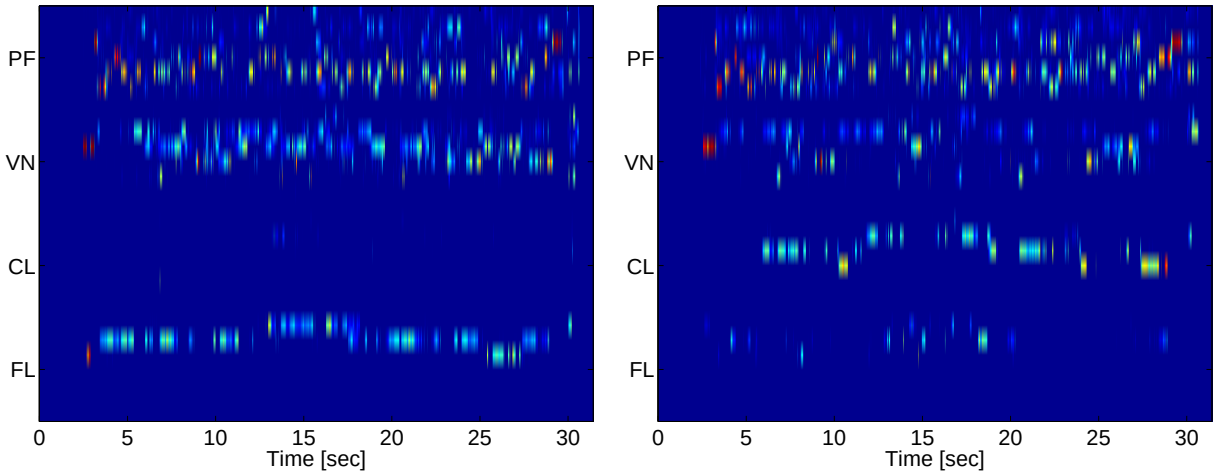


Figure 6.4: Markov chain model used in symbolic annotation. The values are transition probabilities, where p_{st} is the probability of staying at the same state at the next time, which was experimentally determined as $1 - 10^{-16}$.

representation using a Markov chain with states that correspond to instruments. Every frequency subregion I_k, we 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 $\omega_0, \omega_1, \cdots, \omega_m$ (Figure 6.4). 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 using the Viterbi search, we can estimate the start and stop times of an event of an instrument ω_i from the transitions between the states ω_0 and ω_i; the transition from ω_0 to ω_i means to start playing the instrument ω_i while that from ω_i to ω_0 means to stop it. 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.

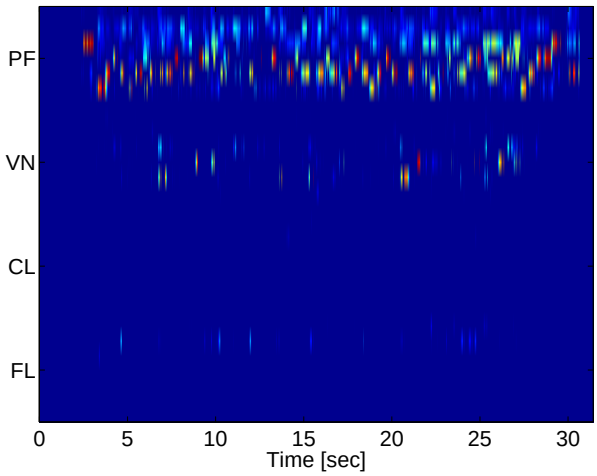
6.4 Experiments

We conducted experiments on obtaining instrograms and their symbolization for both audio data generated on a computer and the recordings of real performances.

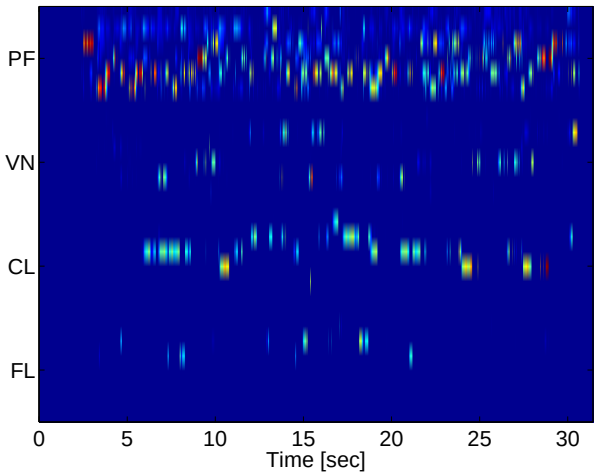


(a) FL–VN–PF

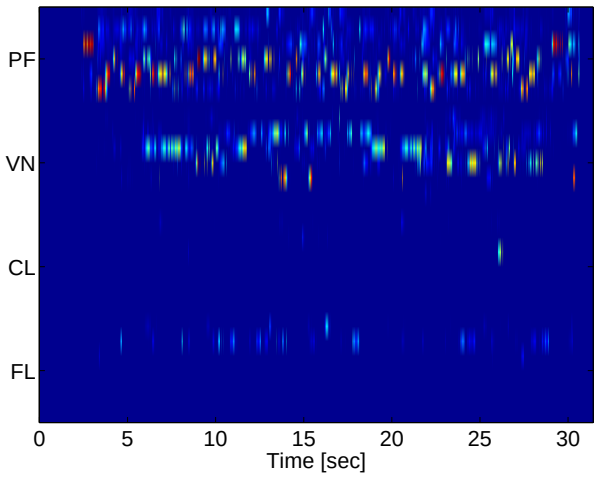
(b) VN–CL–PF



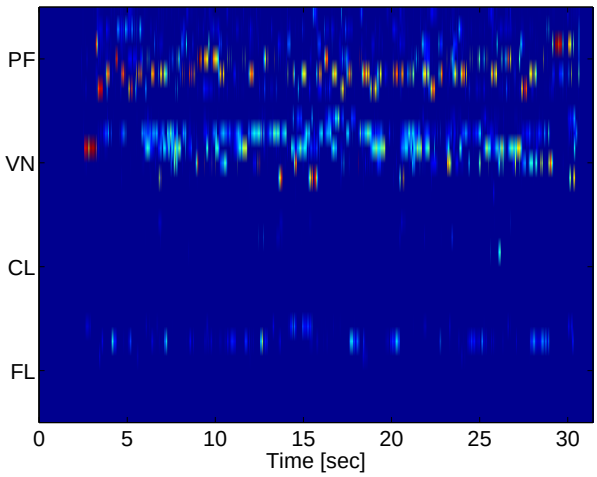
(c) PF–PF–PF



(d) PF–CL–PF



(e) PF–VN–PF



(f) VN–VN–PF

Figure 6.5: 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.

6.4.1 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.* [37]. The audio signals were generated by mixing audio data from RWC-MDB-I-2001[103] (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 Figure 6.5. 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.

We then conducted experiments on symbolization of the instrograms obtained above. 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

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

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

The results are shown in Table 6.2. 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

Table 6.2: Results of event-oriented (symbolic) description 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%

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 the symbolization by merging all the frequency subregions; in other words, we ignored the differences between frequency subregions. This was because the results of instrument recognition are 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

$$R' = \frac{\sum_{i=1}^{m}(\#\text{ frames correctly identified as } \omega_i)}{\sum_{i=1}^{m}(\#\text{ frames that should be identified as } \omega_i)},$$

$$P' = \frac{\sum_{i=1}^{m}(\#\text{ frames correctly identified as } \omega_i)}{\sum_{i=1}^{m}(\#\text{ frames identified as } \omega_i\ )}.$$

The results are listed in Table 6.3. 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

Table 6.3: Results of event-oriented (symbolic) description 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%

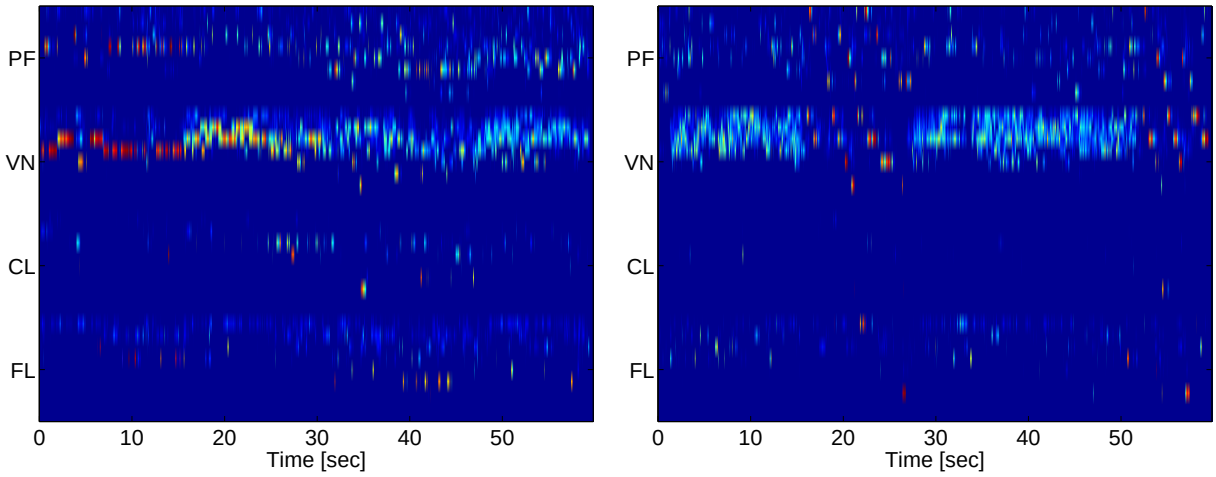
the recall rates were approximately between 30 and 60%.

6.4.2 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[111]. The instrumentation of all pieces is listed in Table 6.4. 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[103] and NTTMSA-P1 (a non-public musical sound database)[3].

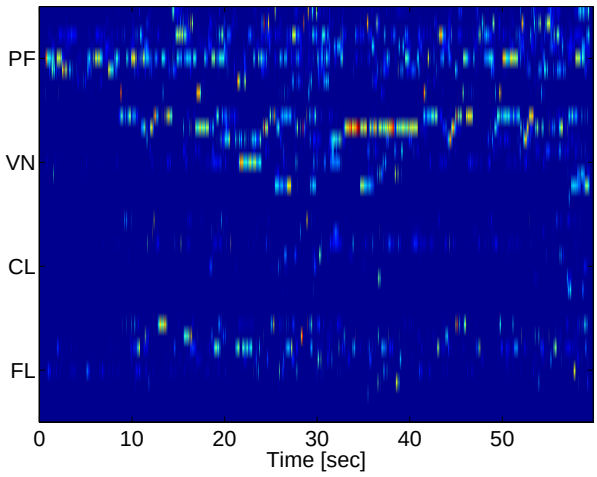
The results, shown in Figure 6.6, show that (a) and (b) have high IEPs for the violin while (e) and (f) have high IEPs for the piano. For (c), the IEPs for the violin increase after 10 sec, whereas those for the piano are initially high. This reflects the actual performances of these instruments. When (d) is compared to (e) and (f), the former has slightly higher

[3]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.

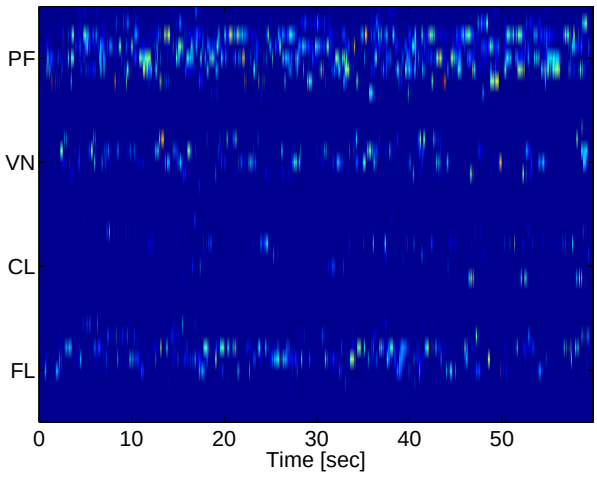


(a) RM-C No. 12 (Str.)

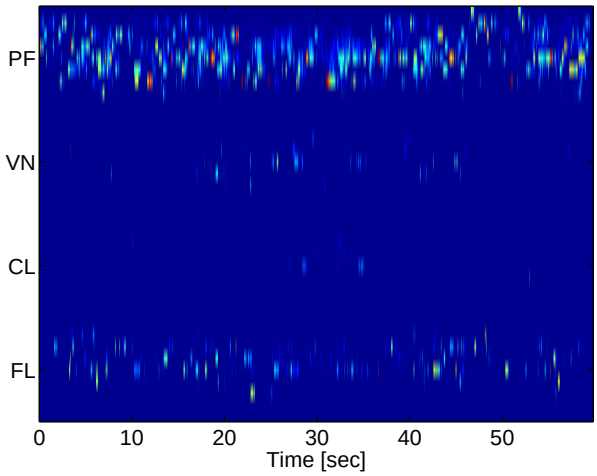
(b) RM-C No. 14 (Str.)



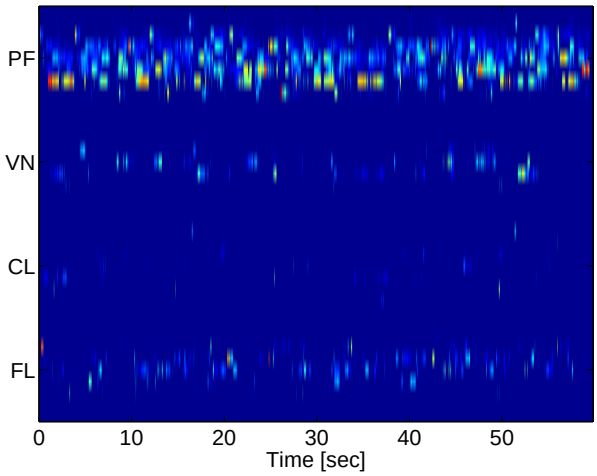
(c) RM-C No. 40 (Pf.+Str.)



(d) RM-C No. 43 (Pf.+Fl.)



(e) RM-J No. 1 (Pf.)



(f) RM-J No. 2 (Pf.)

Figure 6.6: Results of calculating instrograms from real-performance audio signals. RM-C and RM-J stand for RWC-MDB-C-2001 and RWC-MDB-J-2001, respectively.

Table 6.4: 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

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 symbolization of these instrograms. 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[4]. The results are listed in Table 6.5. 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.

6.4.3 Application to MPEG-7 Annotation

Describing multimedia content including musical one in a universal framework is an important task for content-based multimedia retrieval. In fact, a universal framework for multimedia description, MPEG-7, has been established. Here, we discuss music description based on our instrogram analysis in the context of the MPEG-7 standard.

There are two choices for transforming instrograms to MPEG-7 annotations. First, we can simply represent the instrument existence probabilities (IEPs) as a time series of vectors. If one aims at the Query-by-Example such as the one discussed in the next chapter, this annotation method should be used. Because the MPEG-7 standard has no tag for the instrogram annotation, we added several original tags as shown in Figure 6.7.

[4]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 6.5: Results of event-oriented (symbolic) description 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%

This example shows the time series of the 8-dimensional IEPs for the piano (line 13) with the 10-ms time resolution (line 4). Each dimension corresponds to a different frequency region, which is defined by dividing the entire range from 65.5 Hz to 1048 Hz by 1/2 octave (line 2).

Second, we can annotate musical instruments as an event-oriented representation described in Section 6.3.5. If one aims at the Query-by-Instrument (i.e., retrieving pieces by specifying instruments by a user), this annotation method may be more useful than the first one. We also added several original tags as shown in Figure 6.8. This example shows that an event of the piano (line 9) at a pitch between 92 and 130 Hz (line 8) occurs at 6.850 s (line 4) and continues during 0.200 s (line 5). This representation can be obtained using the Viterbi search as described in Section 6.3.5.

6.4.4 Discussion

The main contribution in this chapter 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

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

Figure 6.7: Excerpt of example of instrogram annotation.

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

Figure 6.8: Excerpt of example of symbolic annotation.

for polyphonic music. Based on similar motivations, Vincent and Rodet [42] and Essid *et al.* [41] proposed new instrument recognition techniques. Vincent and Rodet'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.

6.5 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. The conclusions of this chapter are summarized as follows:

- We proposed a new representation of instrumentation called the *instrogram*. The instrogram graphically visualizes the temporal trajectories of IEPs, that is, how likely each target instrument is played. This probabilistic representation facilitated a probabilistic formulation of instrument recognition in polyphonic music.
- We formulated a method for obtaining an instrogram (in other words, calculating IEPs) based on probabilistic calculation. The IEP is defined as the product of the NIEP and CIEP, which are calculated using the PreFEst and HMMs, repectively. Because errors of one probability do not influence the calculation of the other probability, the IEP can be robustly calculated.

Chapter 7

Application

This chapter describes an application of the instrogram analysis to similarity-based MIR.

7.1 Introduction

In this chapter, we apply the instrogram analysis to similarity-based MIR. Similarity-based MIR, also known as the Query-by-Example, aims to search for musical pieces similar to that specified by the user. Similarity-based MIR is useful because it requires no special musical knowledge. It has therefore been widely studied as reviewed in Chapter 2 [79–86]. These studies basically used low-level features (*e.g.*, MFCCs) extracted from signals containing multiple instrument sounds. Such features can be relatively easily extracted and similarity in them coarsely correspond to perceptual similarity in music. In fact, they attained successful results to some extent. Low-level features, however, are unsuitable to focus on a certain musical element (*e.g.*, melody, rhythm, harmony, and instrumentation). Some users may focus on the harmony while listening to music, but other users may focus on the instrumentation. Calculation of music similarity should therefore take into account this dependency of the importance of musical elements on users. Because low-level features do not clearly correspond to a specific musical element, it is difficult to achieve the user-adaptive music similarity calculation.

For this music similarity calculation, we have to design a higher-level feature corresponding to each specific musical element. In other words, a feature representing a melody should reflect only the melody and a feature representing instrumentation should reflect only the instrumentation. We therefore adopt the instrogram, proposed in the previous chapter, as the feature representing instrumentation. Specifically, as the first step for user-adaptive similarity-based MIR, we design a method for calculating the similarity between

instrograms and build a prototype system that searches for musical pieces based on the similarity of instrogram. Although we have to design features representing other musical elements and a method for weighting the features according to the user's preference to complete the user-adaptive similarity-based MIR, we leave them as future issues in this thesis.

7.2 Music Information Retrieval based on Instrumentation Similarity

We achieve MIR based on instrumentation similarity using instrograms. We consider that instrumentation is deeply connected to listeners' impression. When the same musical piece is played on different instruments, listeners may have different impressions on the piece. This implies a deep connection between instrumentation and listeners' impression. MIR based on instrumentation similarity will therefore be useful, for example, for playlist generation for background music. Here, instead of calculating similarity, we calculate the distance (dissimilarity) between instrograms by using DTW as follows:

(a) 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.

(b) 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 that gives the relationship between elements. One may want to give a high similarity to pieces where the same instrument is played at different pitch regions (e.g., $p(\omega_1; t, I_1)$ vs. $p(\omega_1; t, I_2)$) or pieces where different instruments within the same instrument family (e.g., violin vs. viola) are played. They can reflect such relations in the distance measure by setting r_{ij} for the corresponding elements to a value more than zero. When R is the unit matrix, $(\boldsymbol{p}, \boldsymbol{q})$ and $||\boldsymbol{p}||$ are equivalent to the standard inner product and norm.

(c) 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[79, 88], The timbral similarity was calculated on the basis of spectral features, such as MFCCs, directly extracted from 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.

7.3 Implementation and Experiments

7.3.1 Implementation

We built a prototype system of MIR based on our instrumentation similarity using Java. This prototype system has two retrieval functions. One is the Query-by-Example, where the query is the musical piece specified by the user (Figure 7.1). The other is the Query-by-IEP, where the user directly specifies instrument existence probabilities (IEPs) (Figure 7.2). This method can be used in a situation where the user wants a piano piece or a strings piece. In the former case, after the user selects a musical piece as a query, the system calculates the (dis)similarity between the selected piece and each of the pieces in a collection using the method described in Section 7.2 and then shows the list of musical pieces in order of similarity. In the latter case, instead of a musical piece, the user specifies the IEP for each instrument using the sliders each of which corresponds to each target instrument. The dissimilarity between the specified IEPs and each of the pieces in the collection is calculated by

$$\mathrm{dist}(\boldsymbol{p}, \{\boldsymbol{q}_t\}) = \sum_t \{1 - (\boldsymbol{p}, \boldsymbol{q}_t)/||\boldsymbol{p}|| \cdot ||\boldsymbol{q}||\},$$

where $\boldsymbol{p}$ is the vector consisting of the IEPs specified by the user and $\{\boldsymbol{q}_t\}$ is a temporal sequence of the IEP vectors of a musical piece in the music collection.

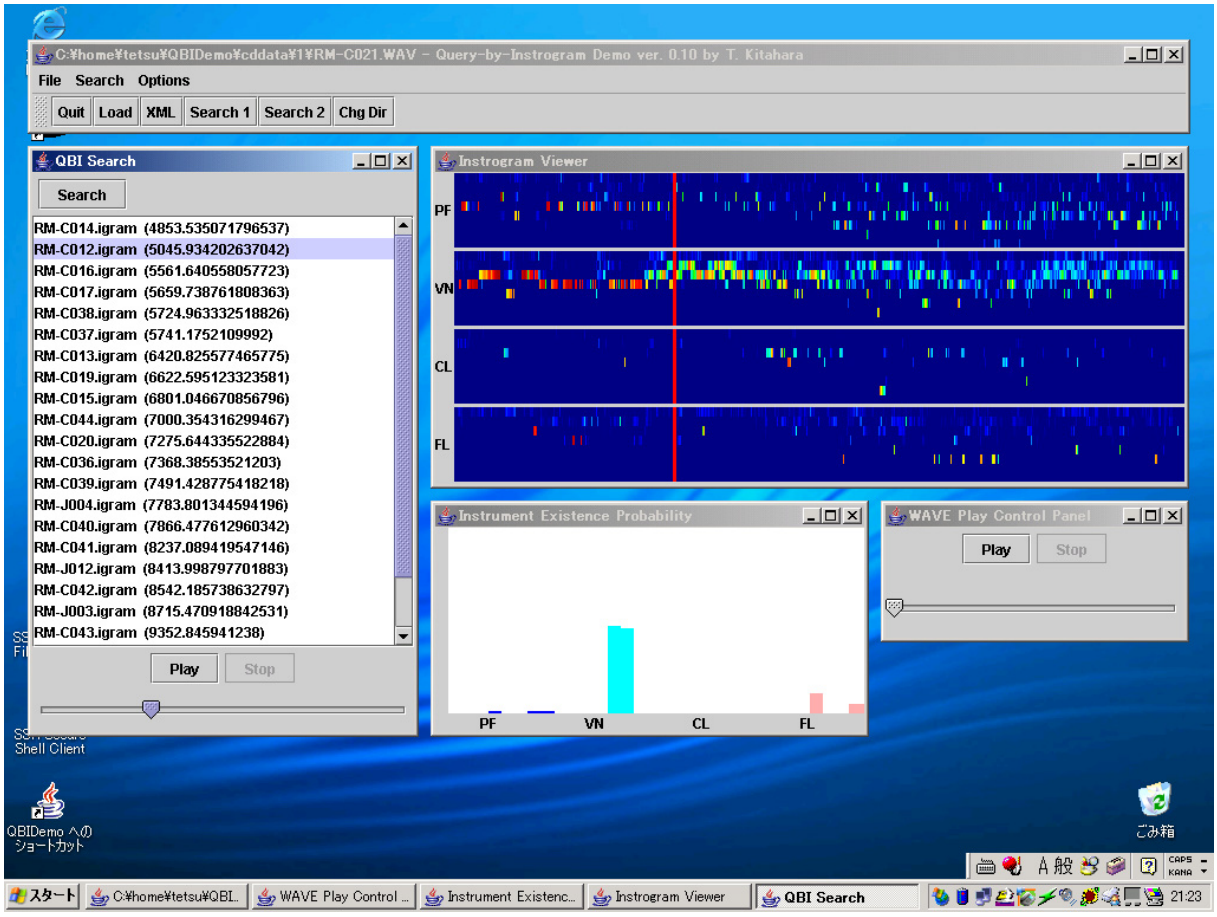


Figure 7.1: Instrogram-based MIR prototype system (Query-by-Example).

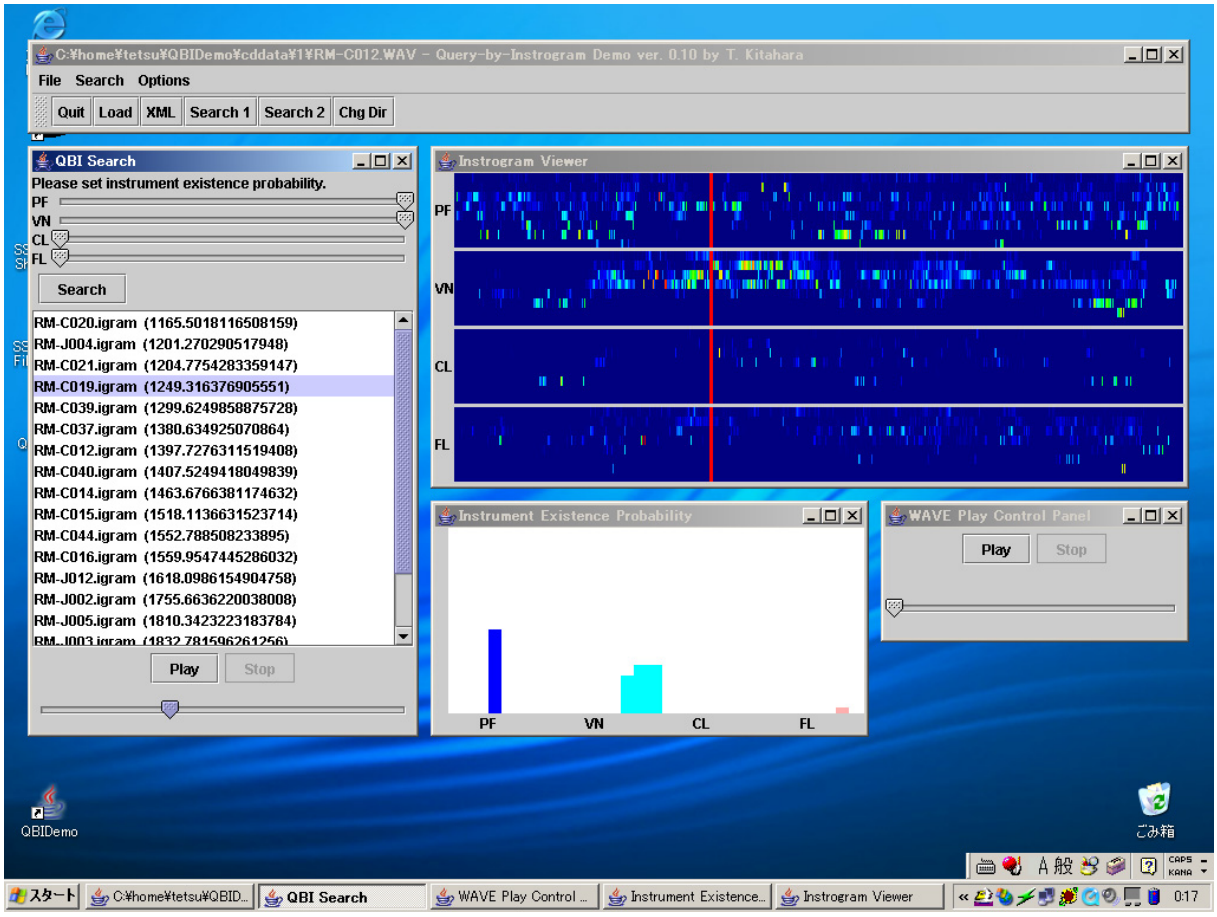


Figure 7.2: Instrogram-based MIR prototype system (Query-by-IEP).

Users can see the IEPs of the piece that they are listening to in two ways: the original instrogram visualization and the synchronized bar-graph visualization. The former is the spectrogram-like representation like Figure 6.2. The latter shows audio-synchronized bar graphs of IEPs in real time like those of the power spectrum display on digital music players. Because the original instrogram visualization shows the temporal variation of IEPs at once, The users can easily jump, for example, to the time point when the violin begins to play.

7.3.2 Experiments on Similarity Calculation

We tested the calculation of the dissimilarities between instrograms. The data used were the real performance recordings used in Section 6.4.2, the details of which were described in Table 6.4. The results, listed in Table 7.1 (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 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 7.1 (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 7.2, 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).

7.4 Conclusion

In this chapter, we applied the instrogram analysis to similarity-based MIR. The conlusions of this chapter is summarized as follows:

- We pointed out that a music similarity measure should be separately designed for each musical element and adaptive to users because the importance of musical elements are dependent on users. This is difficult by using low-level features that have been commonly used because they correspond coarse characteristics of music, not specific musical elements. New higher-level features corresponding to specific musical elements are therefore needed.

- As the first step for tackling the above-mentioned problem, we designed a method for measuring similarity between instrograms. The instrogram directly correspond to instrumentation and therefore is considered a higher-level feature that satisfies the above requirement.

- We demonstrated a prototype system of MIR based on similarity between instrograms. Our prototype system not only searches musical pieces based on instrogram similarity but also visualizes the instrumentation of a piece that the user is listening to. Through this demonstration, we confirmed that our musical instrument recognition technique can actually be used for content-based MIR and a visual music player.

Table 7.1: Instrumentation dissimilarities 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

Chapter 8

Discussion

This chapter first discusses the main contributions of this study, then discusses the remaining issues and future directions.

8.1 Major Contributions

In this thesis, we pointed out four issues in instrument recognition, (1) the pitch dependency of timbre, (2) the input of non-registered instruments, (3) the overlapping of simultaneous played notes, and (4) the unreliability of the precedent note estimation process, and proposed solutions to these issues. To solve the first issue, we proposed the F0-dependent multivariate normal distribution. We solved the second issue by recognizing non-registered instruments at the category level. We tackled the third issue by introducing feature weighting based on how each feature is affected by the overlapping. To solve the fourth issue, we presented a new framework of instrument recognition based on a probabilistic representation of instrumentation called an *instrogram*. The main contributions of these are summarized as follows:

Towards Computational Auditory Scene Analysis (CASA)

- **Computational Modeling of Simultaneous and Sequential Grouping**

  Bregman [54] formulated auditory scene analysis as two kinds of grouping problems, that is, simultaneous grouping and sequential grouping. The former aims at segregation of a signal containing multiple sounds while the latter aims at temporal modeling of a signal. When we interpret the instrogram analysis from the viewpoint of CASA, the calculation of the *nonspecific instrument existence probability* (NIEP) and *conditional instrument existence probability* (CIEP) can be considered

as new approaches for simultaneous grouping and sequential grouping, respectively. Sakuraba and Okuno [116] formulated automatic music transcription based on simultaneous grouping and sequential grouping. Their framework first performs simultaneous grouping and then sequential grouping. This cascade framework has a severe limitation: errors of simultaneous grouping degrade the performance of sequential grouping. On the other hand, the instrogram analysis is based on probabilistic formulation; both grouping problems are formulated as calculation of probabilities and integrated by multiplication of the probabilities. Our probabilistic formulation is a promising approach for CASA.

- **No Assumption of Complete Sound Source Separation**

  Previous studies separately investigated sound source (especially speech) recognition and sound source separation. Sound source separation research aimed to generate the signal from each source given a mixture of sounds while sound source recognition research aimed to recognize a single source. If sound source separation technology could generate separated signals without experiencing any distortion, this approach is completely reasonable. In practice, however, it is almost impossible to separate mixed sounds without distortion, and therefore we take into consideration the influence of the overlapping of multiple sounds at the sound source recognition phase. From this viewpoint, we modeled the phenomenon in the overlapping of multiple sounds and a solution to this overlapping problem. First, the use of the harmonic structure model restricted the influence of the overlapping into the overlapping of common-frequency partials. Second, we quantitatively analyzed the influence by taking training data from polyphonic sounds. Third, we calculated the feature weights that minimized the influence by means of linear discriminant analysis.

- **Sound Recognition Scheme for Non-registered Sources**

  Although CASA aims to develop a unified framework for handling a variety of sounds, no previous studies have discussed how to handle sounds that were not contained in training data. Some speech recognition studies dealt with a similar problem called out-of-vocabulary, but they did not deal with recognition of such sounds based on a hierarchical taxonomy of sounds. We provided a solution of category-level recognition of such sounds, inspired by human understanding of musical sounds. Humans often understand musical sounds coarsely even when listening

for the first time. For example, they can distinguish violin-like sounds from piano-like ones even if they have not listened to the sounds in the past. Our approach is an implementation of such human-like flexible sound recognition. Although we have tested this approach only on solo musical sounds, it will be possible to apply this concept to polyphonic music and other general sounds.

- **Development of Basic Technologies for Handling Musical Sounds**

  Because musical sounds are indispensable parts of auditory scenes, developing basic technologies for handling musical sounds is an important subtask for achieving CASA. To this end, we developed feature extraction from musical sounds and a method for the modeling of feature distributions using the F0-dependent multivariate normal distribution. When F0 estimation is not accurate enough, the use of the F0-dependent multivariate normal distribution may decrease the performance of instrument recognition. This problem, however, can be solved by applying this concept to the instrogram analysis. In the instrogram analysis, the likelihoods of hidden Markov models (HMMs), each corresponding to a specific target instrument, are calculated. This calculation is performed for every possible F0, instead of estimating the F0s for the notes contained in the signal. This means that our concept of F0-dependent timbre modeling will be freed from the unreliability of F0 estimation if we extend it to HMMs (*i.e.*, F0-dependent HMMs) and use the extended one for calculating an instrogram.

Towards Content-based Music Information Retrieval (MIR)

- **MIR based on Instrumentation Similarity**

  We achieved an MIR based on instrumentation similarity by designing a similarity measure between instrograms. Although both timbral similarity calculation and instrument recognition have been actively investigated, no attempts have been made to calculate the instrumentation similarity on the basis of instrument recognition techniques, because previous instrument recognition aimed to determine the instruments playing in given signals. The instrogram, which represents instrumentation as a set of continuous values, is an effective approach for the design of a continuous similarity measure.

- **Middle-level Music Descriptor**

In previous content-based MIR studies, lower-level audio descriptors such as MFCCs were mainly used. These features are easy to extract automatically and have attained successful results to some extent. These, however, have a limitation due to the unclear correspondence to musical meaning; the difference of MFCCs, for example, may be caused by the difference of both instrumentation and chords. We therefore need higher-level but automatically extractable descriptors (which we call "middle-level" here). The instrogram is an example of such descriptors because it can be automatically calculated and directly corresponds to instrumentation.

- **Detailed (High-resolution) Instrument Annotation**

  Our instrogram analysis can calculate instrument existence probabilities with any time resolution (*e.g.*, 10 ms) and any frequency resolution (*e.g.*, 100 cent). This high-resolution annotation is difficult and too time-consuming by manual annotation even when done by music experts.

Towards Music Visualization

- **Establishment of Visual Representation of Instrumentation**

  Because transforming a musical piece into a graphical representation allows us to grasp the content of multiple musical pieces at once, music visualization is an important issue in making human music retrieval more efficient. Different researchers have therefore attempted music visualization through different approaches.

  The most traditional music visualization is a musical score. The musical score is very useful but automatically obtaining the musical score from audio signals is still a challenging problem. On the other hand, the most basic visualization of audio signals is a spectrogram. The spectrogram can be automatically obtained and is useful to check the distribution of spectral peaks on the time-frequency plane, although it is not easy for untrained people to understand musical elements from that.

  Various other methods of music visualization have been proposed. Hiraga *et al.* [117, 118] dealt with some visualization methods of musical performances, but they used MIDI signals and the methods are not easily applied to audio signals. Sagayama*et al.* [10] proposed a new visualization of audio signals, called *specmurt*. The specmurt is made by suppressing overtones from a spectrogram with the specmurt analysis,

which was described in Section 2.1.1. This is similar to the piano-roll expression and is therefore more useful than the spectrogram for understanding musical content. Goto [119] achieved visualization of chorus sections and repeated sections of popular music in his music listening system *SmartMusicKIOSK*.

Tzanetakis [120] proposed the *GenreGram* and *TimbreBall* monitors. The GenreGram monitor shows genre classification results. Each genre is represented as a cylinder that moves up and down in real time based on a classification confidence measure ranging from 0.0 to 1.0. Each cylinder is texture-mapped with a representative image for each genre. The TimbreBall monitor visualizes in real time the evolution of feature vectors extracted from an audio signal. In this animation, each feature vector is mapped to the x, y, z coordinates of a small ball inside a cube. The ball moves in the space as the sound is playing, following the evolution of the corresponding feature vectors. Tzanetakis also proposed TimbreGram, which maps audio files to sequences of vertical color stripes where each stripe corresponds to a short slice of sound. The similarity of different files is shown as overall color similarity. This is close to the instrogram concept, but the user can know only the similarity from the TimbreGram; the TimbreGram does not state which instruments are playing in the signal. Thus, visualization that shows instrumentation has not been achieved. We achieved the visualization of instrumentation, called the *instrogram*, by displaying instrument existence probabilities as a spectrogram-like representation.

Towards Computational Modeling of Understanding Music

- **Untrained Listeners' Music Understanding**

  Goto [96] claimed that people, especially those who are musically untrained, would understand music without mentally representing it as a score, and pointed out the importance of developing a method for understanding music not based on scores. He then claimed that good music descriptors should be musically intuitive, fundamental to a professional method of understanding music, and useful for various applications. From this standpoint, he developed a method for recognizing melody and bass lines, hierarchical beat structures, and chorus and repeated sections. The instrogram also satisfies the above-mentioned requirements and is useful for developing an integrated framework of understanding music in a untrained-people-like

fashion through integration with Goto's work.

Towards Other Music Applications

- **Educational Applications**

  Visualization of instrumentation of a musical piece based on the instrogram is useful for learning the instrumentational structure of the piece in detail (for example, a musical piece starts with the piano followed by the violin). Because the user can easily jump to the time point where the violin begins to play by clicking the corresponding point in the instrogram window, it is also useful for repeated practice of playing an instrument. In addition, once the instrogram representation is transformed into an animated illustration of the instruments, it will be useful for music education.

8.2 Remaining Issues and Future Directions

We have many remaining issues and future directions of research. Some of these are summarized below.

To Further Improve Musical Instrument Recognition

- **Introduction of Top-down Model**

  The instrogram analysis is a model of a bottom-up process, and no top-down processes were introduced here. Musical knowledge (*e.g.*, a certain instrument is rarely played in a certain pitch range) and musical context will be able to be introduced by modeling them in a probabilistic framework.

- **F0-dependent GMMs and HMMs**

  Our concept of F0-dependent timbre modeling is not limited to the F0-dependent multivariate normal distribution. It can be easily extended to Gaussian mixture models (GMMs) and HMMs. In particular, the extension to HMMs will make it possible to apply this concept to the instrogram analysis.

- **Evaluation Using Music Containing Vocal and Drums**

  We have not used in evaluation any musical pieces that contain vocals or drum sounds. If they are contained, the performance of instrument recognition may de-

crease. It is therefore important to extend our technique for musical pieces that contain vocals and drum sounds.

To Further Improve Music Information Retrieval

- **Integration with Other Musical Elements**

  We have dealt with only musical instrument timbres and have left other musical elements such as melodies, rhythm, and harmony as future work. Because we have already developed methods for other musical elements, for example chord progression [121], we plan to integrate them. Once the instrogram is integrated with other musical elements, the similarity for each musical element can be designed and integrated with weighting. This weighting is important because which musical element is emphasized is dependent on the listener. It is important to develop a user-adaptive music similarity measure after the above integration.

- **Users' Trial Tests of MIR Application**

  We have not conducted users' trial tests of our MIR application. After we integrate the instrogram with other musical elements and develop an integrated music exploration system, we will conduct users' trial tests to evaluate the effects of our instrumentation similarity measure.

- **Extension to Integrated Music Exploration and Browsing System**

  We applied our instrogram analysis to the Query-by-Example retrieval here, but our technique has a broader range of applications, for example, automatic playlist generation and music recommendation. In addition, instrogram visualization can be extended to visualization that helps listeners understand and enjoy music deeply. If it is transformed into enjoyable animations, it will, for example, be useful for music education.

Other Future Directions

- **Implementation of Instrogram Analysis based on Parallel Processing**

  The instrogram analysis calculates two kinds of probabilities, nonspecific instrument existence probability (NIEP) and conditional instrument existence probability (CIEP), which can be calculated independently of each other. In addition, NIEPs

for each frame (time) are basically independent and CIEPs for each F0 are also independent. The calculation can therefore be separated into many subcalculations that do not need to synchronize with one another. This means that the time to calculate the instrogram can be reduced using parallel processing. We therefore plan to implement this parallel-processing-based instrogram analysis.

- **Integration with Other Sound Recognition**

  From the viewpoint of computational auditory scene analysis, the frameworks for recognizing musical sounds and other sounds should be unified. Although we have already dealt with singer identification [122] and XML-based description of sound effects [123], the frameworks were separately developed.

- **Application to Automatic Music Transcription**

  The instrogram has been applied to MIR, but it provides more detailed information that makes possible its use for music transcription. In fact, the flute part of Figure 6.1 shows a trajectory similar to the melody. We therefore plan to develop a music transcription system based on such instrogram representations.

- **Comparison with Human Timbre Perception**

  Comparing our results with human timbre perception would be interesting. In particular, comparing TimbreTree obtained in Chapter 4 with the taxonomy based on human perception will be an important issue for future work. Although hierarchical classification of timbres based on human perception has been attempted [67, 68, 75], the scale of the experiments was small because of burdens of subjects. If hypotheses about human perception can be made using computational processing, burden on subjects (*e.g.*, variations of conditions of listening tests) could be reduced. A future issue is to establish such a method for planning strategies of listening tests that reduce the burden on subjects based on experimental results on a computer.

Chapter 9

Conclusions

In this thesis, we dealt with recognition of musical instruments from audio signals. Although F0 estimation and automatic transcription for musical audio signal processing have a long history, musical instrument recognition has a relatively short history; studies were only started in earnest in the late 1990s. Especially in polyphonic music, very few studies have dealt with musical instrument recognition.

At the first stage, we investigated musical instrument recognition in monophonic sounds. We pointed out the following two issues at this stage:

Issue 1 Pitch dependency of timbre,

Issue 2 Input of non-registered instruments.

To resolve Issue 1, we proposed an F0-dependent multivariate normal distribution, where the pitch dependency of timbre is approximated as a function of F0. To resolve Issue 2, we proposed category-level recognition of non-registered instruments. When a given sound is registered, its instrument name, *e.g.* violin, is identified. Even if it is not registered, its category name, *e.g.* strings, can be identified. The effects of these proposals were tested on a monophonic musical instrument sound database.

At the second stage, we scaled up the target of instrument recognition from monophonic to polyphonic sounds. We pointed out the following two issues in order to deal with polyphonic music:

Issue 3 Overlapping of simultaneously played notes,

Issue 4 Unreliability of the precedent note estimation process.

To resolve Issue 3, we proposed a method for feature weighting based on how each feature suffered from the overlapping. This feature weighting has been achieved with a mixed-

sound template, which is a set of training data extracted from polyphonic musical audio signals, and linear discriminant analysis (LDA). To resolve Issue 4, we proposed a novel musical instrument recognition framework that does not explicitly use note estimation. The effects of these proposals were tested on audio signals of polyphonic music. In addition, we developed a prototype system of MIR using our musical instrument recognition technique.

We summarize each chapter as follows.

In Chapter 1, we first described the motivation for and goal of this study. We then briefly described the above-mentioned issues and approaches.

In Chapter 2, we reviewed state-of-the-art work in related fields. The review covers a wide range of topics, from musical audio signal processing to content-based MIR. We then discussed the positioning of this thesis from different viewpoints.

In Chapter 3, we proposed a method for coping with the pitch dependency of timbre, called an *F0-dependent multivariate normal distribution*. The key idea behind this is to approximate feature variations caused by pitch as a function of F0. Because it is not easy in general to extract a factor causing feature variations as a physical value, it is also difficult to approximate the relationship between varied features and the factor causing the feature variations like in our approach. On the other hand, we focused on pitch, which can be extracted as F0, as a factor causing feature variations. The F0-dependent multivariate normal distribution involves a mean vector each element of which is designed as a function of F0. This F0-dependent mean function represents the pitch dependency of timbres while the F0-normalized covariance, which is the other parameter of the F0-dependent multivariate normal distribution, represents the non-pitch dependency of timbres. Experimental results from 6,247 monophonic sounds of 19 instruments showed that the recognition rate was improved from 75.73% to 79.73%. This performance is considered state-of-the-art.

In Chapter 4, we resolved the problem of non-registered instruments by recognizing instruments at the category level even when they were not registered. Previously, the importance of dealing with non-registered instruments has not been pointed out. All of the previous studies assumed that all of the instruments to be recognized in given audio signals are included in the training data. In practice, however, it is difficult to thoroughly prepare the training data to cover all existing instruments. We therefore pointed out the importance of dealing with non-registered instruments and proposed their category-

level recognition as a solution. For the category-level recognition, it is important to appropriately design a musical instrument taxonomy. Thus we proposed a method for constructing musical instrument taxonomy from acoustic similarities using hierarchical clustering. We called the taxonomy based on acoustic similarities TimbreTree. When we gave electric sounds that are similar to but different from the sounds of real instruments to a system that had been trained in the sounds of real instruments, the system correctly recognized the categories of the given sounds, distinguishing them from those of the trained real instruments. We also discussed the differences between TimbreTree and the musical instrument taxonomy built based on knowledge of the mechanisms of instruments.

In Chapter 5, we provided a new solution to Issue 3. Our solution is to weight features such that features suffering more from overlapping have lower weights and those suffering less have higher weights. This feature weighting was achieved by collecting training data extracted from polyphonic sounds and applying LDA to them. Although the approach of collecting training data from polyphonic sounds is simple, no previous studies had attempted it. One possible reason may be that a tremendously large amount of data is required to prepare a thorough training data set containing all possible sound combinations. From our experiments, however, we found that a data set extracted from a few musical pieces was sufficient to improve the robustness of instrument recognition in polyphonic music. Furthermore, we improved instrument recognition using musical context. Use of musical context has made it possible to avoid musically unnatural errors by taking the temporal continuity of melodies into consideration. By combining these two methods and the F0-dependent multivariate normal distribution, we achieved recognition rates of 84.1% for duo, 77.6% for trio, and 72.3% for quartet music. This performance is also considered state-of-the-art.

In Chapter 6, we proposed a novel musical instrument recognition framework called the *instrogram* analysis. The instrogram is a time-frequency representation of instrument existence probabilities (IEPs), each of which is the probability that the sound of each target instrument exists at each time and each F0. By formulating instrument recognition as the problem of calculating the IEP at each time and each F0 for every target instrument, we have made it possible to omit the preceding processing such as onset detection and F0 estimation, which are still challenging problems for polyphonic music. The IEP is decomposed into two probabilities: the nonspecific instrument existence probability (NIEP) and the conditional instrument existence probability (CIEP). The NIEP is cal-

culated using the PreFEst and the CIEP is calculated using hidden Markov models. The main reason for the robustness of the instrogram analysis is that the NIEP and CIEP can be calculated independently; calculation of one probability does not negatively influence calculation of the other, unlike the conventional framework where note estimation and instrument determination are sequentially connected. Experiments were conducted on not only synthesized music but also real performance recordings of classical and jazz music.

In Chapter 7, we developed a prototype system of similarity-based MIR by applying the instrogram analysis. Because most previous similarity-based MIR systems used low-level features such as MFCCs, similarities for musical elements such as the melody, rhythm, harmony, and instrumentation could not be separately measured. The music similarity measure that we developed based on the instrogram representation can be considered the significant first step towards music similarity separately measured for each musical element, because instrumentation is an important factor determining the impression of music.

In Chapter 8, we discussed the major contributions of this study towards different research fields. We described that this study contributes, in particular, to the fields of computational auditory scene analysis, content-based MIR, and music visualization. We also discussed remaining issues and future directions of research.

We thus achieved development of a processing module that recognizes musical instruments from audio signals of polyphonic music. We hope that our study will trigger further attempts to clarify the mechanism of understanding music and ultimately to develop a computer that can understand music.

Bibliography

[1] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Math. Comput.*, 19:297–301, 1965.

[2] J. A. Moorer. *On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer*. PhD thesis, Stanford University, 1975.

[3] K. D. Martin. Automatic transcription of simple polyphonic music. In *3rd Joint Meeting of Acoust. Soc. Am. and Jpn.*, 1996.

[4] A. Klapuri, T. Virtanen, A. Eronen, and J. Seppanen. Automatic transcription of musical recordings. In *Proc. Consistent & Reliable Acoustic Cues*, 2001.

[5] A. Klapuri. Multipitch estimation and sound separation by the spectral smoothness principle. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2001.

[6] M. Marolt. A connectionist approach to automatic transcription of polyphonic piano music. *IEEE Trans. Multimedia*, 6(3):439–49, 2004.

[7] M. Goto. A real-time music-scene-description system: Predominant-F0 estimation for detecting melody and bass lines in real-world audio signals. *Speech Comm.*, 43(4):311–329, 2004.

[8] H. Kameoka, T. Nishimoto, and S. Sagayama. Separation of harmonic structures based on tied gaussian mixture model and information criterion for concurrent sounds. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume IV, pages 297–300, 2004.

[9] H. Kameoka, T. Nishimoto, and S. Sagayama. Harmonic-temporal-structured clustering via deterministic annealing EM algorithm for audio feature extraction. In

Proceedings of the International Conference on Music Information Retrieval (ISMIR), pages 115–122, 2005.

[10] S. Sagayama, K. Takahashi, H. Kameoka, and T. Nishimoto. Specmurt anasylis: A piano-roll-visualization of polyphonic music by deconvolution of log-frequency spectrum. In *Proc. SAPA*, 2004.

[11] S. Saito, H. Kameoka, T. Nishimoto, and S. Sagamaya. Specmurt analysis of multi-pitch music signals with adaptive estimation of common harmonic structure. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 84–91, 2005.

[12] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Proceedings of the Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 177–180, 2003.

[13] S. A. Abdallah and M. D. Plumbley. Polyphonic music transcription by non-negative sparse coding of power spectra. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 318–325, 2004.

[14] A.T. Cemgil, B. Kappen, and D. Barber. Generative model based polyphonic music transcription. In *Proceedings of the Workshop on Applications of Signal Processing to Audio and Acoustics*, 2003.

[15] C. Yeh, A. Robel, and X. Rodet. Multiple fundamental frequency estimation of polyphonic music signals. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume III, pages 225–228, 2005.

[16] K. Kashino and S. J. Godsill. Bayesian estimation of simultaneous musical notes based on frequency domain modelling. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume IV, pages 305–308, 2004.

[17] M. Davy. Multiple fundamental frequency estimation based on gerative models. In A. Klapuri and M. Davy, editors, *Signal Processing Methods for Music Transcription*. Springer, 2006.

[18] A. Klapuri. Auditory model-based methods for multiple fundamental frequency estimation. In A. Klapuri and M. Davy, editors, *Signal Processing Methods for Music Transcription*. Springer, 2006.

[19] T. Virtanen. Unsupervised learning methods for source separation in monaural music signals. In A. Klapuri and M. Davy, editors, *Signal Processing Methods for Music Transcription*. Springer, 2006.

[20] K. D. Martin. *Sound-Source Recognition: A Theory and Computational Model*. PhD thesis, MIT, 1999.

[21] J. C. Brown. Computer identification of musical instruments using pattern recognition with cepstral coefficients as features. *Journal of the Acoustic Society of America*, 103(3):1933–1941, 1999.

[22] J. C. Brown. Feature dependence in the automatic identification of musical woodwind instruments. *Journal of the Acoustic Society of America*, 109(3):1064–1072, 2001.

[23] A. Eronen and A. Klapuri. Musical instrument recognition using cepstral coefficients and temporal features. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 735–756, 2000.

[24] I. Fujinaga. Example-based learning in adaptive optical music recognition system. In *Proceedings of the International Computer Music Conference (ICMC)*, 1996.

[25] I. Fujinaga. Machine recognition of timbre using steady-state tone of acoustic musical instruments. In *Proceedings of the International Computer Music Conference (ICMC)*, 1998.

[26] A. Fraser and I. Fujinaga. Toward real-time recognition of acoustic musical instruments. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 175–177, 1999.

[27] I. Fujinaga and K. MacMillan. Realtime recognition of orchestral instruments. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 141–143, 2000.

[28] J. Marques and P. J. Moreno. A study of musical instrument classification using Gaussian mixture models and support vector machines. CRL Technical Report Series CRL/4, Compaq Cambridge Research Laboratory, 1999.

[29] A. A. Livshin, G. Peeters, and X. Rodet. Studies and improvements in automatic classification of musical sound samples. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 171–174, 2003.

[30] A. Eronen. Musical instrument recognition using ICA-based transform of features and discriminatively trained HMMs. In *Proc. 7th Int'l Symp. Signal Process. and Its Applications*, pages 133–136, 2003.

[31] A. G. Krishna and T. V. Sreenivas. Music instrument recognition: From isolated notes to solo phrases. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume IV, pages 265–268, 2004.

[32] S. Essid, G. Richard, and B. David. Musical instrument recognition by pairwise classification strategies. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1401–1412, 2006.

[33] G. Agostini, M. Longari, and E. Pollastri. Musical instrument timbres classification with spectral features. *EURASIP J. Applied Signal Process.*, 2003(1):5–14, 2003.

[34] B. Kostek. Musical instrument classification and duet analysis employing music information retrieval techniques. *Proc. IEEE*, 92(4):712–729, 2004.

[35] A. A. Livshin and X. Rodet. Musical instrument identification in continuous recordings. In *Proc. 7th Int'l Conf. Digital Audio Effects (DAFx 2004)*, pages 222–227, 2004.

[36] I. Kaminskyj and T. Czaszejko. Automatic recognition of isolated monophinic musical instrument sounds using kNNC. *J. Intell. Information Syst.*, 24(2/3):199–221, 2005.

[37] K. Kashino and H. Murase. A sound source identification system for ensemble music based on template adaptation and music stream extraction. *Speech Comm.*, 27:337–349, 1999.

[38] T. Kinoshita, S. Sakai, and H. Tanaka. Musical sound source identification based on frequency component adaptation. In *Proc. IJCAI CASA Workshop*, pages 18–24, 1999.

[39] J. Eggink and G. J. Brown. A missing feature approach to instrument identification in polyphonic music. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume V, pages 553–556, 2003.

[40] J. Eggink and G. J. Brown. Application of missing feature theory to the recognition of musical instruments in polyphonic audio. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 2003.

[41] S. Essid, G. Richard, and B. David. Instrument recognition in polyphonic music based on automatic taxonomies. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):68–80, 2006.

[42] E. Vincent and X. Rodet. Instrument identification in solo and ensemble music using independent subspace analysis. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 576–581, 2004.

[43] H. Katayose and S. Inokuchi. An intelligence transcription system. In *Proceedings of the International Conference on Music Perception and Cognition (ICMPC)*, pages 95–98, 1989.

[44] A. Klapuri. Sound onset detection by applying psychoacoustic knowledge. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1999.

[45] T. Virtanen and A. Klapuri. Separation of harmonic sound sources using multipitch analysis and iterative parameter estimation. In *Proceedings of the Workshop on Applications of Signal Processing to Audio and Acoustics*, 2001.

[46] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka. Application of the Bayesian probability network to music scene analysis. In D. F. Rosenthal and H. G. Okuno, editors, *Computational Auditory Scene Analysis*, pages 115–137. Lawrence Erlbaum Associates, 1998.

[47] D. F. Rosenthal and H. G. Okuno, editors. *Computational Auditory Scene Analysis*. Lawrence Erlbaum Associates, 1998.

[48] J. Yin, T. Sim, Y. Wang, and A. Shenoy. Music transcription using an instrument model. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume III, pages 217–220, 2005.

[49] M.S. Puckette, T. Apel, and D. D. Zicarelli. Real-time audio analysis tools for Pd and MSP. In *Proceedings of the International Computer Music Conference (ICMC)*, 1998.

[50] G. Peeters and X. Rodet. Automatically selecting signal descriptors for sound classification. In *Proceedings of the International Computer Music Conference (ICMC)*, 2002.

[51] T. Hastie and R. Tibshirani. Classification by pairwise coupling. Technical report, Stanford Univ./Univ. Tronto, 1996.

[52] P. Herrera-Boyer, G. Peeters, and S. Dubnov. Automatic classification of musical instrument sounds. *Journal of New Music Research*, 32(1):3–21, 2003.

[53] P. Herrera-Boyer, A. Klapuri, and M. Davy. Automatic classification of pitched instrument sounds. In A. Klapuri and M. Davy, editors, *Signal Processing Methods for Music Transcription*. Springer, 2006.

[54] A. S. Bregman. *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, 1990.

[55] D. Marr. *Vision*. Freeman Pub., 1982.

[56] H. Nanjo and T. Kawahara. Language model and speaking rate adaptation for spontaneous presentation speech recognition. *IEEE Trans. Speech, Audio Process.*, 12(4), 2004.

[57] M. Nishida and T. Kawahara. Speaker model selection based on the bayesian information criterion applied to unsupervised speaker indexing. *IEEE Trans. Speech, Audio Peocess*, 13(4):583–592, 2004.

[58] Y. Mori, H. Saruwatari, T. Takahashi, S. Ukai, K. Shikano, T. Hiekata, Y. Ikeda, H. Hashimoto, and T. Morita. Blind separation of acoustic signals combining simomobel-based independent component analysis and binary masking. *EURASIP J. Applied Signal Process.*, 2006:Article ID 34970, 1–17, 2006.

[59] R. Mukai, H. Sawada, S. Araki, and S. Makino. Frequency-domain blind source separation of many speech signals using near-field and far-field models. *EURASIP J. Applied Signal Process.*, 2006:Article ID 83683, 1–13, 2006.

[60] M. Cooke, P. Green, L. Josifovski, and A. Vizinho. Robust automatic speech recognition with missing and unreliable acoustic data. *Speech Comm.*, 34:267–285, 2001.

[61] S. Yamamoto, J.-M. Valin, K. Nakadai, T. Ogata, and H. G. Okuno. Enhanced robot speech recognition based on microphone array source separation and missing feature theory. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2005)*, pages 1489–1494, 2005.

[62] Y. Gong. Speech recognition in noisy environments: A survey. *Speech Comm.*, 16:261–291, 1995.

[63] F. I. Klassner. *Data Reprocessing in Signal Understanding Systems*. PhD thesis, University of Massachusetts Amherst, 1996.

[64] M. Cowling and R. Sitte. Comparison of techniques for environmental sound recognition. *Pattern Recognition Letters*, 24:2895–2907, 2003.

[65] S. Namba. Definition of timbre. *J. Acoust. Soc. Jpn.*, 49(11):823–831, 1993. (in Japanese).

[66] L. Wedin and G. Goude. Dimension analysis of the perception of instrument timbre. *Scand. J. Psychol.*, 13:228–240, 1972.

[67] G. Bismark. Timbre of steady sounds: A factorial investigation of its verbal attributes. *Acoustica*, 30:146–159, 1974.

[68] J. M. Grey. Multidimensional perceptual scaling of musical timbres. *J. Acoust. Soc. Am.*, 61(5):1270–1277, 1977.

[69] J. M. Grey. Timber discrimination in musical patterns. *J. Acoust. Soc. Am.*, 64(2):467–472, 1978.

[70] J. R. Miller and E. C. Carterette. Perceptual space for musical structures. *J. Acoust. Soc. Am.*, 58(3):711–720, 1975.

[71] D. L. Wessel. Timbre space as a musical control structure. *Computer Music Journal*, 3(2):45–52, 1979.

[72] P. Iverson and C. Krumhansl. Isolating the dynamic attributes of musical timbre. *J. Acoust. Soc. Am.*, 94(5), 1993.

[73] P. Toiviainen, K. Kaipainen, and J. Louhivuori. Musical timbre: Similarity ratings correlate with computational feature space distances. *J. New Music Res.*, 24:292–298, 1995.

[74] S. McAdams, S. Winsberg, S. Donnadieu, G. D. Soete, and J. Krimphoff. Perceptual scaling of synthesized musical timbres: Common dimensions, specificities, latent subject classes. *Psychol. Res.*, 58:177–192, 1995.

[75] J. Marozeau, A. Cheveigne, S. McAdams, and S. Winsberg. The dependency of timbre on fundamental frequency. *J. Acoust. Soc. Am.*, 114(5):2946–2957, 2003.

[76] T. Sonoda, M. Goto, and Y. Muraoka. A WWW-based melody retrieval system. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 349–352, 1998.

[77] K. Kashino, T. Kurozumi, and H. Murase. A quick search method for audio and videl signals based on histogram pruning. *IEEE Transactions on Multimedia*, 5:348–357, 2003.

[78] T. Kumamoto. Design and evaluation of a music retrieval scheme that adapts to the user's impressions. In *Proceedings of the International Conference on User Modeling (UM '05)*, Lecture Notes in Artificial Intelligence, LNAI 3538, pages 287–296. Springer, 2005.

[79] J.-J. Aucouturier and F. Pachet. Music similarity measure: What's the use? In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 157–163, 2002.

[80] J. Paulus and A. Klapuri. Measuring the similarity of rhythmic patterns. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 150–156, 2002.

[81] E. Pampalk. A MATLAB toolbox to compute music similarity from audio. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 254–257, 2004.

[82] M. Casey and M. Slaney. The importance of sequences in musical similarity. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume V, pages 5–8, 2006.

[83] A. Rauber, E. Pampalk, and D. Merkl. Using psycho-acoustic models and self-organizing maps to create a hierarchical structuring of music by sound similarity. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 71–80, 2002.

[84] A. Berenzweig, B. Logan, D. P. W. Ellis, and B. Whiman. A large-scale evaluation of acoustic and subjective music similarity measure. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 2003.

[85] J.-J. Aucouturier and F. Pachet. Tools and architecture for the evaluation of similarity measures: Case study of timbre similarity. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 198–203, 2004.

[86] J.-J. Aucouturier and F. Pachet. Improving timbre similarity: How high's the sky? *Journal of Negative Results in Speech and Audio Sciences*, 2004.

[87] E. Pampalk. *Computational Models of Music Similarity and their Application in Music Information Retrieval*. PhD thesis, Technischen Universitat Wien, 2006.

[88] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.

[89] J.-J. Aucouturier and F. Pachet. Representing musical genres: A state of the art. *Journal of New Music Research*, 32(1):83–93, 2003.

[90] B. S. Manjunath, P. Salembier, and T. Sikora. *Introduction of MPEG-7*. Jorn Wiley & Sons Ltd., 2002.

[91] E. Gomez, F. Gouyon, P. Herrera, and X. Amatriain. Using and enhancing the current mpeg-7 standard for a music content processing tool. In *Audio Engineering Society 114th Convention*, 2003.

[92] M. Good. MusicXML: An internet-friendly format for sheet music. In *The XML 2001 Conf. Proc.*, 2001.

[93] P. Bellini and P. Nesi. WEDELMUSIC format: An XML music notation format for emerging applications. In *Proc. Int'l Conf. WEB Delivering of Music*, pages 79–86, 2001.

[94] H. Vinet, P. Herrera, and F. Pachet. The CUIDADO project. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 2002.

[95] F. Pachet, A. Laburthe, and J.-J. Aucouturier. The Cuidado music browser: An end-to-end electronic music distribution system. In *Proceedings of the 3rd International Workshop on Content-based Multimedia Indexing (CBMI '03)*, 2003.

[96] M. Goto. Music scene description project: Toward audio-based real-time music understanding. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 231–232, 2003.

[97] P. Herrera, J. Bello, G. Widmer, M. Sandler, O. Celma, F. Vignoli, E. Pampalk, P. Cano, S. Pauws, and X. Serra. SIMAC: Semantic interaction with music audio contents. In *Proceedings of the 2nd European Workshop on the Integration of Knowledge, Semantic, and Digital Media Technologies*, 2005.

[98] J.-J. Aucouturier and F. Pachet. Scaling up music playlist generation. In *Proceedings of the International Conference on Multimedia & Expo*, 2002.

[99] K. Yoshii, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno. Hybrid collaborative and content-based music recommendation using probabillistic model with latent user preferences. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 296–301, 2006.

[100] F. Pachet. Content management for electronic music distribution. *Communications of the ACM*, 46(4):71–75, 2003.

[101] L. Lu, D. Liu, and H.-J. Zhang. Automatic mood detection and tracking of music audio signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1), 2006.

[102] A. Savitzky and M. J. E. Golay. Smoothing and differentiation of data by simplified least squares procudures. *Anal. Chem.*, 36(8):1627–1639, 1964.

[103] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Music genre database and musical instrument sound database. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 229–230, 2003.

[104] K. W. Berger. Some factors in the recognition of timbre. *J. Acoust. Soc. Am.*, 36(10):1888–1891, 1964.

[105] H. F. Olson. *Music, Physics, and Engineering*. Dover Publications, 1966.

[106] M. Casey. General sound classification and similarity in mpeg-7. *Organaized Sound*, 6(2), 2002.

[107] S. Dubnov and N. Tishby. Clustering of musical sounds using polyspectral distance measures. In *Proceedings of the International Computer Music Conference (ICMC)*, 1995.

[108] F. Nack and L. Hardman. Towards a syntax for multimedia semantics. In *CWI Reports of INS 2 (Multimedia and Human-Computer Interaction)*, INS-R0204, 2002.

[109] Y. Sakuraba, T. Kitahara, and H. G. Okuno. Comparing features for forming music streams in automatic music transcription. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume IV, pages 273–276, 2004.

[110] D. Huron. Tone and voice: A derivation of the rules of voice-leading from perceptual principles. *Music Perception*, 19(1):1–64, 2001.

[111] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Popular, classical, and jazz music databases. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 287–288, 2002.

[112] J. Eggink and G. J. Brown. Extracting melody lines from complex audio. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 84–91, 2004.

[113] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77(2):257–286, 1989.

[114] T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno. Instrument identification in polyphonic music: Feature weighting with mixed sounds, pitch-dependent timbre modeling, and use of musical context. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 558–563, 2005.

[115] M. Goto. A robust predominant-f0 estimation method for real-time detection of melody and bass lines in cd recordings. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume II, pages 757–760, 2000.

[116] Y. Sakuraba and H. G. Okuno. Note recognition of polyphonic music by using timbre similarity and direction proximity. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 167–170, 2003.

[117] R. Hiraga. A look of performance. In *IEEE Visualization*, 2002.

[118] R. Hiraga, R. Miyazaki, and I. Fujishiro. Performance visualization—a new challenge to music through visualization. In *ACM Multimedia*, 2002.

[119] M. Goto. A chorus section detection method for musical audio signals and its application to a music listening system. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(5):1783–1794, 2006.

[120] G. Tzanetakis. *Manipulation, Analysis and Retrieval Systems for Audio Signals*. PhD thesis, Princeton University, 2002.

[121] T. Yoshioka, T. Kitahara, K. Komatani, T. Ogata, and H. G. Okuno. Automatic chord transcription with concurrent recognition of chord symbols and boundaries. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR 2004)*, pages 100–105, 2004.

[122] H. Fujihara, T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno. Singer identification based on accompaniment sound reduction and reliable frame selection. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, pages 329–336, 2005.

[123] A. Taguchi, T. Kitahara, K. Ishihara, K. Komatani, T. Ogata, and H. G. Okuno. Design of XML tagset for environmental sounds based on sound-imitation words and its automatic annotation, 2006. (in Japanese).

Relevant Publications

Chapter 3

1. Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Musical Instrument Identification Considering Pitch-dependent Characteristics of Timbre: A Classifier based on F0-dependent Multivariate Normal Distribution", *IPSJ Journal*, Vol.44, No.10, pp.2448–2458, October 2003 *(in Japanese)*.

2. Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Pitch-dependent Identification of Musical Instrument Sounds", *Applied Intelligence*, Vol.23, No.3, pp.267–275, December 2005.

3. Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Musical Instrument Identification based on F0-dependent Multivariate Normal Distribution", *Proceedings of the 2003 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2003)*, Vol.V, pp.421–424, April 2003. *(Cancelled because of SARS)*

4. Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Musical Instrument Identification based on F0-dependent Multivariate Normal Distribution", *Proceedings of the 2003 IEEE International Conference on Multimedia & Expo (ICME 2003)*, Vol.III, pp.409–412, June 2003. *(Reprint of the paper published in ICASSP 2003 (cancelled))*

5. Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Pitch-dependent Musical Instrument Identification and Its Application to Musical Sound Ontology", In P. W. H. Chung, C. Hinde and M. Ali (Eds.) *Developments in Applied Artificial Intelligence, Lecture Notes in Artificial Intelligence 2718, Proceedings of the 16th International Conference on Industrial Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE-2003)*, pp.112–122, Springer, June 2003.

Chapter 4

1. Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Acoustic-feature-based Musical Instrument Hierarchy and Its Application to Category-level Recognition of Unknown Musical Instruments", *IPSJ Journal*, Vol.45, No.3, pp.680–689, March 2004 *(in Japanese)*.

2. Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Acoustical-similarity-based Musical Instrument Hierarchy and Its Application to Musical Instrument Identification", *Proceedings of International Symposium on Musical Acoustics (ISMA 2004)*, 3-S2-12, pp.297–300, April 2004.

3. Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Category-level Identification of Non-registered Musical Instrument Sounds", *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2004)*, Vol.IV, pp.253–256, May 2004.

Chapter 5

1. Tetsuro Kitahara, Masataka Goto, Kazunori Kamatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrument Identification in Polyphonic Music: Feature Weighting based on Mixed-sound Template and Use of Musical Context", *IEICE Transactions on Information and Systems*, Vol.J89-D, No.12, pp.2721–2733, December 2006. *(in Japanese)*.

2. Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrument Identification in Polyphonic Music: Feature Weighting to Minimize Influence of Sound Overlaps", *EURASIP Journal on Applied Signal Processing*, Vol.2007, Article ID 51979, 15 pages, 2007.

3. Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata and Hiroshi G Okuno: "Instrument Identification in Polyphonic Music: Feature Weighting with Mixed Sounds, Pitch-dependent Timbre Modeling, and Use of Musical Context", *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, pp.558–563, Septermber 2005.

Chapters 6 and 7

1. Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrogram: Probabilistic Representation of Instrument Existence for Polyphonic Music", *IPSJ Journal*, Vol.48, No.1, pp.214–226 (also published in *IPSJ Digital Courier*, Vol.3, pp.1–13), Janurary 2007.

2. Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata and Hiroshi G. Okuno: "Instrogram: A New Musical Instrument Recognition Technique without Using Onset Detection nor F0 Estimation", *Proceedings of the 2006 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, Vol.V, pp.229–232, May 2006.

3. Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Musical Instrument Recognizer "Instrogram" and Its Application to Music Retrieval based on Instrumentation Similarity", *Proceedings of the 8th IEEE International Symoposium on Multimedia (ISM 2006)*, pp.265–272, December 2006.

List of All Publications by the Author

Major Publications

Journal Papers

1) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Musical Instrument Identification Considering Pitch-dependent Characteristics of Timbre: A Classifier based on F0-dependent Multivariate Normal Distribution", *IPSJ Journal*, Vol.44, No.10, pp.2448–2458, October 2003 *(in Japanese)*.

2) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Acoustic-feature-based Musical Instrument Hierarchy and Its Application to Category-level Recognition of Unknown Musical Instruments", *IPSJ Journal*, Vol.45, No.3, pp.680–689, March 2004 *(in Japanese)*.

3) Katsuhisa Ishida, Tetsuro Kitahara, and Masayuki Takeda: "N-gram based Melody Correction for Improvisation", *IPSJ Journal (Technical Note)*, Vol.45, No.3, pp.743–746, March 2004 *(in Japanese)*.

4) Katsuhisa Ishida, Tetsuro Kitahara, and Takayuki Takeda: "Improvisation Supporting System Using N-gram-based Melody Appropriateness Determination", *IPSJ Journal*, Vol.46, No.7, pp.1548–1559, July 2005 *(in Japanese)*.

5) Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Pitch-dependent Identification of Musical Instrument Sounds", *Applied Intelligence*, Vol.23, No.3, pp.267–275, December 2005.

6) Hiromasa Fujihara, Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "A Singer Identification Method for Muscial Pieces

on the Basis of Acoompaniment Sound Reduction and Reliable Frame Selection", *IPSJ Journal*, Vol.47, No.6, pp.1831–1843, June 2006 *(in Japanese)*.

7) Tetsuro Kitahara, Masataka Goto, Kazunori Kamatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrument Identification in Polyphonic Music: Feature Weighting based on Mixed-sound Template and Use of Musical Context", *IEICE Transactions on Information and Systems*, Vol.J89-D, No.12, pp.2721–2733, December 2006. *(in Japanese)*.

8) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrument Identification in Polyphonic Music: Feature Weighting to Minimize Influence of Sound Overlaps", *EURASIP Journal on Applied Signal Processing*, Vol.2007, Article ID 51979, 15 pages, 2007.

9) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrogram: Probabilistic Representation of Instrument Existence for Polyphonic Music", *IPSJ Journal*, Vol.48, No.1, pp.214–226 (also published in *IPSJ Digital Courier*, Vol.3, pp.1–13), Janurary 2007.

International Conference Papers

10) Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Musical Instrument Identification based on F0-dependent Multivariate Normal Distribution", *Proceedings of the 2003 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2003)*, Vol.V, pp.421–424, April 2003. *(Cancelled because of SARS)*

11) Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Musical Instrument Identification based on F0-dependent Multivariate Normal Distribution", *Proceedings of the 2003 IEEE International Conference on Multimedia & Expo (ICME 2003)*, Vol.III, pp.409–412, June 2003. *(Reprint of the paper published in ICASSP 2003 (cancelled))*

12) Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Acoustical-similarity-based Musical Instrument Hierarchy and Its Application to Musical Instrument Identification", *Proceedings of International Symposium on Musical Acoustics (ISMA*

2004), 3-S2-12, pp.297–300, April 2004.

13) Tetsuro Kitahara, Masataka Goto and Hiroshi G. Okuno: "Category-level Identification of Non-registered Musical Instrument Sounds", *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2004)*, Vol.IV, pp.253–256, May 2004.

14) Yohei Sakuraba, Tetsuro Kitahara and Hiroshi G. Okuno: "Comparing Features for Forming Music Streams in Automatic Music Transcription", *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2004)*, Vol.IV, pp.273–276, May 2004.

15) Katsuhisa Ishida, Tetsuro Kitahara and Masayuki Takeda: "ism: Improvisation Supporting System based on Melody Correction", *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 04)*, pp.177–180, June 2004.

16) Takuya Yoshioka, Tetsuro Kitahara, Kazunori Komatani, Tetsuya Ogata and Hiroshi G. Okuno: "Automatic Chord Transcription with Concurrent Recognition of Chord Symbols and Boundaries", *Proceedings of the International Conference on Music Information Retrieval (ISMIR 2004)*, pp.100–105, October 2004.

17) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata and Hiroshi G Okuno: "Instrument Identification in Polyphonic Music: Feature Weighting with Mixed Sounds, Pitch-dependent Timbre Modeling, and Use of Musical Context", *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, pp.558–563, Septermber 2005.

18) Hiromasa Fujihara, Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata and Hiroshi G. Okuno: "Singer Identification based on Accompaniment Sound Reduction and Reliable Frame Selection", *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, pp.329–336, September 2005.

19) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata and Hiroshi G. Okuno: "Instrogram: A New Musical Instrument Recognition Technique with-

out Using Onset Detection nor F0 Estimation", *Proceedings of the 2006 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, Vol.V, pp.229–232, May 2006.

20) Hiromasa Fujihara, Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata and Hiroshi G. Okuno: "F0 Estimation Method for Singing Voice in Polyphonic Audio Signal based on Statistical Vocal Model and Viterbi Search", *Proceedings of the 2006 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, Vol.V, pp.253–256, May 2006.

21) Hiromasa Fujihara, Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Speaker Identification under Noisy Environments by using Harmonic Structure Extraction and Reliable Frame Weighting", *Proceedings of the International Conference on Spoken Language Processing (ICSLP 2006)*, pp.1459–1462, September 2006.

22) Katsutoshi Itoyama, Tetsuro Kitahara, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Automatic Feature Weighting in Automatic Transcription of Specified Part in Polyphonic Music", *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*, pp.172–175, October 2006.

23) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Musical Instrument Recognizer "Instrogram" and Its Application to Music Retrieval based on Instrumentation Similarity", *Proceedings of the 8th IEEE International Symoposium on Multimedia (ISM 2006)*, pp.265–272, December 2006.

Book Chapters

24) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Pitch-dependent Musical Instrument Identification and Its Application to Musical Sound Ontology", In P. W. H. Chung, C. Hinde and M. Ali (Eds.) *Developments in Applied Artificial Intelligence, Lecture Notes in Artificial Intelligence 2718, Proceedings of the 16th International Conference on Industrial Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE-2003)*, pp.112–122, Springer, June 2003.

25) Tetsuro Kitahara, Katsuhisa Ishida, and Masayuki Takeda: "ism: Improvisation

Supporting Systems with Melody Correction and Key Vibration", In F. Kishino, Y. Kitamura, H. Kato and N. Nagata (Eds.) *Entertainment Computing, Lecture Notes in Computer Science 3711, Proceedings of the 4th International Conference on Entertainment Computing (ICEC 2005)*, pp.315–327, Springer, September 2005.

Other Publications (All in Japanese)

Refereed Domestic Conference Papers

26) Katsutoshi Ishida, Tetsuro Kitahara, and Masayuki Takeda: "ism: An Improvisation Supporting System based on Correcting Unnatural Melodies", *Proceedings of the 11th Workshop on Interactive Systems and Software (WISS 2003)*, pp.19–24, December 2003.

27) Katsutoshi Ishida, Tetsuro Kitahara, and Masayuki Takeda: "A Musical Keyboard that Presents Musical Information to the Player with Vibration", *Proceedings of the 12th Workshop on Interactive Systems and Software (WISS 2004)*, pp.59–64, December 2004.

28) Yuu Misawa, Yutaka Hosono, Akifumi Nishina, Katsuhisa Ishida, Tetsuro Kitahara, Masataka Goto, and Masayuki Takeda: "Openism: An Open-to-the-public Distributed Session System with a Melody-correction-based Improvisation Support Function", *Proceedings of the 13th Workshop on Interactive Systems and Software (WISS 2005)*, pp.87–92, December 2005.

Technical Reports

29) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Musical Instrument Identification Considering Pitch-dependent Characteristics of Timbre", *IPSJ SIG Notes*, 2001-MUS-40-2, Vol.2001, No.45, pp.7–14, May 2001.

30) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Musical Instrument Identification: A Classifier Considering Pitch-dependent Characteristics of Timbre", *IPSJ SIG Notes*, 2002-MUS-46-1, Vol.2002, No.63, pp.1–8, July 2002.

31) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Acoustic-feature-based Musical Instrument Hierarchy and Its Application to Category-level Musical Instru-

ment Recognition", *IPSJ SIG Notes*, 2003-MUS-51-9, Vol.2003, No.82, pp.51–58, August 2003.

32) Kazuyoshi Yoshii, Tetsuro Kitahara, Yohei Sakuraba, and Hiroshi G. Okuno: "Automatic Transcription of Drum Patterns Using Unsupervised Clustering by Self-Organizing Map", *IPSJ SIG Notes*, 2003-MUS-51-8, Vol.2003, No.82, pp.43–50, August 2003.

33) Masataka Goto, Keiji Hirata, Haruhiro Katayose, Shigeyuki Hirai, Masatoshi Hamanaka, Haruto Takeda, and Tetsuro Kitahara: "Panel Discussion: What Researchers in Music Information Processing {Are Expected To Do, Hope For}", *IPSJ SIG Notes*, 2003-MUS-51-5, Vol.2003, No.82, pp.25–28, August 2003.

34) Katsuhisa Ishida, Tetsuro Kitahara, and Masayuki Takeda: "ism: An Improvisation Supporting System based on Realtime Melody Correction", *IPSJ SIG Notes*, 2003-HI-106-2, 2003-MUS-52-2, Vol.2003, No.111, pp.9–15, November 2003.

35) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "TimbreTree: Hierarchical Classification of Musical Instruments based on their Timbres", *Transactions of the Technical Committee on Musical Acoustics, the Acoustical Society of Japan*, MA2004-7, Vol.23, No.2, pp.13–18, June 2004.

36) Takuya Yoshioka, Tetsuro Kitahara, Tetsuya Ogata, and Hiroshi G. Okuno: "Automatic Chord Transcription for Musical Audio Signals", *Transactions of the Technical Committee on Musical Acoustics, the Acoustical Society of Japan*, MA2004-8, Vol.23, No.2, pp.19-24, June 2004.

37) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Musical Instrument Identification Using a Mixed-sound Template", *IPSJ SIG Technical Report*, 2004-MUS-56-9, Vol.2004, No.84, pp.57–64, August 2004.

38) Takuya Yoshioka, Tetsuro Kitahara, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Automatic Chord Recognition Considering Mutual Dependency of Chord-Boundary Detection and Chord-Symbol Identification", *IPSJ SIG Technical Report*, 2004-MUS-56-6, Vol.2004, No.84, pp.33–40, August 2004.

39) Masatoshi Hamanaka, Tetsuro Kitahara, Katsuhisa Ishida, Akio Yatui, Yoshinari Takegawa, Kazuyoshi Yoshii, Homei Miyashita, and Kentaro Ueda: "Demonstrations: Introdution of Research by Young Researchers", *IPSJ SIG Technical Report*, 2004-MUS-56-6, Vol.2004, No.84, pp.27–32, August 2004.

40) Tetsuro Kitahara, Katsuhisa Ishida, Masayuki Takeda: "An Improvisation Supporting System using a Vibrating Keyboard 'Buru-Buru-kun'", *IPSJ SIG Technical Report*, 2005-MUS-60-5, Vol.2005, No.45, pp.25-30, May 2005.

41) Masatoshi Hamanaka, SeungHee Lee, Yuya Iketsuki, Kazushi Ishihara, Kenzi Noike, Tomoyasu Nakano, Katsuhiko Kaji, Yoshinori Oka, Kenji Hirata, Shu Matsuda, Shinobu Aoki, and Kentaro Ueda: "Demonstrations: Introdution of Research by Young Researchers II", *IPSJ SIG Technical Report*, 2005-MUS-61-5, Vol.2005, No.82, pp.27-33, August 2005.

42) Hiromasa Fujihara, Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "A Singer Identification Method for Singing Voices in Muscial Pieces on the Basis of Accompaniment Sound Reduction and Reliable Frame Selection", *IPSJ SIG Technical Report*, 2005-MUS-61-16, Vol.2005, No.82, pp.97-104, August 2005.

43) Tetsuro Kitahara, Masataka Goto, Kazunori Kamatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrogram: Musical Instrument Recognition Method Without Requiring Onset Detection Nor F0 Estimation", *IPSJ SIG Technical Report*, 2006-MUS-66-12, Vol.2006, No.90, pp.69–76, August 2006.

44) Masatoshi Hamanaka, Yoshinari Takegawa, Tomoko Hashida, Yoichi Motokawa, Tetsuaki Baba, Key Higurashi, Tomoyasu Nakano, Kazuyoshi Yoshii, Masaki Matsubara, Katsuhiko Kaji, and Tetsuro Kitahara: "Demonstrations: Introduction of Research by Young Researchers III", *IPSJ SIG Technical Report*, 2006-MUS-66-10, Vol.2006, No.90, pp.55–61, August 2006.

45) Masatoshi Hamanaka, Yoshinari Takegawa, Kenichi Iwai, Naoya Takahashi, Tomoyasu Nakano, Yasunori Ohishi, Katsutoshi Itoyama, Tetsuro Kitahara, and Kazuyoshi Yoshii: "Demonstrations: Introduction of Research by Young Researchers IV", *IPSJ SIG Technical Report*, 2006-MUS-67-3, Vol.2006, No.113, pp.9–14, October 2006.

National Convention Papers

46) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Feature Extraction of Musical Instrument Sounds for Constructing Musical Sound Ontology", *Proceedings of the 62th IPSJ National Convention*, 4M-5, March 2001.

47) Takahiro Yanagawa, Tetsuro Kitahara, and Masayuki Takeda: "A Pitch Correcting System for Improvisation", *Proceedings of the 64th IPSJ National Convention*, 1L-5, March 2002.

48) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Musical Instrument Identification Considering Pitch-dependent Characteristics of Timbre Space", *Proceedings of the 2002 Autumn Meeting of the Acoustical Society of Japan*, 1-1-4, pp.643–644, September 2002.

49) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Hirarchical Clustering of Musical Instrument Sounds", *Proceedings of the 65th IPSJ National Convention*, 1P-1, March 2003.

50) Kazuyoshi Yoshii, Tetsuro Kitahara, Yohei Sakuraba, and Hiroshi G. Okuno: "Percussive Instrument Identification Using Unsupervised Clustering and Recognition Error Patterns", *Proceedings of the 65th IPSJ National Convention*, 1P-3, March 2003.

51) Katsuhisa Ishida, Tetsuro Kitahara, Takahiro Yanagawa, and Masayuki Takeda: "Statistics-based Performance Correction for Improvisation", *Proceedings of the 65th IPSJ National Convention*, 2P-3, March 2003.

52) Tetsuro Kitahara, Masataka Goto, and Hiroshi G. Okuno: "Musical Instrument Identification Considering Non-registered Instruments", *Proceedings of the 66th IPSJ National Convention*, 3ZA-3, March 2004.

53) Takuya Yoshioka, Kazuyoshi Yoshii, Tetsuro Kitahara, Yohei Sakuraba, Tetsuya Ogata, and Hiroshi G. Okuno: "Concurrent Recognition of Chord Changes and Chord Symbols for Musical Audio Signals", *Proceedings of the 66th IPSJ National Convention*, 3ZA-4, March 2004.

54) Katsuhisa Ishida, Tetsuro Kitahara, and Masayuki Takeda: "Improvisatoin Support using Statistical-model-based Melody Appropriateness Determination", *Proceedings of the 2004 Autumn Meeting of the Acoustical Society of Japan*, 2-6-8, pp.783–784, September 2004.

55) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Feature Template Construction from Sound Mixtures for Instrument Identification in Polyphonic Music", *Proceedings of the 67th IPSJ National Convention*, 3G-4, March 2005.

56) Hiromasa Fujihara, Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Singer Identification Using the Harmonic Structure of Singer's Voice", *Proceedings of the 67th IPSJ National Convention*, 3R-8, March 2005.

57) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrument Identification in Polyphonic Music by Constructing Feature Vector Template from Polyphonic Sounds", *Proceedings of the 2005 Autumn Meeting of the Acoustical Society of Japan*, 3-10-15, Septermber 2005.

58) Satoshi Kaijiri, Kazushi Ishihara, Tetsuro Kitahara, Jean-Marc Valin, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Environmental Sound Recognition under Noises for Understanding Situation by Robots", *Proceedings of the 6th SICE System Integration Division Annual Conference (SI2005)*, pp.65–66, December 2005.

59) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Instrogram: Music Visual Representation based on Instrument Existence Probability", *Proceedings of the 2006 Spring Meeting of the Acoustical Society of Japan*, 2-2-13, March 2006.

60) Katsutoshi Itoyama, Tetsuro Kitahara, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Automatic Feature Weighting in Automatic Transcription of Specified Part in Polyphonic Music", *Proceedings of the 68th IPSJ National Convention*, 2L-6, March 2006.

61) Masahiro Nishiyama, Tetsuro Kitahara, Kazunori Komatani, Tetsuya Ogata, and

Hiroshi G. Okuno: "Hierarchical Narrative Tag Design for Program Music Annotation", *Proceedings of the 68th IPSJ National Convention*, 3L-6, March 2006.

62) Akihiro Taguchi, Tetsuro Kitahara, Kazushi Ishihara, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Design of XML Tagset for Environmental Sounds based on Sound-imitation Words and Its Automatic Annotation", *Proceedings of the 68th IPSJ National Convention*, 3L-7, March 2006.

63) Hiromasa Fujihara, Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Speaker Identification under Noisy Environments based on Harmonic Structure Extraction and Reliable Frame Selection", *Proceedings of the 2006 Spring Meeting of the Acoustical Society of Japan*, 1-11-17, March 2006.

64) Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno: "Music Information Retrieval based on Instrumentation Similarity Using Instrogram", *Proceedings of the 2006 Autumn Meeting of the Acoustical Society of Japan*, 2-7-1, September 2006.