A Chorus Section Detection Method for Musical Audio Signals and Its Application to a Music Listening Station

Masataka Goto

***Abstract*—This paper describes a method for obtaining a list of repeated chorus ("hook") sections in compact-disc recordings of popular music. The detection of chorus sections is essential for the computational modeling of music understanding and is useful in various applications, such as automatic chorus-preview/search functions in music listening stations, music browsers, or music retrieval systems. Most previous methods detected as a chorus a repeated section of a given length and had difficulty identifying both ends of a chorus section and dealing with modulations (key changes). By analyzing relationships between various repeated sections, our method, called *RefraiD*, can detect all the chorus sections in a song and estimate both ends of each section. It can also detect modulated chorus sections by introducing a perceptually motivated acoustic feature and a similarity that enable detection of a repeated chorus section even after modulation. Experimental results with a popular music database showed that this method correctly detected the chorus sections in 80 of 100 songs. This paper also describes an application of our method, a new music-playback interface for trial listening called *SmartMusicKIOSK*, which enables a listener to directly jump to and listen to the chorus section while viewing a graphical overview of the entire song structure. The results of implementing this application have demonstrated its usefulness.**

***Index Terms*—Chorus detection, chroma vector, music-playback interface, music structure, music understanding.**

I. Introduction

CHORUS ("hook" or refrain) sections of popular music are the most representative, uplifting, and prominent thematic sections in the music structure of a song, and human listeners can easily understand where the chorus sections are because these sections are the most repeated and memorable portions of a song. Automatic detection of chorus sections is essential for building a music-scene-description system [1], [2] that can understand musical audio signals in a human-like fashion, and is useful in various practical applications. In music browsers or music retrieval systems, it enables a listener to quickly preview a chorus section as an "audio thumbnail" to find a desired song. It can also increase the efficiency and precision of music retrieval systems by enabling them to match a query with only the chorus sections.

Manuscript received January 31, 2005; revised October 10, 2005. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Malcom Slaney.

The author is with the Information Technology Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Ibaraki 305-8568, Japan (e-mail: m.goto@aist.go.jp).

Digital Object Identifier 10.1109/TSA.2005.863204

This paper describes a method, called Refrain Detecting Method (*RefraiD*), that exhaustively detects all repeated chorus sections appearing in a song with a focus on popular music. It can obtain a list of the beginning and end points of every chorus section in real-world audio signals and can detect modulated chorus sections. Furthermore, because it detects chorus sections by analyzing various repeated sections in a song, it can generate an intermediate-result list of repeated sections that usually reflect the song structure; for example, the repetition of a structure like verse A, verse B, and chorus is often found in the list.

This paper also describes a music listening station called *SmartMusicKIOSK* that was implemented as an application system of the RefraiD method. In music stores, customers typically search out the chorus or "hook" of a song by repeatedly pressing the fast-forward button, rather than passively listening to the music. This activity is not well supported by current technology. Our research has led to a function for jumping to the chorus section and other key parts (repeated sections) of a song, plus a function for visualizing the song structure. These functions eliminate the hassle of searching for the chorus and make it easier for a listener to find desired parts of a song, thereby facilitating an active listening experience.

The following sections introduce related research, describe the problems dealt with, explain the RefraiD method in detail, and show experimental results indicating that the method is robust enough to correctly detect the chorus sections in 80 of 100 songs of a popular-music database. Finally, the SmartMusicKIOSK system and its usefulness are described.

II. Related Work

Most previous chorus detection methods [3]–[5] only extract a single segment from several chorus sections by detecting a repeated section of a designated length as the most representative part of a song. Logan and Chu [3] developed a method using clustering techniques and hidden Markov models (HMMs) to categorize short segments (1 s) in terms of their acoustic features, where the most frequent category is then regarded as a chorus. Bartsch and Wakefield [4] developed a method that calculates the similarity between acoustic features of beat-length segments obtained by beat tracking and finds the given-length segment with the highest similarity averaged over its segment. Cooper and Foote [5] developed a method that calculates a similarity matrix of acoustic features of short frames (100 ms) and

finds the given-length segment with the highest similarity between it and the whole song. Note that these methods assume that the output segment length is given and do not identify both ends of a chorus section.

Music segmentation or structure discovery methods [6]–[13] where the output segment length is not assumed have also been studied. Dannenberg and Hu [6], [7] developed a structure discovery method of clustering pairs of similar segments obtained by several techniques such as efficient dynamic programming or iterative greedy algorithms. This method finds, groups, and removes similar pairs from the beginning to group all the pairs. Peeters *et al.* [8] and Peeters and Rodet [9] developed a supervised learning method of modeling dynamic features and studied two structure discovery approaches: the sequence approach of obtaining repetitions of patterns and the state approach of obtaining a succession of states. The dynamic features are selected from the spectrum of a filter-bank output by maximizing the mutual information between the selected features and hand-labeled music structures. Aucouturier and Sandler [14] developed two methods of finding repeated patterns in a succession of states (texture labels) obtained by HMMs. They used two image processing techniques, the kernel convolution and Hough transform, to detect line segments in the similarity matrix between the states. Foote and Cooper [10], [11] developed a method of segmenting music by correlating a kernel along the diagonal of the similarity matrix, and clustering the obtained segments on the basis of the self-similarity of their statistics. Chai and Vercoe [12] developed a method of detecting segment repetitions by using dynamic programming, clustering the obtained segments, and labeling the segments based on heuristic rules such as the rule of first labeling the most frequent segments, removing them, and repeating the labeling process. Wellhausen and Crysandt [13] studied the similarity matrix of spectral-envelope features defined in the MPEG-7 descriptors and a technique of detecting noncentral diagonal line segments.

None of these methods, however, address the problem of detecting all the chorus sections in a song. Furthermore, while chorus sections are sometimes modulated (the key is changed) during their repetition in a song, previously reported methods did not deal with modulated repetition.

III. Chorus Section Detection Problem

To enable the handling of a large number of songs in popular music, this research aims for a general and robust chorus section detection method using no prior information on acoustic features unique to choruses. To this end, we focus on the fact that chorus sections are usually the most repeated sections of a song and adopt the following basic strategy: find sections that repeat and output those that appear most often. It must be pointed out, however, that it is difficult for a computer to judge repetition because it is rare for repeated sections to be exactly the same. The following summarizes the main problems that must be addressed in this regard.

[Problem 1] Acoustic Features and Similarity: Whether a section is a repetition of another must be judged on the basis of the similarity between the acoustic features obtained from each section. In this process, the similarity must be high between acoustic features even if the accompaniment or melody line changes somewhat in the repeated section (e.g., the absence of accompaniment on bass and/or drums after repetition). This condition is difficult to satisfy if acoustic features are taken to be simple power spectrums or mel-frequency cepstral coefficients (MFCC) as used in audio/speech signal processing.

[Problem 2] Repetition Judgment Criterion: The criterion establishing how high similarity must be to indicate repetition depends on the song. For a song containing many repeated accompaniment phrases, for example, only a section with very high similarity should be considered the chorus section repetition. For a song containing a chorus section with accompaniments changed after repetition, on the other hand, a section with somewhat lower similarity can be considered the chorus section repetition. This criterion can be easily set for a small number of specific songs by manual means. For a large open song set, however, the criterion should be automatically modified based on the song being processed.

[Problem 3] Estimating Both Ends of Repeated Sections: Both ends (the beginning and end points) of repeated sections must be estimated by examining the mutual relationships among the various repeated sections. For example, given a song having the structure (A B C B C C), the long repetition corresponding to (B C) would be obtained by a simple repetition search. Both ends of the C section in (B C) could be inferred, however, from the information obtained regarding the final repetition of C in this structure.

[Problem 4] Detecting Modulated Repetition: Because the acoustic features of a section generally undergo a significant change after modulation (key change), similarity with the section before modulation is low, making it difficult to judge repetition. The detection of modulated repetition is important since modulation sometimes occurs in chorus repetitions, especially in the latter half of a song.[1]

IV. Chorus Section Detection Method: RefraiD

Fig. 1 shows the process flow of the RefraiD method. First, a 12-dimensional feature vector called a *chroma vector*, which is robust with respect to changes of accompaniments, is extracted from each frame of an input audio signal and then the similarity between these vectors is calculated (*solution to Problem 1*). Each element of the chroma vector corresponds to one of the 12 pitch classes (C, C#, D, D#, E, F, F#, G, G#, A, A#, and B) and is the sum of magnitude at frequencies of its pitch class over six octaves. Pairs of repeated sections are then listed (found) using an adaptive repetition-judgment criterion that is configured by an automatic threshold selection method based on a discriminant criterion [17] (*solution to Problem 2*). To organize common repeated sections into groups and to identify both ends of each section, the pairs of repeated sections are integrated (grouped) by analyzing their relationships over the whole song (*solution*

[1]Although a reviewer of this paper pointed out that songs with modulation are generally rare in Western popular music, they are not rare in Japanese popular music, which has been influenced by Western music. We conducted a survey on Japan's popular music hit chart (top 20 singles ranked weekly from fiscal 2000 to fiscal 2003) and found that modulation occurred in chorus repetitions in 152 songs (10.3%) out of 1481.

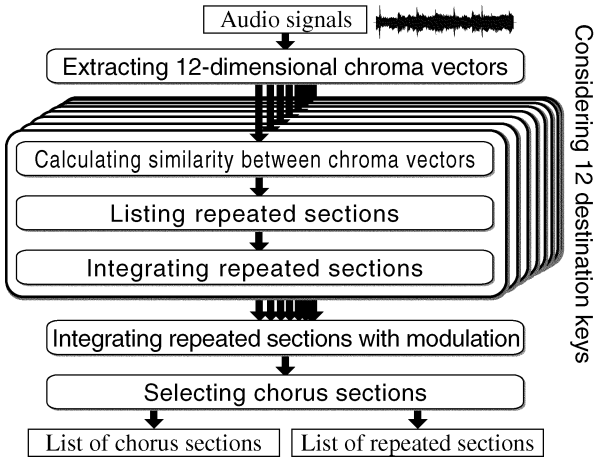

Fig. 1. Overview of chorus section detection method *RefraiD*.

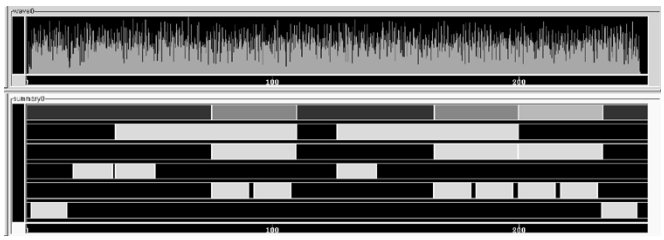

Fig. 2. Example of chorus sections and repeated sections detected by the RefraiD method. The horizontal axis is the time axis (in seconds) covering the entire song. The upper window shows the power. The top row in the lower window shows the list of the detected chorus sections, which were correct for this song (RWC-MDB-P-2001 no. 18 of the RWC Music Database [15], [16]) and the last of which was modulated. The bottom five rows show the list of various repeated sections (only the five longest repeated sections are shown).

to Problem 3). Because each element of a chroma vector corresponds to a different pitch class, a before-modulation chroma vector is close to the after-modulation chorus vector whose elements are shifted (exchanged) by the pitch difference of the key change. By considering 12 kinds of shift (pitch differences), 12 sets of the similarity between nonshifted and shifted chroma vectors are then calculated, pairs of repeated sections from those sets are listed, and all of them are integrated (*solution to Problem 4*). Finally, the *chorus measure*, which is the possibility of being chorus sections for each group, is evaluated, and the group of chorus sections with the highest chorus measure as well as other groups of repeated sections are output (Fig. 2).

The main symbols used in this section are listed in Table I.

TABLE I
LIST OF SYMBOLS

symbol	description
t	Time (the discrete time step is 80 ms)
$\Psi_p(f,t)$	Magnitude spectrum at log-scale frequency f
$\vec{v}(t)$	12-dimensional chroma vector
$v_c(t)$	Element of $\vec{v}(t)$ ($c = 1, 2, \ldots 12$)
$r(t,l)$	Similarity between chroma vectors $\vec{v}(t)$ and $\vec{v}(t-l)$
$[T1, T2]$	Section between times $T1$ and $T2$
$R_{all}(t,l)$	Possibility of containing line segments at lag l
Φ	Set of groups ϕ_i ($i = 1, 2, \ldots, N$)
N	Number of groups ϕ_i in set Φ
ϕ_i	Group of line segments having $[Ts_i, Te_i]$
Γ_i	Set of lags γ_{ij} of line segments in group ϕ_i ($j = 1, 2, \ldots, M_i$)
M_i	Number of line segments in group ϕ_i
γ_{ij}	Lag of j-th line segment in group ϕ_i
$R_{[Ts_i, Te_i]}(l)$	Possibility of containing line segments at lag l within $[Ts_i, Te_i]$
ζ	Pitch difference of key change (modulation) ($\zeta = 0, 1, \ldots, 11$)
S	12-by-12 shift matrix
$r_\zeta(t,l)$	Similarity between $S^\zeta\vec{v}(t)$ and $\vec{v}(t-l)$
$[Ps_{ij}, Pe_{ij}]$	Unfolded repeated section corresponding to γ_{ij}
λ_{ij}	Possibility that sections $[Ts_i, Te_i]$ and $[Ps_{ij}, Pe_{ij}]$ are repeated
ν_i	Chorus measure (possibility of being chorus sections for each group i)

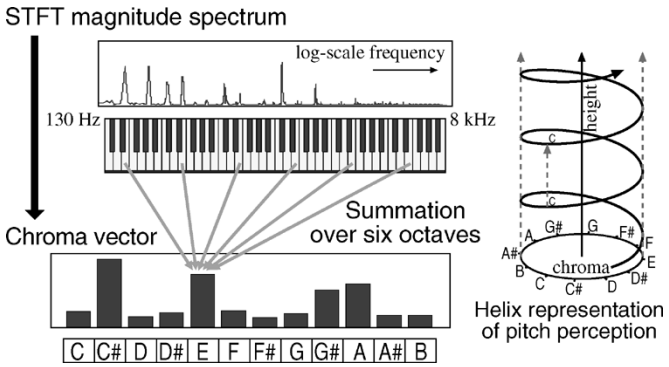

Fig. 3. Overview of calculating a 12-dimensional chroma vector. The magnitude at six different octaves is summed into just one octave which is divided into 12 log-spaced divisions corresponding to pitch classes. The Shepard's helix representation of musical pitch perception [18] is shown at the right.

A. Extract Acoustic Feature

Fig. 3 shows an overview of calculating the *chroma vector*, which is a perceptually-motivated feature vector using the concept of *chroma* in the Shepard's helix representation of musical pitch perception [18]. According to Shepard [18], the perception of pitch with respect to a musical context can be graphically represented by using a continually cyclic helix that has two dimensions, *chroma* and *height*, as shown at the right of Fig. 3. Chroma refers to the position of a musical pitch within an octave that corresponds to a cycle of the helix: it refers to the position on the circumference of the helix seen from directly above. On the other hand, height refers to the vertical position of the helix seen from the side (the position of an octave). Here, there are two major types of cue for pitch perception: "temporal cues" based on the periodicity of auditory nerve firing and "place cues" based on the position on the basilar membrane [19]. A study by Fujisaki and Kashino [20] indicates that the temporal cue is important for chroma identification, and that the place cue is important for height judgment.

The chroma vector represents magnitude distribution on the chroma that is discretized into 12 pitch classes within an octave: the basic idea is to coil the magnitude spectrum around the helix and squash it flat to project the frequency axis to the chroma. The 12-dimensional chroma vector $\vec{v}(t)$ is extracted from the magnitude spectrum, $\Psi_p(f,t)$ at the log-scale frequency f at time t, calculated by using the short-time Fourier transform (STFT). Each element of $\vec{v}(t)$ corresponds to a pitch class c ($c =$

$1, 2, \ldots, 12)$ in the equal temperament and is represented as $v_c(t)$

$$v_c(t) = \sum_{h=\mathrm{Oct_L}}^{\mathrm{Oct_H}} \int_{-\infty}^{\infty} \mathrm{BPF}_{c,h}(f)\Psi_p(f,t)df. \quad (1)$$

The $\mathrm{BPF}_{c,h}(f)$ is a bandpass filter that passes the signal at the log-scale frequency $F_{c,h}$ (in cents) of pitch class c (chroma) in octave position h (height)

$$F_{c,h} = 1200h + 100(c-1) \quad (2)$$

where frequency f_{Hz} in hertz is converted to frequency f_{cent} in cents so that there are 100 cents to a tempered semitone and 1200 to an octave

$$f_{\mathrm{cent}} = 1200 \log_2 \frac{f_{\mathrm{Hz}}}{440 \times 2^{\frac{3}{12}-5}}. \quad (3)$$

The $\mathrm{BPF}_{c,h}(f)$ is defined using a Hanning window as follows:

$$\mathrm{BPF}_{c,h}(f) = \frac{1}{2}\left(1 - \cos\frac{2\pi\left(f - (F_{c,h} - 100)\right)}{200}\right). \quad (4)$$

This filter is applied to octaves from $\mathrm{Oct_L}$ to $\mathrm{Oct_H}$.

In the current implementation, the input signal is digitized at 16 bit/16 kHz, and then the STFT with a 4096-sample Hanning window is calculated using the fast Fourier transform (FFT). Since the FFT frame is shifted by 1280 samples, the discrete time step (1 frame shift) is 80 ms. The $\mathrm{Oct_L}$ and $\mathrm{Oct_H}$, the octave range for the summation of (1), are, respectively, three and eight. This covers six octaves (130 Hz–8 kHz).

There are several advantages to using the chroma vector.[2] Because it captures the overall harmony (pitch-class distribution), it can be similar even if accompaniments or melody lines are changed in some degree after repetition. In fact, we have confirmed that the chroma vector is effective for identifying chord names [22], [23].[3] The chroma vector also enables modulated repetition to be detected as described in Section IV-E.

B. Calculate Similarity

The similarity $r(t,l)$ between the chroma vectors $\vec{v}(t)$ and $\vec{v}(t-l)$ is defined as

$$r(t,l) = 1 - \frac{\left|\frac{\vec{v}(t)}{\max_c v_c(t)} - \frac{\vec{v}(t-l)}{\max_c v_c(t-l)}\right|}{\sqrt{12}} \quad (5)$$

where l $(0 \le l \le t)$ is the lag. Since the denominator $\sqrt{12}$ is the length of the diagonal line of a 12-dimensional hypercube with edge length 1, $r(t,l)$ satisfies $0 \le r(t,l) \le 1$. In our experience with chroma vectors, the combination of the above similarity using the Euclidean distance and the vector normalization using a maximum element is superior to the similarity using the cosine angle (scalar product) and other vector normalization techniques.

[2]The chroma vector is similar to the chroma spectrum [21] that is used in reference [4], although its formulation is different.

[3]Other studies [24]–[26] have also shown the effectiveness of using the concept of chroma for identifying chord names.

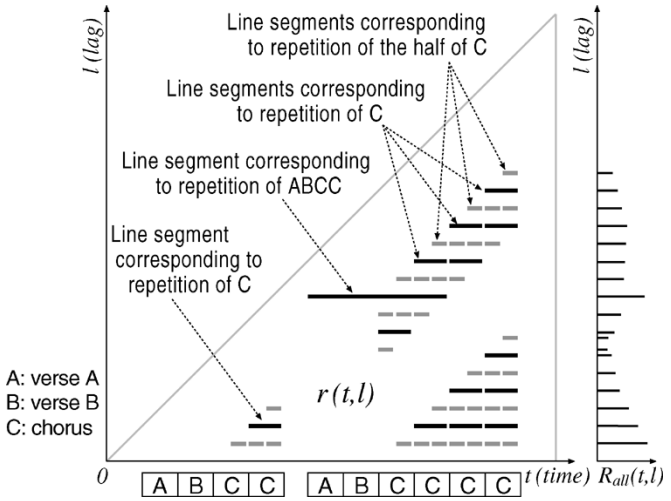

Fig. 4. Sketch of line segments, the similarity $r(t,l)$, and the possibility $R_{\mathrm{all}}(t,l)$ of containing line segments. The similarity $r(t,l)$ is defined in the right-angled isosceles triangle (time-lag triangle) in the lower right-hand corner. The actual $r(t,l)$ is noisy and ambiguous and usually contains many line segments irrelevant to chorus sections.

C. List Repeated Sections

Pairs of repeated sections are obtained from the similarity $r(t,l)$. Considering that $r(t,l)$ is drawn within a right-angled isosceles triangle in the two-dimensional time-lag $(t-l)$ space (*time-lag triangle*) as shown in Fig. 4, the method finds *line segments* that are parallel to the horizontal time axis and that indicate consecutive regions with high $r(t,l)$. When the section between times $T1$ and $T2$ is denoted $[T1, T2]$, each line segment between the points $(T1, L1)$ and $(T2, L1)$ is represented as $(t = [T1, T2], l = L1)$, which means that the section $[T1, T2]$ is similar to (i.e., is a repetition of) the section $[T1 - L1, T2 - L1]$. In other words, each horizontal line segment in the time-lag triangle indicates a repeated-section pair.

We, therefore, need to detect all horizontal line segments in the time-lag triangle $r(t,l)$. To find a horizontal line segment $(t = [T1, T2], l = L1)$, the possibility of containing line segments at the lag l, $R_{\mathrm{all}}(t,l)$,[4] is evaluated at the current time t (e.g., at the end of a song) as follows (Fig. 4):

$$R_{\mathrm{all}}(t,l) = \frac{1}{t-l+1}\sum_{\tau=l}^{t} r(\tau,l). \quad (6)$$

Before this calculation, $r(t,l)$ is normalized by subtracting a local mean value while removing noise and emphasizing horizontal lines. In more detail, given each point $r(T,L)$ in the time-lag triangle, six-directional local mean values of L_{size} points along the right, left, upper, lower, upper-right, and lower-left directions starting from the point $r(T,L)$ are calculated, and the maximum and minimum are obtained ($L_{\mathrm{size}} = 15$ points (1.2 s)). If the local mean along the right or left direction (i.e., $\sum_{\tau=1}^{L_{\mathrm{size}}} r(T+\tau, L)/L_{\mathrm{size}}$ or $\sum_{\tau=1}^{L_{\mathrm{size}}} r(T-\tau, L)/L_{\mathrm{size}}$) takes the maximum, $r(T,L)$ is considered part of a horizontal line and emphasized by subtracting the minimum from $r(T,L)$. Otherwise, $r(T,L)$ is

[4]This can be considered the Hough transform where only horizontal lines are detected: the parameter (voting) space $R_{\mathrm{all}}(t,l)$ is, therefore, simply one dimensional along l.

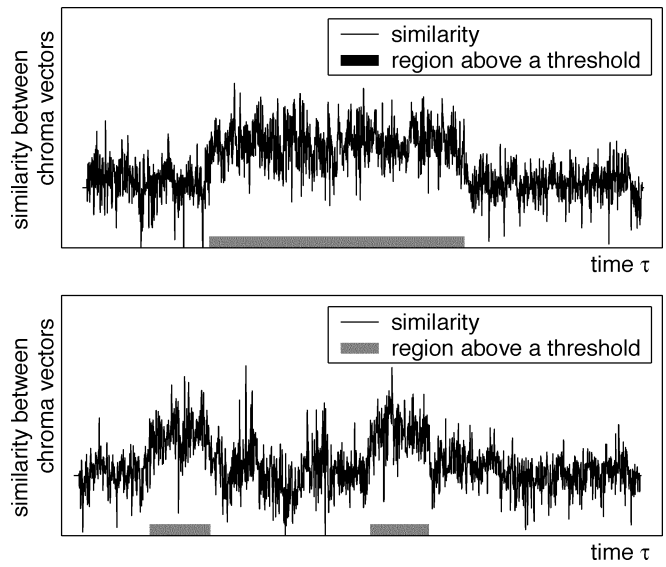

Fig. 5. Examples of the similarity $r(\tau, L1)$ at high-peak lags $L1$. The bottom horizontal bars indicate the regions above an automatically adjusted threshold, which means they correspond to line segments.

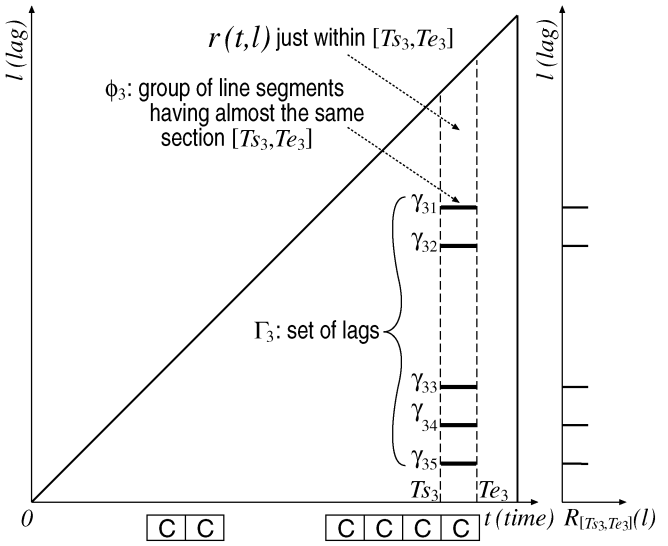

Fig. 6. Sketch of a group $\phi_3 = ([Ts_3, Te_3], \Gamma_3)$ of line segments that have almost the same section $[Ts_3, Te_3]$, a set Γ_3 of those lags γ_{3j} $(j = 1, 2, \ldots, 5)$, and the possibility $R_{[Ts_3,Te_3]}(l)$ of containing line segments within $[Ts_3, Te_3]$.

considered noise and suppressed by subtracting the maximum from $r(T, L)$; noise tends to appear as lines along the upper, lower, upper right, and lower left directions.

The method then picks up each peak in $R_{\rm all}(t, l)$ along the lag l by finding a point where the smoothed differential of $R_{\rm all}(t, l)$

$$\sum_{w=-K_{\rm size}}^{K_{\rm size}} w R_{\rm all}(t, l + w) \tag{7}$$

changes sign from positive to negative [27] ($K_{\rm size} = 4$ points (0.32 s)). Before this calculation, it removes the global drift caused by cumulative noise in $r(t, l)$ from $R_{\rm all}(t, l)$: it subtracts, from $R_{\rm all}(t, l)$, a smoothed $R_{\rm all}(t, l)$ low-pass filtered by using a moving average whose weight function is the second-order cardinal B-spline having $B_{\rm size}$ points on each slope ($B_{\rm size} = 200$ points (16 s)); this subtraction is equivalent to obtaining a high-pass-filtered $R_{\rm all}(t, l)$.

The method then selects only high peaks above a threshold to search the line segments. Because this threshold is closely related to the repetition-judgment criterion which should be adjusted for each song, we use an automatic threshold selection method based on a discriminant criterion [17]. When dichotomizing the peak heights into two classes by a threshold, the optimal threshold is obtained by maximizing the discriminant criterion measure defined by the following between-class variance:

$$\sigma_B^2 = \omega_1 \omega_2 (\mu_1 - \mu_2)^2 \tag{8}$$

where ω_1 and ω_2 are the probabilities of class occurrence (number of peaks in each class/total number of peaks), and μ_1 and μ_2 are the means of the peak heights in each class.

For each picked-up high peak with lag $L1$, the line segments are finally searched in the direction of the horizontal time axis on the one-dimensional function $r(\tau, L1)$ $(L1 \le \tau \le t)$ (Fig. 5). After smoothing $r(\tau, L1)$ using a moving average filter whose weight function is the second-order cardinal B-spline having $B'_{\rm size}$ points on each slope ($B'_{\rm size} = 25$ points (2 s)), the method obtains line segments on which the smoothed $r(\tau, L1)$ is above a threshold and whose length is long enough (more than 6.4 s). This threshold is also adjusted using the above automatic threshold selection method based on the discriminant criterion. Here, instead of dichotomizing peak heights, the method selects the top five peak heights of $R_{\rm all}(t, l)$, obtains the five lags l_k $(k = 1, 2, \ldots 5)$ corresponding to those selected high peaks, and dichotomizes all the values of the smoothed $r(\tau, l_k)$ $(l_k \le \tau \le t)$ at those lags l_k.

D. Integrate Repeated Sections

Since each line segment indicates just a pair of repeated sections, it is necessary to organize into a group the line segments that have common sections. Suppose a section is repeated n times $(n \ge 3)$, the number of line segments to be grouped together should theoretically be $n(n-1)/2$ if all of them are found in the time-lag triangle. First, line segments that have almost the same section $[Ts_i, Te_i]$ are organized into a group; more specifically, two line segments are grouped when both the difference between their beginning points and the difference between their end points are smaller than a dynamic threshold equal to $T_{\rm ratio}$ percent of the segment length with a ceiling of $T_{\rm size}$ points ($T_{\rm ratio} = 20\%$ and $T_{\rm size} = 45$ points (3.6 s)). The group is represented as $\phi_i = ([Ts_i, Te_i], \Gamma_i)$, where $\Gamma_i = \{\gamma_{ij} | j = 1, 2, \ldots, M_i\}$ (M_i is the number of line segments in the group) is a set of the lags γ_{ij} of those segments—corresponding to the high peaks in $R_{\rm all}(t, l)$—in this group (Fig. 6). A set of these groups is denoted by $\Phi = \{\phi_i | i = 1, 2, \ldots, N\}$ (N is the number of all groups).

Aiming to exhaustively detect all the repeated (chorus) sections, the method then redetects some missing (hidden) line segments not found in the bottom-up detection process (described in Section IV-C) through top-down processing using information on other detected line segments. In Fig. 4, for example, we

can expect that two line segments corresponding to the repetition of the first and third C and the repetition of the second and fourth C, which overlap with the long line segment corresponding to the repetition of ABCC, are found even if they were hard to find in the bottom-up process.

For this purpose, line segments are searched again by using $r(t,l)$ just within $[Ts_i, Te_i]$ of each group ϕ_i. Starting from

$$R_{[Ts_i,Te_i]}(l) = \frac{1}{Te_i - Ts_i + 1} \sum_{\tau=Ts_i}^{Te_i} r(\tau, l) \tag{9}$$

instead of $R_{\mathrm{all}}(t,l)$, the method performs almost the same peak-picking process described in Section IV-C and forms a new set Γ_i of high peaks γ_{ij} above a threshold in $R_{[Ts_i,Te_i]}(l)$ (Fig. 6). In more detail, it picks up each peak by finding a point where the smoothed differential of $R_{[Ts_i,Te_i]}(l)$

$$\sum_{w=-K'_{\mathrm{size}}}^{K'_{\mathrm{size}}} w R_{[Ts_i,Te_i]}(l+w) \tag{10}$$

changes sign from positive to negative [$K'_{\mathrm{size}} = 35$ points (2.8 s)]. Before this calculation, it also removes the global drift in the same way by smoothing with the second-order cardinal B-spline having B_{size} points on each slope. This threshold is again adjusted using the above automatic threshold selection method based on the discriminant criterion. Here, the method optimizes the threshold by dichotomizing all local peak heights of $R_{[Ts_i,Te_i]}(l)$ taken from all groups ϕ_i of Φ.

The method then removes inappropriate peaks in each Γ_i as follows.

1) Remove unnecessary peaks that are equally spaced.

   When similar accompaniments are repeated throughout most of a song, peaks γ_{ij} irrelevant to chorus sections tend to appear at even intervals in $R_{[Ts_i,Te_i]}(l)$. A group ϕ_i where the number of equally spaced peaks exceeds U_{th} is judged to be irrelevant to chorus sections and is removed from Φ ($U_{\mathrm{th}} = 10$). For this judgment, we consider only peaks that are higher than a threshold determined by the standard deviation of the lower half of peaks. In addition, when the number of equally spaced low peaks is more than U'_{th}, those peaks γ_{ij} are judged to be irrelevant to chorus sections and are removed from Γ_i ($U'_{\mathrm{th}} = 5$). For this judgment, we consider only peaks that are higher than the above threshold and lower than the average of the above threshold and the highest peak.
2) Remove a peak whose line segment has a highly deviated similarity.

   When only part of similarity $r(t, \gamma_{ij})$ at a peak γ_{ij} within $[Ts_i, Te_i]$ is high, its peak is not appropriate for use. A peak γ_{ij} is removed from Γ_i when the standard deviation of $r(\tau, \gamma_{ij})$ ($Ts_i \le \tau \le Te_i$) after smoothing with the above second-order cardinal B-spline (having B'_{size} points on each slope) is larger than a threshold. Since peaks detected in Section IV-C can be considered reliable, this threshold is determined as U_{ratio} multiplied by the maximum of the above standard deviation at all those peaks ($U_{\mathrm{ratio}} = 1.4$).
3) Remove a peak that is too close to other peaks and causes sections to overlap.

   To avoid sections overlapping, it is necessary to make the interval between adjacent peaks along the lag greater than the length of its section. One of every pair of peaks having an interval less than the section length is removed so that higher peaks can remain overall.

Finally, by using the lag γ_{ij} corresponding to each peak of Γ_i, the method searches for a group whose section is $[Ts_i - \gamma_{ij}, Te_i - \gamma_{ij}]$ (i.e., is shared by the current group Γ_i) and integrates it with Γ_i if it is found. They are integrated by adding all the peaks of the found group to Γ_i after adjusting the lag values (peak positions); the found group is then removed. In addition, if there is a group that has a peak indicating the section $[Ts_i - \gamma_{ij}, Te_i - \gamma_{ij}]$, it too is integrated.

E. Integrate Repeated Sections With Modulation

The processes described above do not deal with modulation (key change), but they can easily be extended to it. A modulation can be represented by the pitch difference of its key change, ζ $(0, 1, \ldots, 11)$, which denotes the number of tempered semitones. For example, $\zeta = 9$ means the modulation of nine semitones upward or the modulation of three semitones downward.

One of the advantages of the 12-dimensional chroma vector $\vec{v}(t)$ is that a transposition amount ζ of the modulation can naturally correspond to the amount by which its 12 elements are shifted (rotated). When $\vec{v}(t)$ is the chroma vector of a certain performance and $\vec{v}(t)'$ is the chroma vector of the performance that is modulated by ζ semitones upward from the original performance, they tend to satisfy

$$\vec{v}(t) \approx S^{\zeta} \vec{v}(t)' \tag{11}$$

where S is a 12-by-12 shift matrix[5] defined by

$$S = \begin{pmatrix} 0 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \\ 1 & 0 & \cdots & \cdots & \cdots & 0 \end{pmatrix}. \tag{12}$$

To detect the modulated repetition by using this feature of chroma vectors and considering 12 destination keys, we calculate 12 kinds of extended similarity for each ζ as follows:

$$r_{\zeta}(t,l) = 1 - \frac{\left| \frac{S^{\zeta}\vec{v}(t)}{\max_c v_c(t)} - \frac{\vec{v}(t-l)}{\max_c v_c(t-l)} \right|}{\sqrt{12}}. \tag{13}$$

Starting from each $r_{\zeta}(t,l)$, the processes of listing and integrating the repeated sections are performed as described in Sections IV-C and D, except that the threshold automatically adjusted at $\zeta = 0$ is used for the processes at $\zeta \neq 0$ (which suppresses harmful false detection of nonrepeated sections). After these processes, 12 sets of line-segment groups are obtained for 12 kinds of ζ. To organize nonmodulated and modulated repeated sections into the same groups, the method integrates several groups across all the sets if they share the same section.

[5]Note that this shift (rotation) operation is not applicable to other acoustic features such as simple power spectrums and MFCC features.

Hereafter, we use $\phi_i = ([Ts_i, Te_i], \Gamma_i)$ to denote the groups of line segments obtained from all the ζ. By unfolding each line segment of γ_{ij} to the pair of repeated sections indicated by it, we can obtain

$$\Lambda_i = \{([Ps_{ij}, Pe_{ij}], \lambda_{ij}) \,|\, j = 1, 2, \ldots, M_i + 1\}, \quad (14)$$

where $[Ps_{ij}, Pe_{ij}]$ represents an unfolded repeated section that corresponds to the lag γ_{ij} and is calculated by $[Ps_{ij}, Pe_{ij}] = [Ts_i - \gamma_{ij}, Te_i - \gamma_{ij}]$. The λ_{ij} is the possibility of being repeated sections (the possibility that the sections $[Ts_i, Te_i]$ and $[Ps_{ij}, Pe_{ij}]$ are really repeated), and is defined as the mean of the similarity $r_\zeta(t, l)$ on the corresponding line segment. For $j = M_i + 1$ corresponding not to a line segment but to the section $[Ts_i, Te_i]$ itself, we define $[Ps_{ij}, Pe_{ij}]$ and λ_{ij} as $[Ps_{ij}, Pe_{ij}] = [Ts_i, Te_i]$ and $\lambda_{ij} = \max_{k=1}^{M_i} \lambda_{ik}$. The modulated sections are labeled with their ζ for reference.

F. Select Chorus Sections

After evaluation of the *chorus measure* ν_i, which is the possibility of being chorus sections for each group i (ϕ_i, Λ_i), the group m that maximizes the chorus measure ν_i is selected as the chorus sections

$$m = \arg\max_i \nu_i. \quad (15)$$

The chorus measure ν_i is a sum of λ_{ij} weighted by the length of the section and is defined by

$$\nu_i = \left(\sum_{j=1}^{M_i+1} \lambda_{ij} \right) \log \frac{Te_i - Ts_i + 1}{D_{\text{len}}} \quad (16)$$

where D_{len} is a constant (1.4 s). Before ν_i is calculated, the possibility λ_{ij} of each repeated section is adjusted according to three assumptions (heuristics), which fit a large class of popular music.

[Assumption 1]: The length of the chorus section has an appropriate, allowed range (7.7 to 40 s in the current implementation). If the length is out of this range, λ_{ij} is set to 0.

[Assumption 2]: When a repeated section is long enough to be likely to correspond to long-term repetition such as verse A, verse B, and chorus, the chorus section is likely to be near its end. If there is a repeated section $[Ps_{ij}, Pe_{ij}]$ whose end is close to the end of another long repeated section (longer then 50 s), its λ_{ij} is doubled; i.e., λ_{ij} is doubled if the difference of the end points of those sections is smaller than T_{size} points.

[Assumption 3]: Because a chorus section tends to have two half-length repeated subsections within its section, a section having such subsections is likely to be the chorus section. If there is a repeated section $[Ps_{ij}, Pe_{ij}]$ that has such subsections in another group, half of the mean of the possibility of the two subsections is added to its λ_{ij}.

The RefraiD method then outputs a list of chorus sections found as explained above as well as a list of repeated sections obtained as its intermediate result. As postprocessing for the chorus sections $[Ps_{mj}, Pe_{mj}]$ of Λ_m determined by (15), only a small gap between adjacent chorus sections is padded (eliminated) by equally prolonging the end of those sections; more specifically, only when the gap is smaller than D_{th} points or half of the section length, it is padded $[D_{\text{th}} = 150$ points $(12$ s$)]$.

TABLE II
PARAMETER VALUES

parameter	value
L_{size} (for the six-directional local mean values)	15 points (1.2 s)
K_{size} (for the smoothed differential of $R_{all}(t, l)$)	4 points (0.32 s)
B_{size} (for the moving average (cardinal B-spline))	200 points (16 s)
B'_{size} (for the moving average (cardinal B-spline))	25 points (2 s)
T_{ratio} (for grouping line segments)	20 percent
T_{size} (for grouping line segments)	45 points (3.6 s)
K'_{size} (for the smoothed differential)	35 points (2.8 s)
U_{th} (for removing inappropriate peaks)	10
U'_{th} (for removing inappropriate peaks)	5
U_{ratio} (for removing inappropriate peaks)	1.4
D_{len} (for calculating the chorus measure ν_i)	17.5 points (1.4 s)
D_{th} (for padding a gap between chorus sections)	150 points (12 s)

V. EXPERIMENTAL RESULTS

The RefraiD method has been implemented in a real-time system that takes a musical audio signal as input and outputs a list of the detected chorus sections and repeated sections. Along with the real-time audio input, the system can display visualized lists of chorus sections and repeated sections, which are obtained using just the past input and are considered the most probable at every instance. The final detected results for a song are obtained at the end of the song. The parameter values in the current implementation are listed in Table II.

We evaluated the accuracy of chorus section detection done through the RefraiD method. The method was tested on 100 songs[6] of the popular-music database "*RWC Music Database: Popular Music*" (RWC-MDB-P-2001 Nos. 1–100) [15], [16], which is an original database available to researchers around the world. These 100 songs were originally composed, arranged, performed, and recorded in a way that reflected the complexity and diversity of real-world music. In addition, to provide a reference for judging whether detection results were right or wrong, correct chorus sections in targeted songs had to be labeled manually. To enable this, we developed a song structure labeling editor that can divide up a song and correctly label chorus sections.

We compared the output of the proposed method with the correct chorus sections that were hand-labeled by using this labeling editor. The degree of matching between the detected and correct chorus sections was evaluated using the F-measure [28], which is the harmonic mean of the recall rate (R) and the precision rate (P)

$$\text{F}-\text{measure} = \frac{2RP}{R+P} \quad (17)$$

$$R = \frac{\text{total length of correctly detected chorus sections}}{\text{total length of correct chorus sections}} \quad (18)$$

$$P = \frac{\text{total length of correctly detected chorus sections}}{\text{total length of detected chorus sections}}. \quad (19)$$

The output for a song was judged to be correct if its F-measure was more than 0.75. For the case of modulation (key change), a chorus section was judged correctly detected only if the relative width of the key shift matched the actual width.

[6]99, 64, and 54 songs out of 100 fit assumptions 1–3, respectively.

TABLE III
RESULTS OF EVALUATING REFRAID: NUMBER OF SONGS WHOSE CHORUS SECTIONS WERE DETECTED CORRECTLY UNDER FOUR SETS OF CONDITIONS

	Condition (enabled:○, disabled:×)			
Modulation detection	○	×	○	×
Use of assumptions 2 & 3	○	○	×	×
Number of songs (out of 100)	80	74	73	68

The results are listed in Table III. The method dealt correctly with 80 songs[7] out of 100 (with the averaged F-measure of those 80 songs being 0.938). The main reasons for the method making mistakes were choruses that did not repeat more often than other sections and the repetition of similar accompaniments throughout most of a song. Among these 100 songs, ten songs (RWC-MDB-P-2001 Nos. 3, 18, 22, 39, 49, 71, 72, 88, 89, and 90) included modulated chorus sections (these songs are referred to as "modulated songs" in the following), and nine of these songs (except for no. 72) were dealt with correctly (the F-measure was more than 0.75). While the modulation itself was correctly detected in all of the ten modulated songs, modulated sections were not correctly selected as the chorus sections in two of the songs (Nos. 72 and 89).[8] There were 22 songs (RWC-MDB-P-2001 Nos. 3, 5, 9, 14, 17, 19, 24, 25, 33, 36, 37, 38, 44, 46, 50, 57, 58, 64, 71, 91, 96, and 100) that had choruses exhibiting significant changes in accompaniment or melody on repetition, and 21 of these (except for no. 91) were dealt with correctly (the F-measure was more than 0.75); the repeated chorus section itself was correctly detected in 16 of these (except for Nos. 5, 17, 25, 44, 57, and 91). These results show that the method is robust enough to deal with real-world audio signals.

When the function to detect the modulated repetition (referred to as the "modulation detector" in the following) was disabled, only 74 songs were dealt with correctly. On the other hand, when assumptions 2 and 3 were not used, the performance fell as shown by the entries in the two rightmost columns of Table III. Enabling the modulation detector without assumptions 2 and 3 increased the number of correctly detected songs from 68 to 73 (the five additional songs were Nos. 3, 4, 22, 88, and 90), and enabling it with assumptions 2 and 3 increased the number from 74 to 80 (additional songs were the above five songs plus no. 39). Using assumptions 2 and 3 increased the number of correctly detected songs from 68 to 74 with the modulation detector and from 73 to 80 songs without it: in the former case, seven additional songs were correctly detected (Nos. 10, 25, 33, 38, 44, 46, and 82), but one song (no. 39) which was previously detected correctly was not detected; in the latter case, the same additional songs were correctly detected. Under the four sets of conditions, four songs (Nos. 18, 49, 71, and 89) of the ten modulated songs were always dealt with correctly and one song (no. 72) was never dealt with correctly, while the averaged F-measure of Nos. 18, 49, and 71 was improved from 0.827 to 0.974 by using the modulation detector. In all cases, the modulated sections themselves were not correctly detected when the modulation detector was disabled because the similarity based on chroma vectors is sensitive to the modulation. These results show the effectiveness of the modulation detector and assumptions 2 and 3.

[7]The F-measure was not more than 0.75 for RWC-MDB-P-2001 Nos. 2, 12, 16, 29, 30, 31, 41, 53, 56, 59, 61, 66, 67, 69, 72, 79, 83, 91, 92, and 95.

[8]Even if the modulated chorus section itself was not selected in RWC-MDB-P-2001 no. 89, the song was detected correctly because its F-measure (0.877) was more than 0.75.

VI. APPLICATION: MUSIC LISTENING STATION WITH CHORUS-SEARCH FUNCTION

When "trial listening" to prerecorded music on compact discs (CDs) at a music store, a listener often takes an active role in the playback of musical pieces or songs by picking out only those sections of interest. This new type of music interaction differs from passive music appreciation in which people usually listen to entire musical selections. To give some background, music stores in recent years have installed *music listening stations* to allow customers to listen to CDs on a trial basis to facilitate a purchasing decision. In general, the main objective of listening to music is to appreciate it, and it is common for a listener to play a musical selection from start to finish. In trial listening, however, the objective is to quickly determine whether a selection is the music one has been looking for and whether one likes it, so listening to entire selections in the above manner is rare. In the case of popular music, for example, customers often want to listen to the chorus to pass judgment on that song. This desire produces a special way of listening in which the trial listener first listens briefly to a song's "intro" and then jumps ahead in search of the chorus by repeatedly pushing the fast-forward button, eventually finding the chorus and listening to it.

The functions provided by conventional listening stations for music CDs, however, do not support this unique way of trial listening very well. These listening stations are equipped with playback-operation buttons typical of an ordinary CD player, and among these, only the fast-forward and rewind buttons can be used to find the chorus section of a song. On the other hand, the digital listening stations that have recently been installed in music stores enable playback of musical selections from a hard disk or over a network. Here, however, only one part (e.g., the beginning) of each musical selection (an interval of about 30–45 s) is mechanically excerpted and stored, which means that a trial listener may not necessarily hear the chorus section.

Against the above background, we propose *SmartMusicKIOSK*, a music listening station equipped with a chorus search function. With SmartMusicKIOSK, a trial listener can jump to the beginning of a song's chorus (perform an instantaneous fast-forward to the chorus) by simply pushing the button for this function. This eliminates the hassle of manually searching for the chorus. SmartMusicKIOSK also provides a function for jumping to the beginning of the next structural (repeated) section of the song.

Much research has been performed in the field of music information processing, especially in relation to music information retrieval and music understanding, but there has been practically none in the area of trial listening. Interaction between people and music can be mainly divided into two types: the creating/active side (composing, performing, etc.) and the receiving/passive side (appreciating music, hearing background music, etc.).

Trial listening, on the other hand, differs from the latter type, that is, musical appreciation, since it involves listening to musical selections while taking an active part in their playback. This is why we felt that this activity would be a new and interesting subject for research.

A. Past Forms of Interaction in Music Playback

The ability to play an interactive role in music playback by changing the current playback position is a relatively recent development in the history of music. In the past, before it became possible to record the audio signals of music, a listener could only listen to a musical piece at the place where it was performed live. Then, when the recording of music to records and tape became a reality, it did become possible to change playback from one musical selection to another, but the bother and time involved in doing so made this a form of nonreal-time interaction. The ability of a listener to play back music interactively really only began with the coming of technology for recording music onto magnetooptical media like CDs. These media made it possible to move the playback position almost instantly with just a push of a button making it easy to jump from one song to another while listening to music.

However, while it became easy to move between selections (CD tracks), there was not sufficient support for interactively changing the playback position within a selection as demanded by trial listening. Typical playback operation buttons found on conventional CD players (including music listening stations) are play, pause, stop, fast-forward, rewind, jump to next track, and jump to previous track (a single button may be used to perform more than one function). Among these, only the fast-forward and rewind buttons can change the playback position within a musical selection. Here, however, listeners are provided with only the following three types of feedback as aids to finding the position desired:

1) sound of fast playback that can be heard while holding down the fast-forward/rewind button;
2) sound after releasing the button;
3) display of elapsed time from the start of the selection in question.

Consequently, a listener who wanted to listen to the chorus of a song, for example, would have to look for it manually by pressing and releasing a button any number of times.

These types of feedback are essentially the same when using media-player software on a personal computer (PC) to listen to songs recorded on a hard disk, although a playback slider may be provided. The total length of the playback slider corresponds to the length of a song, and the listener can manipulate the slider to jump to any position in a song. Here as well, however, the listener must use manual means to search out a specific playback position, so nothing has really changed.

B. Intelligent Music Listening Station: SmartMusicKIOSK

For music that would normally not be understood unless some time was taken for listening, the problem here is how to enable changing between specific playback positions before actual listening. We propose the following two functions to solve this problem, assuming the main target is popular music.

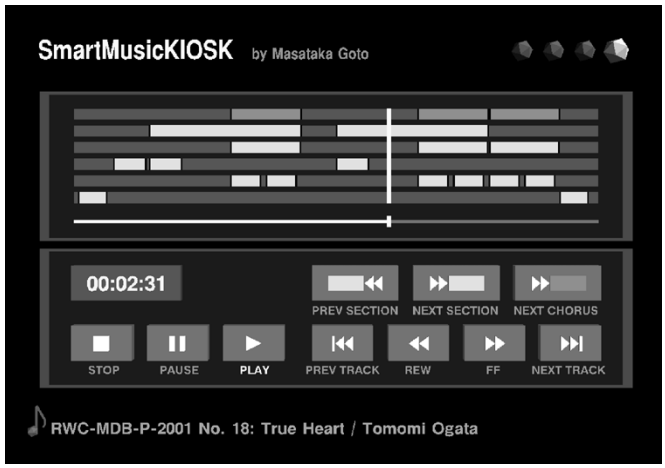

Fig. 7. SmartMusicKIOSK screen display. The lower window presents the playback operation buttons and the upper window provides a visual representation of a song's contents (results of automatic chorus section detection using RWC-MDB-P-2001 no. 18 of the RWC Music Database [15], [16]).

1) *"Jump to chorus" function: automatic jumping to the beginning of sections relevant to a song's structure:* Functions are provided enabling automatic jumping to sections that will be of interest to listeners. These functions are "jump to chorus (NEXT CHORUS button)," "jump to previous section in song (PREV SECTION button)," and "jump to next section in song (NEXT SECTION button)," and they can be invoked by pushing the buttons shown above in parentheses. With these functions, a listener can directly jump to and listen to chorus sections, or jump to the previous or next repeated section of the song.
2) *"Music map" function: visualization of song contents:* A function is provided to enable the contents of a song to be visualized to help the listener decide where to jump next. Specifically, this function provides a visual representation of the song's structure consisting of chorus sections and repeated sections, as shown in Fig. 7. While examining this display, the listener can use the automatic jump buttons, the usual fast-forward/rewind buttons, or a playback slider to move to any point of interest in the song.

The following describes the lower and upper windows shown in Fig. 7.

- *Playback operation window* (lower window): The three automatic jump buttons added to the conventional playback-operation buttons are named NEXT CHORUS, PREV SECTION, and NEXT SECTION. These buttons are marked with newly designed symbols.

  Pressing the NEXT CHORUS button causes the system to search for the next chorus in the song from the present position (returning to the first one if none remain) and to jump to the start of that chorus. Pressing the other two buttons causes the system to search for the immediately following section or immediately preceding section with respect to the present position and to jump to the start of that section. While searching, the system ignores section-end points.

- *Song-structure display window* (upper window): The top row of this display provides a visual representation of chorus sections while the lower rows (five maximum in the current implementation) provide a visual representation of repeated sections. On each row, colored sections indicate similar (repeated) sections. In Fig. 7, for example, the second row from the top indicates the structural repetition of "verse A $\Rightarrow$ verse B $\Rightarrow$ chorus" (the longest repetition of a visual representation often suggests such a structural repetition); the bottom row with two short colored sections indicates the similarity between the "intro" and "ending" of this song. In addition, the thin horizontal bar at the very bottom of this window is a playback slider whose position corresponds to elapsed time in the song.

  Clicking directly on a section (touching in the case of a touch panel or tablet PC) plays that section, and clicking the playback slider changes the playback position.

The above interface functions promote a type of listening in which the listener can first listen to the intro of a song for just a short time and then jump and listen to the chorus with just a push of a button.[9] Furthermore, visualizing the entire structure of a song allows the listener to choose various parts of a song for trial listening.

C. System Implementation and Results

We built a SmartMusicKIOSK system incorporating all the functions described in Section VI-B. The system is executed with files that include descriptions of chorus sections and repeated sections, which can be obtained beforehand by the RefraiD method. Although the results of automatic detection include errors and are, therefore, not 100% accurate as described in Section V, they still provide the listener with a valuable aid to finding a desired playback position and make a listening station much more convenient than in the past. If, however, there are times when an accurate description is required, results of automatic detection may be manually corrected. The song structure labeling editor described in Section V can also be used for this manual correction and labeling. This is useful especially for songs not suitable for automatic detection or outside the category of popular music.

In the SmartMusicKIOSK system, the song file playback engine, graphical user interface (GUI) module, and audio device control module are all implemented as separate processes to improve extendibility. These processes have been ported on several operating systems, such as Linux, SGI IRIX, and Microsoft Windows, and can be distributed over a LAN (Ethernet) and connected by using a network protocol called *Remote Audio Control Protocol (RACP)*, which we have designed to enable efficient sharing of audio signals and various types of control information. This protocol is an extension of remote music control protocol (RMCP) [29] enabling the transmission of audio signals.

[9]Both a "PREV CHORUS" and "NEXT CHORUS" button may also be prepared in the playback operation window. Only one button was used here for the following reasons. 1) Pushing the present NEXT CHORUS button repeatedly loops through all chorus sections enabling the desired chorus to be found quickly. 2) A previous chorus can be returned to immediately by simply clicking on that section in the song structure display window.

Fig. 8. Demonstration of SmartMusicKIOSK implemented on a tablet PC.

Fig. 8 shows a photograph of the SmartMusicKIOSK system taken during a technical demonstration in February 2003. This system can be executed on a stand-alone tablet PC (Microsoft Windows XP Tablet PC Edition, Pentium III 933-MHz CPU) as shown in the center of the photograph. It can be operated by touching the screen with a pen or by pushing the keys of an external keypad (center-right of the photograph) that duplicates the playback button group shown on the screen.

Our experience with the SmartMusicKIOSK demonstration showed that the proposed interface was effective enough to enable listeners to play back songs in an interactive manner by pushing jump buttons while receiving visual assistance from the music map display. The music map facilitated jump operations and the listening to various parts of a song while moving back and forth as desired on the song structure. The proposed functions were intuitively easy to use requiring no training: listeners who had received no explanation about jump button functions or display windows were nevertheless able to surmise their purpose in little time.

D. Discussion

In the following, we consider how interaction in music playback need not be limited to trial listening scenarios, and discuss what kinds of situation our method can be applied to.

1) Interface for Active Listening of Music: In recent years, the music usage scene has been expanding and usage styles of choosing music as one wishes, checking its content, and at times even extracting portions of music have likewise been increasing. For example, in addition to trial listening of CDs at music stores, end users select musical ring tones for cellular phones, find background music appropriate to certain situations, and use music on the World Wide Web. On the other hand, interfaces for music playback have become fixed to standard playback operation buttons even after the appearance of the CD player and computer-based media players as described in Section VI-A. Interfaces of this type, while suitable for passive appreciation of music, are inadequate for interactively finding sections of interest within a song.

As a general interface for music playback, we can see SmartMusicKIOSK as adding an interface that targets structural sections of a song as operational units in contrast to the conventional interface (e.g., a CD player) that targets only songs as operational units. With this conventional interface, songs of no interest to the listener can easily be skipped, but skipping sections of no interest within a particular song is not as easy. An outstanding advantage of the SmartMusicKIOSK interface is the ability to "listen to any part of a song whenever one likes" without having to follow the timeline of the original song. Extending this idea, it would be interesting to add a "shuffle play" function in units of musical sections by drawing an analogy from operation in song units.

While not expected when building this interface, an interesting phenomenon has appeared in situations that permit long-term listening as opposed to trial listening. Specifically, we have found some listeners tend to listen to music in a more analytical fashion, compared to past forms of music appreciation, when they can interactively change the playback position while viewing the structure of a musical piece. For example, we have observed listeners checking the kind of structure possessed by an entire piece, listening to each section in that structure, and comparing sections that repeat. Another finding is that visualization of a song's structure has proven to be interesting and useful for listeners who just want to passively appreciate music.

2) Other Applications: In addition to the SmartMusicKIOSK application, the RefraiD method has a potentially wide range of application. The following presents other application examples.

- *Digital listening station:* The RefraiD method could enable digital listening stations to excerpt and store chorus sections instead of mechanically stored excerpts. In the future, we hope to see digital listening stations in music stores upgrade to functions such as those of SmartMusicKIOSK.
- *Music thumbnail:* The ability to playback (preview) just the beginning of a chorus section detected by the RefraiD method would provide added convenience when browsing through a large set of songs or when presenting search results of music information retrieval. This function can be regarded as a music version of the image thumbnail.
- *Computer-based media players:* A variety of functions have recently been added to media players, such as exchangeable appearance (skins) and music-synchronized animation in the form of geometrical drawings moving synchronously with waveforms and frequency spectrums during playback. No essential progress, however, has been seen in the interface itself. We hope not only that the SmartMusicKIOSK interface will be adopted for various media players, but also that other approaches of reexamining the entire functional makeup of music playback interfaces will follow.

VII. Conclusion

We have described the RefraiD method which detects chorus sections and repeated sections in real-world popular music audio signals. It basically regards the most repeated sections as the chorus sections. Analysis of the relationships between various repeated sections enables all the chorus sections to be detected with their beginning and end points. In addition, introducing the similarity between nonshifted and shifted chroma vectors makes it possible to detect modulated chorus sections. Experimental results with the "RWC Music Database: Popular Music" showed that the method was robust enough to correctly detect the chorus sections in 80 of 100 songs.

We have also described the SmartMusicKIOSK application system, which is a music listening station based on the RefraiD method. It provides content-based playback controls allowing a listener to skim rapidly through music, plus a graphical overview of the entire song structure. While entire songs of no interest to a listener can be skipped on conventional music playback interfaces, SmartMusicKIOSK is the first interface that allows the listener to easily skip sections of no interest even within a song.

The RefraiD method has relevance to music summarization studies [6], [8]–[12], [30], none of which has addressed the problem of detecting all the chorus sections. One of the chorus sections detected by our method can be regarded as a song summary, as could another long repeated section in the intermediate-result list of repeated sections. Music summarization studies aimed at shortening the length of a song are also related to SmartMusicKIOSK because they share one of the objectives of trial listening, that is, to listen to music in a short time. Previous studies, however, have not considered an interactive form of listening as taken up by our research. From the viewpoint of trial listening, the ability of a listener to easily select any section of a song for listening in a true interactive fashion is very effective as discussed in Section VI-D.1.

Our repetition-based approach of the RefraiD method has proven effective for popular music. To improve the performance of the method, however, we will need to use prior information on acoustic features unique to choruses. We also plan to experiment with other music genres and extend the method to make it widely applicable. In addition, our future work will include research on new directions of making interaction between people and music even more active and enriching.

Acknowledgment

The author would like to thank H. Asoh (National Institute of Advanced Industrial Science and Technology) for his valuable discussions and the anonymous reviewers for their helpful comments and suggestions.

References

[1] M. Goto, "Music scene description project: toward audio-based real-time music understanding," in *Proc. Int. Conf. Music Information Retrieval*, 2003, pp. 231–232.

[2] ——, "A real-time music scene description system: Predominant-F0 estimation for detecting melody and bass lines in real-world audio signals," *Speech Commun.*, vol. 43, no. 4, pp. 311–329, 2004.

[3] B. Logan and S. Chu, "Music summarization using key phrases," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing* , 2000, pp. II-749–II-752.

[4] M. A. Bartsch and G. H. Wakefield, "To catch a chorus: using chroma-based representations for audio thumbnailing," in *Proc. IEEE Workshop Applications of Signal Processing to Audio and Acoustics*, 2001, pp. 15–18.

[5] M. Cooper and J. Foote, "Automatic music summarization via similarity analysis," in *Proc. Int. Conf. Music Information Retrieval*, 2002, pp. 81–85.
[6] R. B. Dannenberg and N. Hu, "Pattern discovery techniques for music audio," *J. New Music Res.*, vol. 32, no. 2, pp. 153–163, 2003.
[7] ——, "Discovering musical structure in audio recordings," in *Proc. Int. Conf. Music and Artificial Intelligence*, 2002, pp. 43–57.
[8] G. Peeters, A. L. Burthe, and X. Rodet, "Toward automatic music audio summary generation from signal analysis," in *Proc. Int. Conf. Music Information Retrieval*, 2002, pp. 94–100.
[9] G. Peeters and X. Rodet, "Signal-based music structure discovery for music audio summary generation," in *Proc. Int. Computer Music Conference*, 2003, pp. 15–22.
[10] J. T. Foote and M. L. Cooper, "Media segmentation using self-similarity decomposition," in *Proc. SPIE Storage and Retrieval for Media Databases*, vol. 5021, 2003, pp. 167–175.
[11] M. Cooper and J. Foote, "Summarizing popular music via structural similarity analysis," in *Proc. IEEE Workshop Applications of Signal Processing to Audio and Acoustics*, 2003, pp. 127–130.
[12] W. Chai and B. Vercoe, "Structural analysis of musical signals for indexing and thumbnailing," in *Proc. ACM/IEEE Joint Conf. Digital Libraries*, 2003, pp. 27–34.
[13] J. Wellhausen and H. Crysandt, "Temporal audio segmentation using mpeg-7 descriptors," *Proc. SPIE Storage and Retrieval for Media Databases*, vol. 5021, pp. 380–387, 2003.
[14] J.-J. Aucouturier and M. Sandler, "Finding repeating patterns in acoustic musical signals: Applications for audio thumbnailing," in *Proc. AES 22nd Int. Conf. Virtual, Synthetic and Entertainment Audio*, 2002, pp. 412–421.
[15] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC music database: Popular, classical, and jazz music databases," in *Proc. Int. Conf. Music Information Retrieval*, 2002, pp. 287–288.
[16] M. Goto, "Development of the RWC music database," in *Proc. Int. Congr. Acoustics*, 2004, pp. I-553–I-556.
[17] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
[18] R. N. Shepard, "Circularity in judgments of relative pitch," *J. Acoust. Soc. Amer.*, vol. 36, no. 12, pp. 2346–2353, 1964.
[19] B. C. J. Moore, *An Introduction to the Psychology of Hearing*, 4th ed. New York: Academic, 1997.
[20] W. Fujisaki and M. Kashino, "Basic hearing abilities and characteristics of musical pitch perception in absolute pitch possessors," in *Proc. Int. Congr. Acoustics*, 2004, pp. V-3607–V-3610.
[21] G. H. Wakefield, "Mathematical representation of joint time-chroma distributions," *Proc. SPIE*, pp. 637–645, 1999.
[22] H. Yamada, M. Goto, H. Saruwatari, and K. Shikano, "Multi-timbre chord classification for musical audio signals (in Japanese)," in *Proc. Autumn Meeting Acoustical Soc. Japan*, Sep. 2002, pp. 641–642.
[23] ——, "Multi-timbre chord classification method for musical audio signals: Application to musical pieces (*in Japanese*)," in *Proc. Spring Meeting Acoustical Soc. Japan*, Mar. 2003, pp. 835–836.
[24] T. Fujishima, "Realtime chord recognition of musical sound: A system using common lisp music," in *Proc. Int. Computer Music Conf.*, 1999, pp. 464–467.
[25] A. Sheh and D. P. Ellis, "Chord segmentation and recognition using EM-trained hidden Markov models," in *Proc. Int. Conf. Music Information Retrieval*, 2003, pp. 183–189.
[26] T. Yoshioka, T. Kitahara, K. Komatani, T. Ogata, and H. G. Okuno, "Automatic chord transcription with concurrent recognition of chord symbols and boundaries," in *Proc. Int. Conf. Music Information Retrieval*, 2004, pp. 100–105.
[27] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, 1964.
[28] C. J. van Rijsbergen, *Information Retrieval*, 2nd ed. London, U.K.: Butterworths, 1979.
[29] M. Goto, R. Neyama, and Y. Muraoka, "RMCP: Remote music control protocol, design and applications," in *Proc. Int. Computer Music Conf.*, 1997, pp. 446–449.
[30] K. Hirata and S. Matsuda, "Interactive music summarization based on GTTM," in *Proc. Int. Conf. Music Information Retrieval*, 2002, pp. 86–93.

Masataka Goto received the Doctor of Engineering degree in electronics, information, and communication engineering from Waseda University, Tokyo, Japan, in 1998.

He then joined the Electrotechnical Laboratory (ETL; reorganized as the National Institute of Advanced Industrial Science and Technology (AIST) in 2001), Tsukuba, Ibaraki, Japan, where he has since been a Research Scientist. He served concurrently as a Researcher in Precursory Research for Embryonic Science and Technology (PRESTO), Japan Science and Technology Corporation (JST) from 2000 to 2003, and an Associate Professor of the Department of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, University of Tsukuba, since 2005. His research interests include music information processing and spoken-language processing.

Dr. Goto is a member of the Information Processing Society of Japan (IPSJ), Acoustical Society of Japan (ASJ), Japanese Society for Music Perception and Cognition (JSMPC), Institute of Electronics, Information, and Communication Engineers (IEICE), and the International Speech Communication Association (ISCA). He has received 17 awards, including the IPSJ Best Paper Award and IPSJ Yamashita SIG Research Awards (special interest group on music and computer, and spoken language processing) from the IPSJ, the Awaya Prize for Outstanding Presentation and Award for Outstanding Poster Presentation from the ASJ, Award for Best Presentation from the JSMPC, Best Paper Award for Young Researchers from the Kansai-Section Joint Convention of Institutes of Electrical Engineering, WISS 2000 Best Paper Award and Best Presentation Award, and Interaction 2003 Best Paper Award.