Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology (UIST 2003), pp.31-40, November 2003.

# SmartMusicKIOSK: Music Listening Station with Chorus-Search Function

*Masataka Goto*
“Information and Human Activity,” PRESTO, Japan Science and Technology Corporation (JST). /
National Institute of Advanced Industrial Science and Technology (AIST).
1-1-1 Umezono, Tsukuba, Ibaraki 305-8568, JAPAN.
m.goto@aist.go.jp

## ABSTRACT

This paper describes a new music-playback interface for trial listening, *SmartMusicKIOSK*. In music stores, short trial listening of CD music is not usually a passive experience — customers often search out the chorus or “hook” of a song using the fast-forward button. Listening of this type, however, has not been traditionally supported. This research achieves a function for jumping to the chorus section and other key parts of a song plus a function for visualizing song structure. These functions make it easier for a listener to find desired parts of a song and thereby facilitate an active listening experience. The proposed functions are achieved by an automatic chorus-section detecting method, and the results of implementing them as a listening station have demonstrated their usefulness.

**KEYWORDS:** music-playback interface, music interaction, chorus detection, song structure, audio visualization

## 1 INTRODUCTION

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, CD stores in recent years have come to install *music listening stations* to allow customers to listen to CDs on a trial basis to facilitate a decision on purchasing. In general, the main objective of listening to music is to appreciate it, and for this reason, 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, and because of this time constraint, the above manner of listening to full selections is rare here. In the case of popular music, for example, customers often want to listen to the most representative, uplifting part of a song, i.e., the *chorus* or *refrain*, to pass judgment on that song.[1] 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 pushing the fast-forward button repeatedly, eventually finding it and listening to it.

The functions provided by conventional listening stations for music CDs, however, do not support this unique way of trial listening. 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, digital listening stations that have recently come to be installed in CD stores enable playback of several hundred thousands of musical selections stored in MP3 or other compression formats from a hard disk or over the network. Here, however, only the beginning of each musical selection (an interval of about 45 seconds) is mechanically excerpted and stored, which means that a trial listener may not necessarily hear the chorus section.[2]

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 searching for the chorus by oneself. SmartMusicKIOSK also provides a function for jumping to the beginning of the next structural section of the song by either estimating or preparing beforehand other repeated sections in addition to chorus sections. For example, using this function on a song structure like “intro ⇒ (verse A ⇒ verse B ⇒ chorus) × 2 ⇒ chorus” would enable the trial listener to jump freely to the beginning of those verse-A or

[1] This is influenced by the common practice of playing the chorus section of songs when introducing a pop-music hit chart on a music program or when using music in commercial messages in broadcasting.

[2] In this regard, it has been said that songs that begin with the chorus are on the increase in Japan’s popular music world. To check the validity of this belief, we conducted a survey on Japan’s popular-music hit chart (top 20 singles ranked weekly from January to December 2001) and found that only about 20% of those songs featured a chorus that begin within 40 seconds of the start of the song.

chorus sections.

Much research has been performed in the field of music information processing especially in relation to music information retrieval [16, 17, 18, 25] and music understanding [3, 5, 6, 7, 8, 10, 22, 27], 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.

This paper is organized as follows. First, Section 2 discusses past interaction formats for playing back music and Section 3 describes the overall configuration of the new SmartMusicKIOSK listening station. Section 4 then proposes an automatic chorus-section detecting method using musical audio signals for achieving a chorus-search function and describes it in detail. Next, Section 5 shows experimental results of evaluating this chorus-section detecting method and testing the music listening station we implemented with the method. Finally, Section 6 discusses related research and applications and Section 7 summarizes this paper's contributions.

## 2 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 non-real-time interaction. The ability of a listener to play back music interactively really only began with the coming of technology for recording music onto magneto-optical 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 listing 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 just after releasing the button
3. Display of elapsed time from the start of the selection in question.

Consequently, if a listener wanted to listen to the chorus of a song, for example, the listener 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-position "slider" may be provided. The total length of such a slider corresponds to the length of a song, and the listener can manipulate the slider lever to jump to any position in a song understanding that the position of the lever corresponds to some percentage of total elapsed time. Here as well, however, the listener must use manual means to search out a specific playback position, so nothing has really changed.

## 3 INTELLIGENT MUSIC LISTENING STATION: SmartMusicKIOSK

The SmartMusicKIOSK music listening station has been developed to support interactive specification of playback position and thereby solve the above problem. Specifically, 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. The following two methods are proposed to solve this problem assuming the main target to be popular music.

1. *Automatic jumping to the beginning of sections relevant to a song's structure ("jump to chorus" function)*

   The structure of a song is analyzed beforehand and functions are provided enabling automatic jumping to sections that would be of interest to listeners. These functions are "jump to chorus," "jump to previous section in song," and "jump to next section in song." With these functions, a listener can directly jump and listen to chorus sections, or jump to the previous or next section of the song.

2. *Visualization of song contents ("music map" function)*

   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 Figure 1. From this display, the positional relationship among the intro, verse A, verse B, chorus, and other sections of the song can be interpreted. While examining this display, the listener can use the automatic jump buttons, the usual fast-forward/rewind buttons, or playback slider to move to any point of interest in the song.

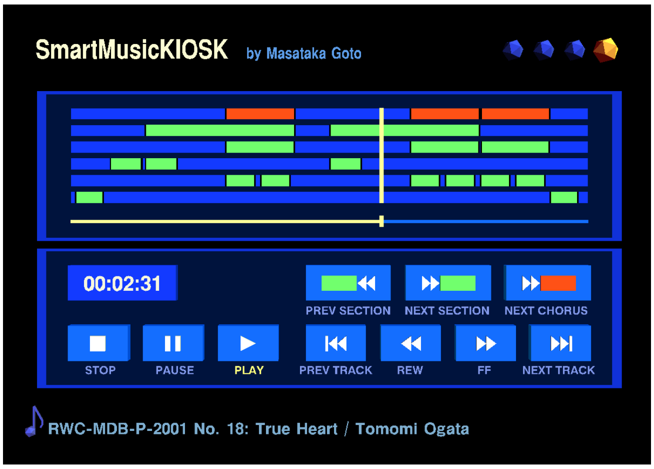

Figure 1: SmartMusicKIOSK screen display. The lower window presents the playback-operation buttons and the upper window provides a visual representation of a song's contents (results of automatic chorus-section detection using RWC Music Database [9] RWC-MDB-P-2001 No. 18). The horizontal axis of the upper window is the time axis covering the entire song; the top row shows chorus sections, the five lower rows show repeated sections, and the bottom horizontal bar is a playback slider.

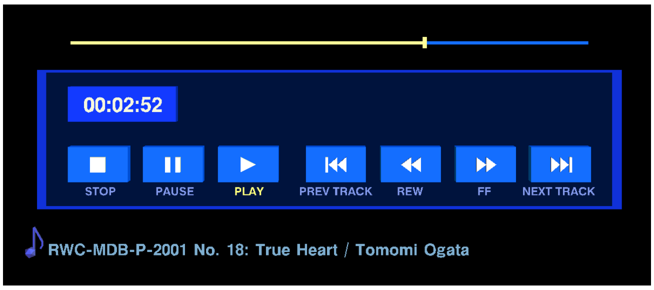

Figure 2: Screen interface as used by a conventional media player. The lower window displays the playback-operation buttons and the upper horizontal bar is a playback slider.

Figure 1 also shows the screen display of the SmartMusicKIOSK interface. For the sake of comparison, Figure 2 shows the interface screen excluding the newly proposed functions so that only the seven standard playback-operation buttons remain. These seven buttons are named, from left to right, STOP, PAUSE, PLAY, PREV TRACK, REW, FF, and NEXT TRACK, each marked by the customary symbol for the function in question. In addition, elapsed time from the beginning of the song is displayed above the STOP button. The following describes the lower and upper windows shown in Figure 1.

- *Playback operation window* (lower window)

  The three automatic jump buttons added to those of Figure 2 are named, from left to right:
  - PREV SECTION
  - NEXT SECTION
  - NEXT CHORUS

  corresponding to the functions of "jump to previous section in song," "jump to next section in song," and "jump to chorus." 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. A chorus is generally repeated several times in a song, and the system will jump to the next chorus every time this button is pressed. 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.[3] On each row, colored sections indicate similar (repeated) sections. In Figure 1, for example, the second row from the top indicates the structural repetition of "verse A ⇒ verse B ⇒ 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 first listens to the intro of a song for just a short time and then jumps and listens to the chorus with just a push of a button.[4] Furthermore, by visualizing the entire structure of a song, the listener can choose various parts of a song for trial listening.

## 4 METHOD FOR ACHIEVING SmartMusicKIOSK

To achieve the SmartMusicKIOSK automatic-jump and visualization functions described above, descriptions of the chorus sections and repeated sections in each song are essential. Furthermore, to accommodate a large number of songs in actual operation, these descriptions must be obtained in an au-

[3] One of the lower rows is usually equivalent to the top row because chorus sections are selected from groups of repeated sections as described in Section 4.2.

[4] 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.

tomated manner. It is difficult, however, to obtain such descriptions from complex real-world audio signals of music, and we cannot use existing techniques for this purpose.

We therefore propose a method, called *RefraiD* (Refrain Detecting Method), that automatically detects the beginning and end points of chorus sections and repeated sections in a song with a focus on popular music. Although other chorus-detection methods have been reported [1, 2, 19], they only extract a single segment from several chorus sections by detecting a repeated section of a designated length as the most representative of a song. None of these previous methods addressed the problem of detecting all the chorus sections in a song and identifying both ends of those chorus sections. Furthermore, while chorus sections are sometimes modulated (the key is changed) during their repetition in a song, the previous methods were not able to deal with modulated repetition. The RefraiD, on the other hand, makes it possible to detect all repeated chorus sections in a song and to estimate their beginning and end points by examining the mutual relationships among various repeated sections. This method also incorporates a new similarity measure that enables detection of a repeated chorus section even after modulation.

Although the results of automatic detection here may include some errors and are therefore not 100% accurate, 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. For this reason, we developed an editor enabling manual labeling and correction of chorus sections and of the song structure on the whole. Manual labeling is a useful tool for songs not suitable for automatic detection or outside the category of popular music.

### 4.1 Problems in chorus-section detection

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 similarity between the acoustic features obtained from each section. In this process, similarity must be high between acoustic features even if the accompaniment or melody line changes somewhat in the repeated section. 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.

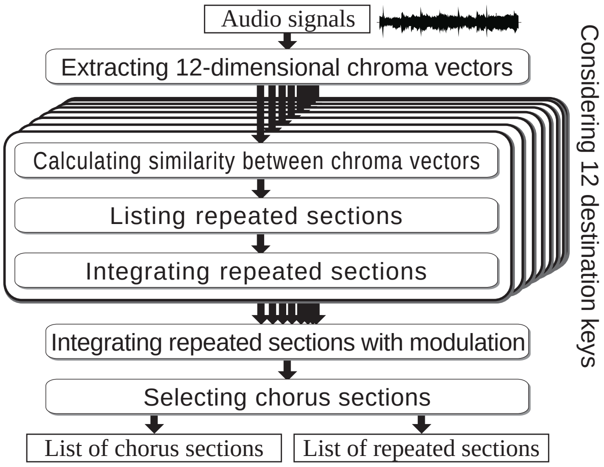
Audio signals
Extracting 12-dimensional chroma vectors
Calculating similarity between chroma vectors
Listing repeated sections
Integrating repeated sections
Integrating repeated sections with modulation
Selecting chorus sections
List of chorus sections
List of repeated sections
Considering 12 destination keys

Figure 3: Overview of chorus-section detecting method *RefraiD*.

*Problem 2: Repetition judgment criterion*

The criterion establishing just how high similarity must be for repetition to occur depends on the song. It could be easily set for a small number of specific songs by manual means. For a large song set, however, this criterion would have to be automatically modified based on the song being processed.

*Problem 3: Estimating both ends (beginning and end points) of repeated sections*

The beginning and end points of repeated sections must be estimated by examining the mutual relationships among the various repeated sections. For example, given the 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. The both ends of the C section in (B C) could be inferred, however, from the information obtained on 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 in the latter half of a song.

### 4.2 Overview of the chorus-section detecting method: RefraiD

Figure 3 shows the process flow of RefraiD, a chorus-section detecting method that solves the problems described above.

1. *Extract 12-dimensional chroma vectors and calculate similarity*

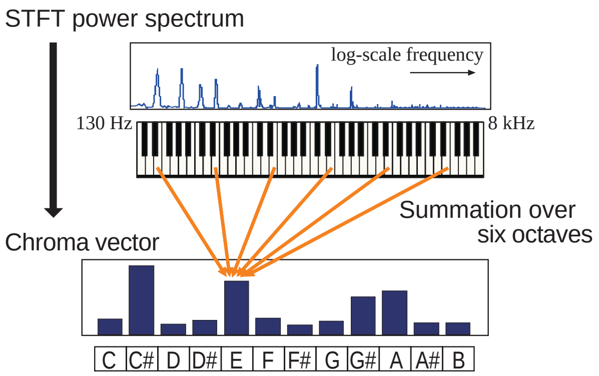

Figure 4: Overview of calculating 12-dimensional chroma vector.

The method first extracts a 12-dimensional feature vector called a *chroma vector*, which is robust to small changes of accompaniments, from each frame of an input audio signal and calculates the similarity between these vectors *(solution to Problem 1)*. Figure 4 shows an overview of calculating the chroma vector. The 12-dimensional *chroma vector* $\vec{v}(t)$ is extracted from the power 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 $v_c(t)$:

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

The $BPF_{c,h}(f)$ is a bandpass filter that passes the signal at the log-scale frequency $F_{c,h}$ (in cents[5]) of pitch class $c$ in octave position $h$[6]

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

and is defined using a Hanning window as follows:

$$BPF_{c,h}(f) = \frac{1}{2}\left(1 - \cos\frac{2\pi(f - (F_{c,h} - 100))}{200}\right). \qquad (3)$$

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-bits/16-kHz, and then the STFT with a 4096-sample Hanning window is calculated by 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 Equation (1), are respectively 3 and 8. This covers six octaves (130 Hz to 8 kHz).

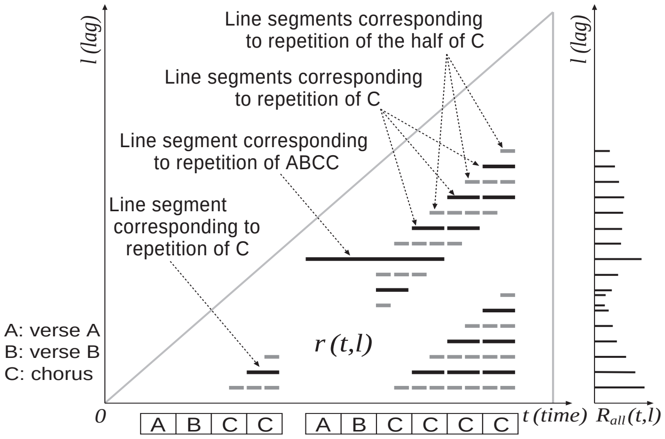

Figure 5: A plot of line segments, the similarity $r(t,l)$, and the possibility $R_{all}(t,l)$ of containing line segments. The similarity $r(t,l)$ is defined in the right-angled isosceles triangle in the lower right-hand corner. The actual $r(t,l)$ is noisy and ambiguous and usually contains many line segments irrelevant to chorus sections.

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

$$r(t,l) = 1 - \frac{\left| \frac{\vec{v}(t)}{\max_c v_c(t)} - \frac{\vec{v}(t-l)}{\max_c v_c(t-l)} \right|}{\sqrt{12}}, \qquad (4)$$

where $l$ ($0 \leq l \leq t$) is the lag. Since the denominator $\sqrt{12}$ is the length of the diagonal line of the 12-dimensional hypercube with edge length 1, $r(t,l)$ satisfies $0 \leq r(t,l) \leq 1$.

2. *List repeated sections*

The method then lists pairs of repeated sections by using an adaptive repetition-judgment criterion that is configured by an automatic threshold selection method based on a discriminant criterion [20] *(solution to Problem 2)*. Pairs of repeated sections are obtained from $r(t,l)$. Considering that $r(t,l)$ is drawn within the right-angled isosceles triangle in the two-dimensional time-lag space as shown in Figure 5, 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 the time $T1$ 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 the repetition of) the section $[T1 - L1, T2 - L1]$. In other words, a line segment indicates a repeated-section pair.

To find $(t = [T1, T2], l = L1)$ in $r(t,l)$, the possibility of containing line segments at the lag $l$, $R_{all}(t,l)$, is evaluated at the current time $t$ (e.g., at the end of the song) as follows (Figure 5):

$$R_{all}(t,l) = \int_l^t \frac{r(\tau,l)}{t-l}\, d\tau. \qquad (5)$$

Before this calculation, $r(t,l)$ is normalized by subtracting the mean of $r(t,l)$ in the adjacent area.

[5] Frequency $f_{\mathrm{Hz}}$ in hertz is converted to frequency $f_{\mathrm{cent}}$ in cents so that there are 100 cents to a tempered semitone and 1200 to an octave: $f_{\mathrm{cent}} = 1200 \log_2(f_{\mathrm{Hz}} \,/\, (440 \times 2^{\frac{3}{12}-5}))$.

[6] In the Shepard's helix representation of pitch perception [23], $c$ and $h$ respectively correspond to *chroma* and *height*.

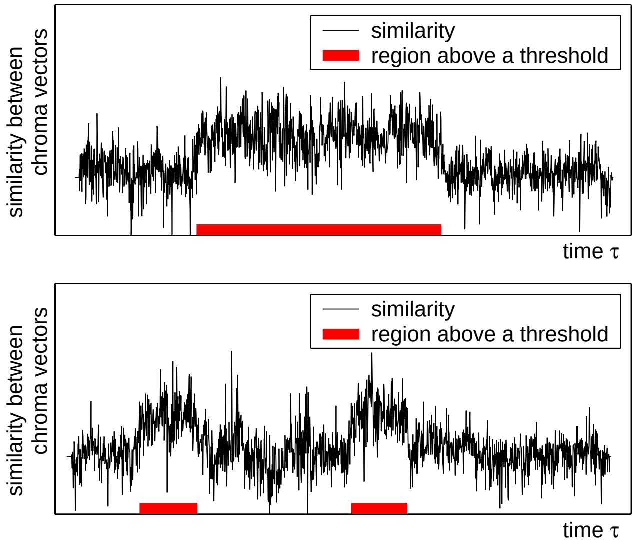

Figure 6: Examples of the similarity $r(\tau, L1)$ at high-peak lags $L1$. The bottom horizontal bars indicate the regions above an automatically adjusted threshold, which means they correspond to line segments.

The method then picks up high peaks above a threshold $Th_R$ of $R_{all}(t,l)$ to search for line segments after smoothing $R_{all}(t,l)$ by using a moving average filter. Because the threshold $Th_R$ is closely related to the repetition-judgment criterion that should be adjusted for each song, we use an automatic threshold selection method based on a discriminant criterion [20]. When dichotomizing the peak heights into two classes by a threshold, the optimal threshold is obtained by maximizing the discriminant criterion measure that is defined by the following between-class variance:

$$\sigma_B^2 = \omega_1 \omega_2 (\mu_1 - \mu_2)^2, \tag{6}$$

where $\omega_1$ and $\omega_2$ are the probabilities of class occurrence (number of peaks in each class / total number of peaks), and $\mu_1$ and $\mu_2$ are the mean of peak heights in each class.

The line segments are finally searched in the direction of the horizontal time axis on the one-dimensional function $r(\tau, L1)$ $(L1 \le \tau \le t)$ at the lag $L1$ of each high peak. After smoothing $r(\tau, L1)$ by using a moving average filter, the method obtains line segments on which the smoothed $r(\tau, L1)$ is above a threshold (Figure 6). This threshold is also adjusted by using the automatic threshold selection method.

3. *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. At this time, the method redetects line segments that were missed in bottom-up detection using information on other line segments. For example, even if two instances of the line segment corresponding to repetition of C are not obtained on the long line segment corresponding to repetition of ABCC in Figure 5, these locations can be expected to be detected at this step. In this way, both ends of sections in each group can be revised appropriately *(solution to Problem 3)* and obtained line segments can be validated.

4. *Integrate repeated sections with modulation*

Denoting the chroma vector of a certain performance as $\vec{v}(t)$ and the chroma vector of the performance modulated by a semitone *tr* as $\vec{v}(t)'$, each dimension of these vectors corresponds to a pitch class. Accordingly, the values resulting from shifting each dimension of $\vec{v}(t)'$ by modulation width *tr* come to approximate the values of $\vec{v}(t)$ before modulation ($\vec{v}(t) \doteqdot S^{tr}\vec{v}(t)'$). This shift operation[7] can be expressed by a shift matrix $S$ as follows.

$$S = \begin{pmatrix} 0 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \\ 1 & 0 & \cdots & \cdots & \cdots & 0 \end{pmatrix} \tag{7}$$

Using this feature of chroma vectors and considering 12 destination keys, 12 types of similarity $r_{tr}(t,l)$ corresponding to each *tr* can be redefined as follows.

$$r_{tr}(t,l) = 1 - \frac{\left| \frac{S^{tr}\vec{v}(t)}{\max_c v_c(t)} - \frac{\vec{v}(t-l)}{\max_c v_c(t-l)} \right|}{\sqrt{12}} \tag{8}$$

As a starting point, detection of repeated sections as described earlier can be performed for the above 12 types of similarity followed by the combining of all repeated sections *(solution to Problem 4)*.

5. *Select chorus sections*

Each group $i$ of repeated sections is evaluated for their possibility $\nu_i$ of being chorus sections and the group for which $m$ = argmax$_i$ $\nu_i$ is taken to be chorus sections. Chorus possibility $\nu_i$ is defined as the summation of reliability (local chorus possibility) $\lambda_{ij}$ determined for each of $M_i$ sections $j$ within group $i$, with this summation weighted by section length $L_i$ as shown below (constant $\mathrm{D_{len}}$ =1.4 sec).

$$\nu_i = \left( \sum_{j=1}^{M_i} \lambda_{ij} \right) \log \frac{L_i}{\mathrm{D_{len}}} \tag{9}$$

Here, $\lambda_{ij}$ is the average of similarity $r_{tr}(t,l)$ in the corresponding line segments and is modified so that those items that satisfy the following three assumptions take on a higher value.

*Assumption 1:* Chorus has an appropriate, allowed range (7.7 to 40 sec in the current implementation).

*Assumption 2:* When there is a repeated section that is long enough to likely correspond to the repetition of a long section like (verse A ⇒ verse B ⇒ chorus) × 2, the chorus section is likely to be at the end of that repeated section.

[7] Note that this operation is not applicable to other acoustic features such as simple power spectrums and MFCC features.

*Assumption 3:* Because a chorus section tends to have two half-length repeated sub-sections within its section, a section having those sub-sections is likely to be the chorus section.

RefraiD outputs a list of chorus sections found in the above way as well as a list of repeated sections obtained as its intermediate result.

## 5 SYSTEM IMPLEMENTATION AND RESULTS

We built a SmartMusicKIOSK system incorporating all the functions described in Section 3. The system is executed with files that include descriptions of chorus sections and repeated sections, which can be obtained by the RefraiD chorus-section detecting method. While various sets of repeated sections were obtained as results of automatic detection, the current implementation uses only the top five levels of long repeated sections for jumping and visual representations.

In this system, the song-file playback engine, 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 *RACP (Remote Audio Control Protocol)*, which we have designed to enable efficient sharing of audio signals and various types of control information. This protocol is an extension of RMCP [11] enabling the transmission of audio signals.

Figure 7 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 could be operated by touching the screen with a pen and by pushing the keys of an external keypad (center-right of the photograph) that duplicates the playback-button group shown on the screen.

The following first describes the results of evaluating the RefraiD chorus-section detecting method and then the results of running the SmartMusicKIOSK system.

### 5.1 Evaluation of the automatic chorus-section detecting method

We evaluated the accuracy of chorus-section detection obtained by the RefraiD method. The method was tested on 100 songs of the popular-music database *"RWC Music Database: Popular Music"* (RWC-MDB-P-2001 Nos. 1–100) [9], 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 are right or wrong, correct chorus sections in targeted songs had to be labeled manually. To enable this task, we developed a song-structure labeling editor that can divide up a song and correctly label chorus sections and repeated sections. This editor can also perform manual labeling of the kind described at the beginning of Section 4.

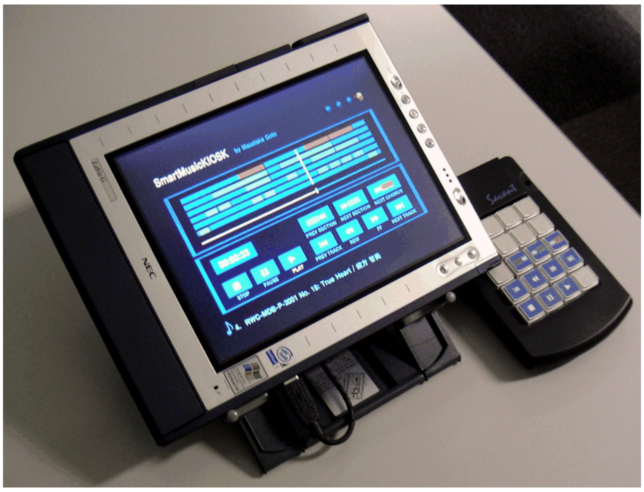

Figure 7: Demonstration of SmartMusicKIOSK implemented on a tablet PC.

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 by using the F-measure [26], which is the harmonic mean of the recall rate ($R$) and the precision rate ($P$):

$$\text{F-measure} = \frac{2RP}{R+P} \qquad (10)$$

$$R = \frac{\text{total length of correctly detected chorus sections}}{\text{total length of correct chorus sections}} \qquad (11)$$

$$P = \frac{\text{total length of correctly detected chorus sections}}{\text{total length of detected chorus sections}}. \qquad (12)$$

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

The results of this evaluation revealed that 80 songs out of 100 were returned correct (with the average F-measure of those 80 songs being 0.938). The main reasons that the method made mistakes were choruses that did not repeat more times than other sections and the repetition of similar accompaniments throughout most of a song. Among these 100 songs, 10 had modulated choruses, and 9 of these could be detected correctly. In addition, 22 songs had choruses exhibiting significant changes in accompaniment or melody on repetition, and 21 of these were detected; the repeated chorus section itself was correctly detected in 16 of these. There results show that the method is robust enough to deal with real-world audio signals.

### 5.2 Results of SmartMusicKIOSK operation

We ran our SmartMusicKIOSK system under the four conditions described below corresponding to the presence or absence of each of the two proposed functions (jump buttons and song-structure display). Subjects chose songs that they had never listened to before from among the RWC Music Database mentioned above (RWC-MDB-P-2001), and for these songs, subjects could view correct structure descriptions obtained by the automatic chorus-section detecting method. Furthermore, to allow a fair comparison of operation results under these four conditions, the function that allows the listener to click directly on a section on the song-structure display for playback was not used.

The following describes these four conditions and corresponding operation results.

*Condition 1: None of the proposed functions provided (as in setup of Figure 2)*

If the chorus was not found at the beginning of the song, the listener would push the fast-forward button and then listen for a while, repeating this 5 to 10 times until the chorus appeared. This fast-forward type of operation while briefly listening to the song is time consuming and troublesome but nevertheless useful when one wants to catch the mood of a song.

*Condition 2: No jump buttons but song-structure display provided*

Listeners found this setup to be more convenient than that of Condition 1 since the song-structure display provided good feedback as to how far one should fast-forward for playback. On the other hand, by displaying points beyond the current playback position, listeners often felt frustrated that they could not jump directly to those points.

*Condition 3: Jump buttons provided but no song-structure display*

After listening to the intro of a song, the tendency here was to push the NEXT CHORUS button to go directly to a chorus section, or to push the NEXT SECTION button and listen briefly, repeating this until the chorus appeared and then listening carefully. Listeners felt that listening while jumping was efficient and preferred this setup to that of Condition 2.

*Condition 4: All proposed functions provided (as in setup of Figure 1)*

This setup combined the advantages of the setups under conditions 2 and 3 and was consequently evaluated as the most convenient by listeners. Furthermore, in addition to the manner of listening under Condition 3, there was a strong tendency to listen while moving back and forth as desired on the song structure. For example, a listener might listen to the first instance of the chorus and then return to verse A, and then jump to a repeated chorus section in the latter half of the song.

Condition 3, which corresponds to the addition of three jump buttons to an ordinary media player, was found to be more convenient than such a player despite the absence of a song-structure display. The results of Condition 4, moreover, revealed that visualization of song structure facilitated jump operations and the listening to various parts of a song. In general, listeners who had received no explanation about jump-button functions or display windows were nevertheless able to surmise their purpose in little time.

The above results demonstrate that the proposed interface works and that pushing jump buttons while receiving visual assistance from the song-structure display enables listeners to play back songs in an interactive manner. Trial-listening operation has also revealed that the proposed functions for jumping to the next chorus or to the previous or next section of a song are intuitively easy to use requiring no training.

## 6 DISCUSSION

While this research dealing with interaction in music playback has not been pursued in the past, there are various examples of research related to music visualization and music summarization. In the following, we introduce some of this research and consider how interaction in music playback need not be limited to trial-listening scenarios. We also discuss what kind of situations the proposed method could be applied to.

### 6.1 Related research

As described in Section 5.2, music visualization as in a song-structure display is effective as a guide to changing playback position during trial listening. In this regard, visualization of the information contained in music is certainly nothing new — one need only think of sheet music or piano-roll displays of MIDI data.[8] In addition, several music visualization techniques have been proposed with the objective of analyzing expression mainly in classical music performances [12, 13, 14, 24]. These techniques, however, are limited to MIDI data and cannot be applied to musical audio signals. They also cannot be used to display the structure of repeated choruses or other sections of a song as in this research, or to interactively select a location for listening while viewing the overall form of a song in popular music.

Needless to say, it is easy to display waveforms of audio signals and frequency spectrums as a visualization of musical audio signals. However, even if listeners can view such displays for an entire song, they often find it difficult to determine musical structure and to accurately judge playback position.

The research presented in this paper is related to several studies of music summarization that have recently come to light [4, 15, 21]. Music summarization aims to shorten the length

[8] A method for displaying colored regions corresponding to notes on a 2D screen with time as the horizontal axis and MIDI note numbers (usually a keyboard display) as the vertical axis.

of a song, and therefore shares one of the objectives of trial listening, that is, to listen to music in a short time. At the same time, music summarization of the types reported has not considered an interactive form of listening as taken up by our research. From the viewpoint of trial listening, the ability of a user to easily select any section of a song for listening in a true interactive fashion is extremely significant. In the following section, we continue this discussion on active listening.

### 6.2 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, musical ring tones can now be selected for cellular phones, appropriate background music can be selected in certain situations, and music can be accessed 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 2. 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., CD player) that targets only songs as operational units. In this conventional interface, songs of no interest to the listener could easily be skipped, but skipping sections of no interest within a particular song was 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 for situations that permit long-term listening as opposed to trial listening. Specifically, we have found a tendency to listen to music in a more analytical fashion compared to past forms of music appreciation when a listener can interactively change the playback position while viewing the structure of a musical piece. For example, listeners have been observed to check the kind of structure possessed by an entire piece, to listen to each section in that structure, and to compare sections that repeat. Another finding is that the visualization of a song's structure has proven to be interesting and useful for listeners who just would like to appreciate music without jumping round.

When engaging in trial listening, a listener first wants to listen to the song's chorus and then wants to know the overall mood of the song. SmartMusicKIOSK enables a listener to catch the mood of a song in a relatively short time by providing functions for jumping to the beginning of sections that repeat in the song. It does not, however, allow jumping to the beginning of non-repeated sections (e.g., interludes, guitar solos). Extension to an "interface that makes it easy to understand the mood of a song" is left as a future research topic.

### 6.3 Applications

In addition to the use described in this paper, the SmartMusicKIOSK functions and the RefraiD chorus-section detecting method have a potentially wide range of application. The following presents key application examples.

- *Digital listening station*

  A digital listening station as introduced in Section 1 mechanically excerpts and stores only the initial sections (intros) of songs.[9] Applying the automatic chorus-section detecting method would enable a digital listening station to excerpt and store chorus sections. 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 just the beginning of a chorus (preview) 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 treated as a music version of the image thumbnail, and it can utilize front portions of choruses extracted by the chorus-section detecting method.

- *Computer-based media players*

  Media players have recently come to add a variety of functions such as exchangeable appearance (skins) and music-synchronized animation in the form of geometrical drawings that move synchronously with waveforms and frequency spectrums during playback. No essential progress, however, has been seen in the interface itself. We hope not only that our interface will be adopted by various media players, but also that other approaches of reexamining the entire functional makeup of music-playback interfaces will follow.

## 7 CONCLUSION

In this paper, we have examined the concept of interactive interfaces for trial listening of music and have proposed *SmartMusicKIOSK* as a music listening station that makes it easy for listeners to play an active role in music playback. SmartMusicKIOSK enables the listener to jump as desired to the beginning of chorus sections and other repeated sections in a song and to simultaneously view the arrangement of those sections for the whole song. We have also proposed an automatic chorus-section detecting method called *RefraiD* to obtain such sections and have confirmed its effectiveness in our experiments using 100 songs from the RWC Music Database.

[9] In Japan, a service for making digital excerpts for on-line listening stations started in 2001.

Actual operation of a SmartMusicKIOSK system revealed that not only was it convenient for trial listening but that it also served as a totally new kind of music-playback interface providing a way of listening to music heretofore not experienced.

In the future, we plan to work on the various extensions to SmartMusicKIOSK discussed in Section 6. Future work will also include research on new directions of making interaction between people and music even more active and enriching.

## ACKNOWLEDGMENTS

The author would like to thank Hideki Asoh (National Institute of Advanced Industrial Science and Technology) for his valuable discussions.

## REFERENCES

1. Mark A. Bartsch and Gregory H. Wakefield. To catch a chorus: Using chroma-based representations for audio thumbnailing. In *Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'01)*, pages 15–18, 2001.
2. Matthew Cooper and Jonathan Foote. Automatic music summarization via similarity analysis. In *Proc. of ISMIR 2002*, pages 81–85, 2002.
3. Roger Dannenberg. Music understanding by computer. In *IAKTA/LIST International Workshop on Knowledge Technology in the Arts Proc.*, pages 41–56, 1993.
4. Roger B. Dannenberg and Ning Hu. Pattern discovery techniques for music audio. In *Proc. of ISMIR 2002*, pages 63–70, 2002.
5. Peter Desain and Henkjan Honing. *Music, Mind and Machine: Studies in Computer Music, Music Cognition and Artificial Intelligence*. Thesis Publishers, 1992.
6. Masataka Goto. A robust predominant-F0 estimation method for real-time detection of melody and bass lines in CD recordings. In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2000)*, pages II–757–760, 2000.
7. Masataka Goto. An audio-based real-time beat tracking system for music with or without drum-sounds. *J. of New Music Research*, 30(2):159–171, 2001.
8. Masataka Goto. Music scene description: Toward audio-based real-time music understanding. *J. Acoust. Soc. Am.*, 111(5, Pt.2):2349, 2002. (Invited Paper of the 143rd Meeting of the Acoustical Society of America).
9. Masataka Goto, Hiroki Hashiguchi, Takuichi Nishimura, and Ryuichi Oka. RWC music database: Popular, classical, and jazz music databases. In *Proc. of ISMIR 2002*, pages 287–288, 2002.
10. Masataka Goto and Yoichi Muraoka. A beat tracking system for acoustic signals of music. In *Proc. of ACM Multimedia 94*, pages 365–372, 1994.
11. Masataka Goto, Ryo Neyama, and Yoichi Muraoka. RMCP: Remote music control protocol — design and applications —. In *Proc. of Intl. Computer Music Conf.*, pages 446–449, 1997.
12. Rumi Hiraga. Case study: A look of performance expression. In *Proc. of IEEE Visualization 2002*, pages 501–504, 2002.
13. Rumi Hiraga, Shigeru Igarashi, and Yohei Matsuura. Visualized music expression in an object-oriented environment. In *Proc. of Intl. Computer Music Conf.*, pages 483–486, 1996.
14. Rumi Hiraga, Reiko Miyazaki, and Issei Fujishiro. Performance visualization – a new challenge to music through visualization. In *Proc. of ACM Multimedia 2002*, pages 239–242, 2002.
15. Keiji Hirata and Shu Matsuda. Interactive music summarization based on GTTM. In *Proc. of ISMIR 2002*, pages 86–93, 2002.
16. *Proc. of International Symposium on Music Information Retrieval (ISMIR 2000)*, 2000.
17. *Proc. of International Symposium on Music Information Retrieval (ISMIR 2001)*, 2001.
18. *Proc. of International Conference on Music Information Retrieval (ISMIR 2002)*, 2002.
19. Beth Logan and Stephen Chu. Music summarization using key phrases. In *Proc. of ICASSP 2000*, pages II–749–752, 2000.
20. Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Trans. SMC*, SMC-9(1):62–66, 1979.
21. Geoffroy Peeters, Amaury La Burthe, and Xavier Rodet. Toward automatic music audio summary generation from signal analysis. In *Proc. of ISMIR 2002*, pages 94–100, 2002.
22. Robert Rowe. *Machine Musicianship*. The MIT Press, 2001.
23. Roger N. Shepard. Circularity in judgments of relative pitch. *J. Acoust. Soc. Am.*, 36(12):2346–2353, 1964.
24. Sean M. Smith and Glen N. Williams. A visualization of music. In *Proc. of IEEE Visualization '97*, pages 499–503, 1997.
25. Tomonari Sonoda, Masataka Goto, and Yoichi Muraoka. A WWW-based melody retrieval system. In *Proc. of Intl. Computer Music Conf.*, pages 349–352, 1998.
26. C. J. van Rijsbergen. *Information Retrieval*. Butterworths, second edition, 1979.
27. Gerhard Widmer. In search of the horowitz factor: Interim report on a musical discovery project. In *Proc. of International Conference on Discovery Science (DS 2002)*, pages 13–21, 2002.