THE JOURNAL OF THE ACOUSTICAL SOCIETY OF KOREA Vol.22, No.4E 2003. 12. pp. 141~149

Extraction of Chord and Tempo from Polyphonic Music Using Sinusoidal Modeling

Do-Hyoung Kim[*], Jae-Ho Chung[*]
[*]DSP Lab., Dept. of Electronic Engineering, Inha University

(Received April 7 2003; revised August 22 2003; accepted October 13 2003)

Abstract

As music of digital form has been widely used, many people have been interested in the automatic extraction of natural information of music itself, such as key of a music, chord progression, melody progression, tempo, etc. Although some studies have been tried, consistent and reliable results of musical information extraction had not been achieved. In this paper, we propose a method to extract chord and tempo information from general polyphonic music signals. Chord can be expressed by combination of some musical notes and those notes also consist of some frequency components individually. Thus, it is necessary to analyze the frequency components included in musical signal for the extraction of chord information. In this study, we utilize a sinusoidal modeling, which uses sinusoids corresponding to frequencies of musical tones, and show reliable chord extraction results of sinusoidal modeling. We could also find that the tempo of music, which is the one of remarkable feature of music signal, interactively supports the chord extraction idea, if used together. The proposed scheme of musical feature extraction is able to be used in many application fields, such as digital music services using queries of musical features, the operation of music database, and music players mounting chord displaying function, etc.

Keywords: *Polyphonic music, Musical features, Chord of music, Tempo of music, Sinusoidal modeling, Automatic music transcription*

I. Introduction

Recently, digital audio application areas become largely available and its usage increases. Among those, the most vigorous aspect is the development of compression of digital audio signal, such as MP3. Many people take a pleasure in listening music of digital forms in these days. Moreover, various services providing audio contents also become more and more utilized. But, for effective and fast service, providers have to solve the problems for accurate categorization and classification of music contents. MPEG-7 has standardized common interfaces to be used for service providers and users, but the standardization of detail schemes, i.e., features of musical signal for classification and search, has not been treated[1].

There are some text-based information used for the expression of musical contents; for example, title of music, singer' s name, composer' s name, lyrics, etc., which are mainly used in current fields of music search or classification. But with currently developed technologies, it seems to be difficult to automatically and precisely describe those information by machines only. Consequently, for operation of music contents, service managers have to label them manually making the use of service inconvenient and also making the accuracy or expandability of service shortened.

More fundamental approach to these fields is the introduction of unique information included in music signal itself. They can characterize the music signal uniquely without any additive informations; for example,

Corresponding author: Do-Hyoung Kim (dokim92@chol.com)
DSP Lab., Dept. of Electronic Engineering, Inha University, #253, Yong-Hyun Dong, Nam Ku, Incheon, Korea

genre of music, mood, tempo and rhythm, melody, key and chord, etc. Among them, the most common musical feature is melody. This is due to facts that most people can easily recognize a melody of music more than any other musical features and that modern music have melodies sung by singers or played by leading instruments. The query-by-humming concept of MPEG-7 and ASA (Auditory Scene Analysis) is also originated from the melody feature based on these facts[2]. But, in case of polyphonic music, the separation of melody-leading instrument or singing voice from the whole music signal is very difficult and inaccurate so that it is hard to get reliable results. Although some recent studies have suggested melody or voice extraction schemes, their simplicities and ambiguities make them inapplicable to the music signals of published music materials[3,4].

The most effective musical information used together with melody is a chord of music. Chord or harmony (in this case, complex composite of musically related frequency components, not simple multiples of fundamental frequency) of musical signal consists of the combination of musical notes and can be heard and described by trained listener or by any experienced people. Because general music favored by most people is the tonality music, which has a progress of chord, and because chord of the tonality music follows general chord making rules, chord characteristic of music is useful in application areas where require musical features characterizing the music signals.

In general, chords can change in specific positions of music, not a random position. In other words, chord-change position gives a good clue on tempo of music. Thus, if one knows the progression of chords, he has had the tempo information obtainable from a temporal chord transition pattern, as well as the key of that music and chord progression itself.

We also considered the use of the tempo information obtained at the beginning part of music, to control the chord analysis interval performed at the rest part of music; that is the interactively supporting system of chord and tempo information extraction. Consequently, chord and tempo information together increase effectiveness and reliability of musical feature extraction system.

Methods applied to extraction of chord information are explained in chapter 2 and tempo extraction based on chord progression is mentioned in chapter 3. We will show algorithm and experimental results in chapter 4. In chapter 5, we consider some future works of this study and conclude finally.

II. Extraction of Chord Information

2.1. The Structure of Chord

Chord is generated by the combination of some musical notes and musical note also consists of some frequency components. Thus, we need to know the physical frequency of each standard musical note for understanding of chord structure. Table 1 and Fig. 1 shows the values and the pattern of frequencies corresponding to chromatic scale (note scale of semitones) system which can be seen in piano or any other melodic instruments. These frequencies can be simply written by the following equation, where f_n is frequency of n-th note. From the figure and the equation, we can recognize that musical notes have increasing frequency differences between neighboring notes as note frequency increased[5].

$$f_n = f_{n-1} \times 1.0594 \tag{1}$$

Several notes organize a chord. For example, if certain block of music has strong frequency components corresponding to note **c**, note **e**, and note **g** in Table 1, we can decide the chord of that block as chord **C**, that is, the tonic harmony of C major key. Similarly, we can find relatively strong and probable chord of certain music block, based on a priori information of frequency components as class of each chord.

In this article, for avoiding confusion, note and chord are represented by small bold letter and by capital bold letter, respectively.

2.2. Chord Extraction Using Wavelets

As mentioned in section 2.1, considering the frequency pattern of monotonically increasing scale, it is desirable

Table 1. Frequencies of notes in chromatic scale.

n	note	octave						
		1	2	3	4	5	6	7
1	c	32.703	65.406	130.81	261.63	523.25	1046.5	2093.0
2	c#(db)	34.648	69.296	138.59	277.18	554.37	1108.7	2217.5
3	d	36.708	73.416	146.83	293.66	587.33	1174.7	2349.3
4	d#(eb)	38.891	77.782	155.56	311.13	622.25	1244.5	2489.0
5	e	41.203	82.407	164.81	329.63	659.26	1318.5	2637.0
6	f	43.654	87.307	174.61	349.23	698.46	1396.9	2793.8
7	f#(gb)	46.249	92.499	185.00	369.99	739.99	1480.0	2960.0
8	g	48.999	97.999	196.00	392.00	783.99	1568.0	3136.0
9	g#(ab)	51.913	103.83	207.65	415.30	830.61	1661.2	3322.4
10	a	55.000	110.00	220.00	440.00	880.00	1760.0	3520.0
11	a#(bb)	58.270	116.54	233.08	466.16	932.33	1864.7	3729.3
12	b	61.735	123.47	246.95	493.88	987.77	1975.5	3951.1

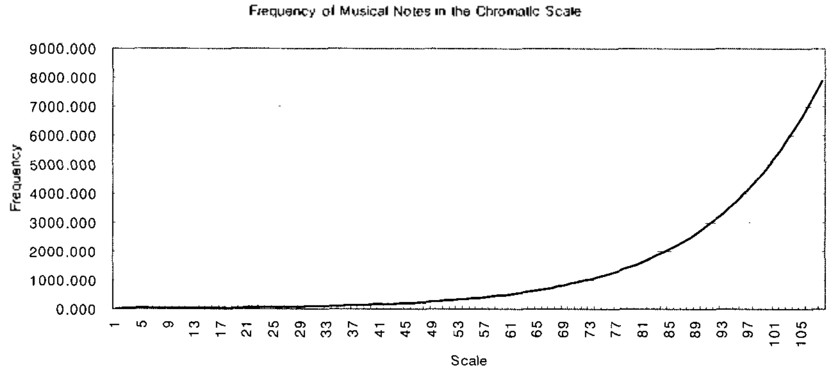


Figure 1. Increasing frequency pattern of chromatic note scale.

to apply the non-uniform analysis methods such as wavelet packet. Analysis methods using wavelet scheme are suggested in[6] and[7]. But, their performance for classification or extraction was not prominent ones. This is because the wavelet based method also can not give enough analysis resolution, especially in low frequency domain.

Fig. 2 shows an example of wavelet packet analysis of pure tone signal having single frequency component. In the case of pure tone, the energy of analyzed signal have to be concentrated on one subband. But, there happens a energy transition to neighboring band as analysis step goes on. This phenomenon makes a precise analysis of note frequency difficult.

Another defect of this method is bandwidth mismatch. Wavelet filter-banks are not adequate to divide analysis-subband particularly well matched to the bandwidth of musical scale. Generally wavelet analysis filter-bank

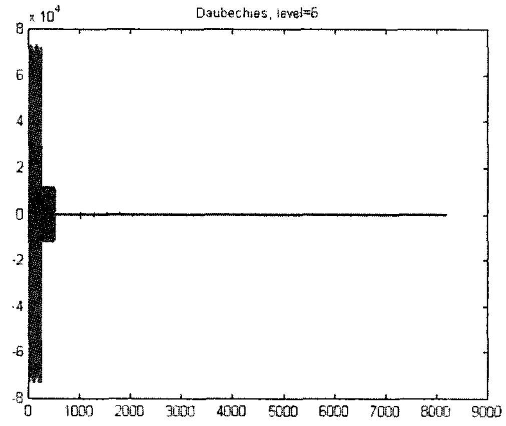


Figure 2. Example of energy transition by wavelet packet analysis.

iteratively divides certain band to a subband of half bandwidth. But, the frequency bandwidth of musical scale does not accord to half division rule. Thus, in spite of its well grounded concept, wavelet scheme is not fit to the application for musical note or chord analysis.

2.3. Chord Extraction Using Sinusoidal Modeling

Sinusoidal modeling, which represents a given signal as the sum of sinusoids, is globally used in processing or modeling of natural signal, such as speech and music[8,9]. This modeling has the same origin with Fourier analysis scheme in that they analyze a given signal using a set of sinusoidal components or similarly complex exponents. But, basis of Fourier transform is fixed as their original resolution and consequently not flexible to dynamic signal such as music. On the other hand, sinusoidal modeling does not have to fix their basis as a set of pre-defined sinusoidal function, so that it has more flexibility than Fourier analysis. In addition, since chord extraction is considered for only musically meaningful frequency components, sinusoidal modeling for corresponding component is very useful and effective.

The sinusoidal modeling scheme can be written as the following equation;

$$d(t) = \sum_{q=1}^{Q(t)} A_q(t)\cos\Theta_q(t) \tag{2}$$

where $Q(t)$ is number of partials used to represent a given signal at time t, and $A_q(t)$ and $\Theta_q(t)$ are amplitude and phase of q-th partial, respectively. For obtaining informations of individual partials, matching pursuit method based on inner-product scheme, may be useful as analysis tool. Matching pursuit algorithm firstly defines a set of basis as dictionary, D, and analyzes the signal using elements in that dictionary. In our case, dictionary is composed of a set of sinusoids which is used to express music signals. Generally, for maximizing its flexibility, matching pursuit uses very redundant dictionary of overcomplete basis. But, because only musically related sinusoids are considered now, the redundancy of dictionary and consequently its complexity can be reduced.

At analysis step of matching pursuit, algorithm performs inner product between the given signal and each element of dictionary. Let the inner product value of step n be a_n;

$$a_n = \langle s_n, g_{\gamma_0} \rangle$$

where s_n is the given signal at that step and $g_{\gamma_0} \in D$.

Then an element of maximum inner product value is selected. This element is regarded as a basis of the given signal. After the selection of one basis, product between maximum a_n and the selected element is subtracted from the given signal, resulting the residual signal;

$$s_{n+1} = s_n - a_n{}' \cdot g_{\gamma_0}{}'$$

where $a_n{}'$ and $g_{\gamma_0}{}'$ is the maximum inner product value and corresponding element at step n. Similarly, calcula tion of inner product for s_{n+1} and subtraction for s_{n+2} are performed until certain stopping criterion is satisfied.

If the dictionary of matching pursuit consists of some sinusoids and their variations, consequently the given signal is modeled by a set of sinusoids as sinusoidal modeling indicated in Eq. (2).

In our case, because only the sinusoidal components of musical notes is interesting, dictionary have to be composed by some sinusoids of frequency components described in Table 1. And Eq. (2) also have to be modified as

$$h_n(t) = \sum_{m=1}^{M} A_{m\times n_0}(t)\cos\Theta_{m\times n_0}(t) \tag{3}$$

where $h_n(t)$ is the synthesized signal using elements of dictionary corresponding to n-th note and M is a number of frequency component used to decide a note. $A_{m\times n_0}(t)$ and $\Theta_{m\times n_0}(t)$ represent amplitudes and phases of harmonic components originated in the base frequency of n-th note. As the chromatic scale has 12 notes in an octave, $n = 1,\cdots,12$.

Suppose $n=1$ is the note c as in the case of Table 1 and $|h_n(t)|$, obtained by matching pursuit algorithm, has its maximum values when n is 1, 5, and 8. Then, we can say that the chord of the given music block is most likely to chord **C**.

III. Extraction of Tempo

3.1. Tempo of Music

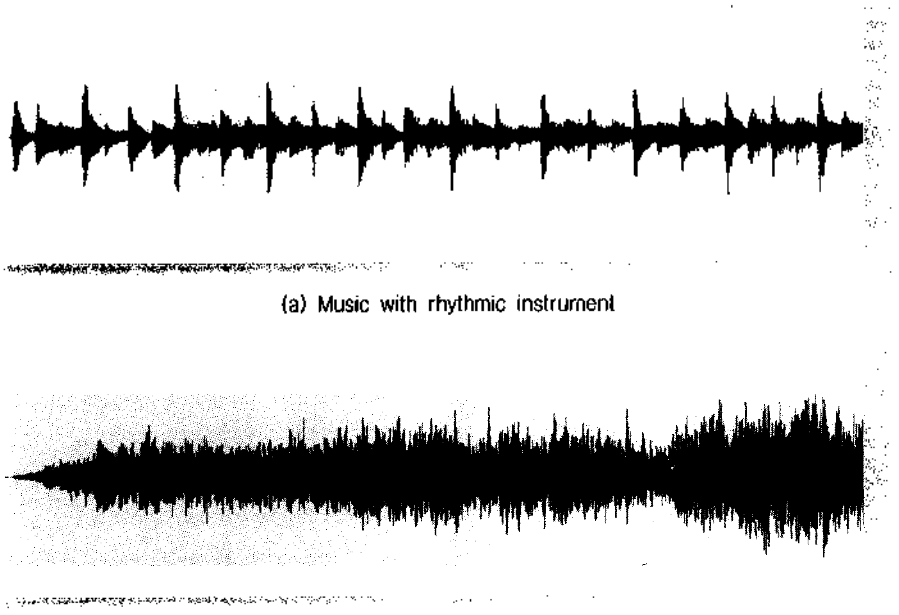
Tempo of music is commonly represented by the number of unit note played in a minute and described as ♩=60, which means that the music has a tempo of 60 quarter notes played in a minute. This information is a critical clue on the mood and the genre of music. Tempo of general tonality music is operated in the range of ♩=40 to ♩=120 having interval of 2 for slow music and 4 for fast music. If musicians are playing a music in real time, it is impossible to guarantee invariant tempo during the whole music. Although computer music programmed by machines becomes available in many fields, there also may exist an intended change of tempo such as *ritardando*. However, tempo of music which we are considering, is the representative characteristic of certain music and an available feature preserved during most part of the whole music.

Tempo is originally a temporal information, so that it is more fundamental to get it in time domain. As shown in Fig. 3(a), if the music is played by strong rhythmic instruments, temporal envelope of the waveform gives an important hint about its tempo. But, not all the music has those clear temporal pattern. Fig. 3(b), which has the same duration as Fig. 3(a), shows the case of no strong rhythmic pattern. Thus, it is difficult to apply the temporal information to the extraction of music tempo.

3.2. Extraction of Tempo in Frequency Domain

Because the sensible dependency of temporal envelope pattern to musical structure, we adopted a spectral method of tempo extraction. Chord changes occur at regular time points generally. Certain music can have combinations of many note durations, such as quarter notes or eighth notes, in one measure. Thus, individual duration of each note can hardly be obtained because of structural complexity. But, since chords are generally changed one or two times in one measure, it can be used to get tempo information of music. Based on this idea, we implemented a tempo extraction algorithm. This algorithm firstly evaluates chord change position of the given music using the chord extraction scheme mentioned in previous chapter. After certain duration for tempo extraction, the most available chord change interval is decided. Tempo is decided from a dividend of this interval.

It is more effective to induce the tempo in the beginning part of music as possible, because if we have the exact



(a) Music with rhythmic instrument

(b) Music without rhythmic instrument

Figure 3. Time pattern of music signal as the use of rhythmic instrument.

tempo information in advance, the chord search of the latter part of music can be performed only at specific positions. Suppose a tempo found in 10 seconds of beginning part is ♩ = 60 and chord change happens maximum 2 times per measure. Then it is sufficient to perform the extraction of chord every 2 seconds.

In our algorithm, the tempo of the given music signal is decided in beginning part and used as a time stamp for chord extraction in the rest part of music, so that the efficiency and accuracy of chord extraction is increased.

IV. Algorithm and Experiments

4.1. Music Samples Used in Experiments

In our experiments, we intended to include various styles of music, from classical music to modern music, with or without singing voices, played by machines (synthesizers or sound modules) or musicians, and so on.

Many recent music is made by computer programming and artificial sound source modules. It is sure that those music will reflect relatively exact frequency of note and tempo, compared to music played by musicians personally. But, we did not except real time played music in the experiment. Music samples used in experiment are of 44.1 kHz sampling frequency and of 16-bits resolution. Samples are captured from CD directly without any loss of data.

4.2. Reference Octave for Chord Extraction

Extremely high frequency notes do not contribute to chord organization compared to low frequency notes, although depending upon instruments. To decide the higher frequency limit of chord extraction, we applied many low-pass filters with various cut-off frequencies to music samples. After hearing of various low-passed samples, we have decided an upper limit of chord analysis as the lowest cut-off frequency judged to keep chord components.

Also in very low frequency area, it is difficult to discriminate neighboring notes each other because of small frequency differences. If we want to achieve chord extraction in low frequency, we have to endure large size of analysis dictionary and corresponding complexity. Considering these facts, we selected 5 octaves above a3 note in Table 1 for our experiment, that is, the notes of from 220 Hz to 7,040 Hz. In this range, the minimum frequency interval of neighboring notes is about 13 Hz, which is practically achievable.

4.3. Algorithm for Chord and Tempo Extraction

Fig. 4 shows roughly described block diagram of chord and tempo extraction algorithm. In this figure, dashed line and solid line represent a data flow mainly working in the beginning part of music and in the latter part of music, respectively. Until a tempo of music is fixed, analysis interval controller maintains short analysis interval. But, after tempo extraction is completed during the given time duration of tempo evaluation, it changes analysis interval as the length corresponding to the estimated chord change position.

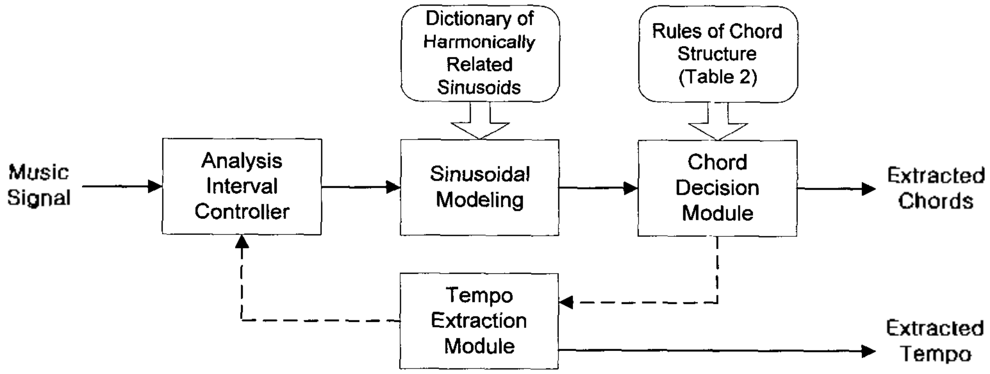

Dictionary of Harmonically Related Sinusoids
Rules of Chord Structure (Table 2)
Music Signal
Analysis Interval Controller
Sinusoidal Modeling
Chord Decision Module
Extracted Chords
Tempo Extraction Module
Extracted Tempo

Figure 4. Block diagram of chord and tempo extraction algorithm.

Tempo extraction is based on the chord change pattern and on the hypothesis about the number of chord changes in one measure; for example, 2 times of change in one measure of 4/4 (four-four time signature) music. It works for a few seconds at the beginning of music.

We assumed that the fastest tempo as ♩ = 120 and that chord can be changed maximally two times in one measure for 4/4 and one for 3/4. To distinguish a tempo, we have to take the fastest tempo into account. Because one quarter note is played every 0.5 seconds in ♩ = 120 and 0.518 seconds in ♩ = 116 (two fastest tempos used in standard music), the minimum difference of note duration must be at most 18msec. But, chord is assumed to change maximum 2 times in a measure. Thus, it is reasonable to fix the chord analysis hop size for tempo extraction as 36msec. Since hop size for the tempo extraction is much shorter than chord analysis block length (187 msec), overlap between neighboring analysis blocks is necessary and this problem affects negatively to calculation time. But if the exact tempo information is extracted in the beginning of music, chord extraction algorithm can be applied sparsely to the rest part of music, based on that information. After all, there is enough gain in calculation time. Suppose chord change happens every n-th analysis block and this music is of 4/4 time signature. Then the length of one note must be a divisor of n×36 msec. In other words, tempo is a multiple of

$$1/(n\times 36\ \text{msec/note}) = 60\times 1000/n\times 36\text{note/min}.$$

Sinusoidal modeling module analyzes a given music signal using a dictionary containing basis functions, which is, in this case, the sinusoids corresponding to the frequency components of musical notes. Because each note consists of harmonically related frequency components, dictionary have to keep these harmonic sinusoids. In this module, the sum of absolute inner product values of harmonic components, which represent the intensity of each note, $|h_n(t)|$ of Eq. (3), are calculated.

At chord decision module, a note of the maximum intensity among 12 notes of 5 octaves is selected as base note. The estimated chord is supposed to contain this note certainly. Considering the fact that chord is a long-term characteristic, we set analysis block length as 187 msec. Chord decision module decides the chord of the given block based on the base note and on chord table, which describes relations of notes consisting certain chord. Table 2 contains some chord types, such as Major, Minor, Diminished, Suspension, Augmented in triad and Dominant 7th. It also marks relative note differences of each chord type as the position of base note. Because base note can not be guaranteed to exist as first note in chord, we have to take into account situations that base exists as second or third note as indicated in the Table.

If note **g** is selected as base note and fifth note (note **c**) and ninth note (note **e**) have bigger intensities, this is most similar to Major row and third column of Table 2. Then we can conclude that the chord of given music block is chord **C**. For example, a case of accompaniments playing chord **C** and bass instrument playing note **g**.

Table 2 does not contain chord types of more complicated structures, such as 6th or any other 7th except dominant. But, the contained chord types are thought as most likely used ones in music field.

4.4. Extraction Results

Example score for chord and tempo extraction is shown in Fig. 5. This music is played by string instruments, wind instruments, percussion instrument and singer's vocal. If you want to read the chord name in this score directly, you have to find it from the combination of each note. The chord progression of this score is

GM | DM | Em | CM | GM | DM | Em | CM | GM

where first letter represents chord alphabet and second

Table 2. Interval index of each chord as the position of base note.

Symbol	Chord Type	Position of Base Note in Chord		
		1st	2nd	3rd
M	Major	4,3	3,5	5,4
m	Minor	3,4	4,5	5,3
o	Diminished	3,3	3,6	6,3
4	Suspension 4	5,2	2,5	5,5
+	Augmented	4,4	4,4	4,4
7	7th	4,3,3	3,3,2	3,2,4

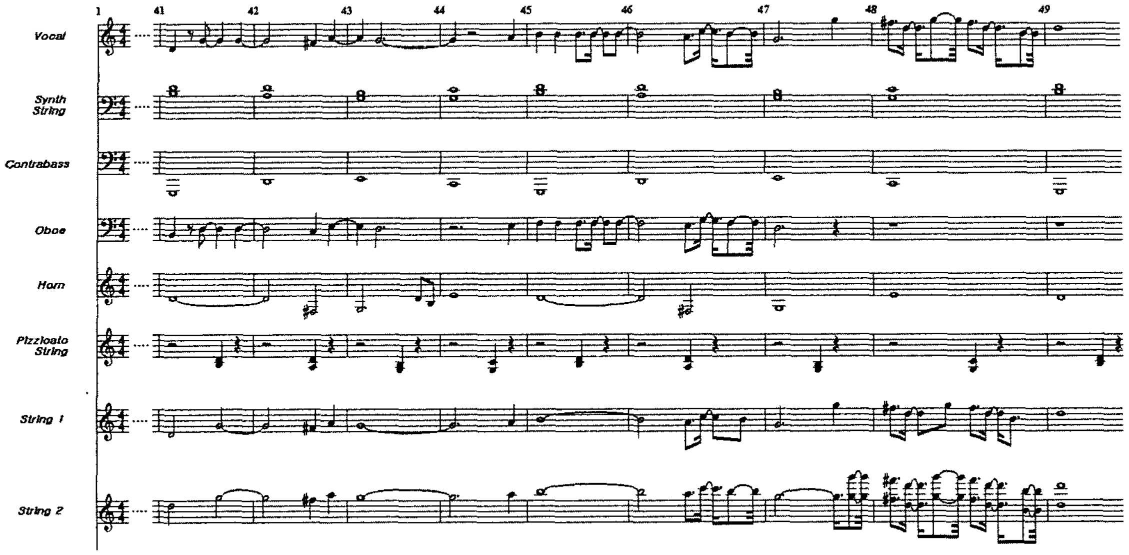


Figure 5. Example score.

letter represents chord type symbol in Table 2. Although this example score does not give any information on its tempo, we could measure it using a metronome. The measured tempo was ♩= 104.

Based on mentioned idea, we implemented a chord/ tempo extraction system using Visual Studio 6.0 in Pentium III 600 MHz PC. Because the complexity of matching pursuit is relatively high, it was not possible to implement a perfect real-time system for chord extraction in the current state. But, if computing power and resonable manipulation of matching pursuit dictionary are achieved, real-time chord displaying system will be possible.

As a result of experiment on the music shown in Fig. 5, the same chord progression above and the tempo of ♩= 104 are reported. However, repeated experiments to other music samples show that it is hard to get the accurate extraction of chord in following cases;

(a) When certain instrument of high sound intensity plays non-harmonic tones

(b) When chord of similar structure is played, such as G and Em

(c) When chord is extracted in non-stationary part due to change of tempo, etc. For example, when analysis block contains hard attack signal of drum

(d) When tonal instrument is not played. For example, only percussion instrument block or silence block

These problems are some exceptions of our idea to overcome in the future works. Because our idea supposed the exact tuned music signal, those exceptions are somewhat reasonable. But, for practical application to real field, detail refinement of our algorithm is a essential condition.

V. Conclusions and Future Works

In this paper, we proposed a method to extract chord and tempo information from digital music waveform, such as CD. This method extracts those information from the waveform directly without any additive side information. Sinusoidal modeling, which is adopted in this study, gives reasonable results of chord extraction. The results are due to the well-matched structure of chord and sinusoids. In addition, we could perform the extraction process more effectively using the tempo information analyzed in the beginning part of music. From tempo information, we decided chord extraction interval. Consequently, redundant sinusoidal analysis could be removed. These processes of musical feature extraction showed a reliable result for general tonality music though there are some deviations of extraction results depending on music style or recording condition.

This idea can be applied to the field which demands music features. The fields cover search or inquiry of music database and various electrical players to display the chord information of currently played music. For example, a user, who wants to use a digital music service through internet, can search music using the following query condition: music starting with C key or music of tempo ♩ = 92, and so forth.

We will process our study focused on three viewpoints: (1) generalization of algorithm to samples reporting inaccurate results, (2) extractions and applications of various musical features such as melody from waveform directly, (3) development of automatic music transcription system.

References

1. Martinez, Jose M. (UPM-GTI, ES), "ISO/IEC JTC1/SC29/WG11 N4980: MPEG-7 Overview (version 8)," http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm, Klangenfurt, July 2002.
2. Bergman, A. S., Auditory Scene Analysis, 4th Ed., The MIT Press, Cambridge, England, 2001.
3. McNab, R. J., and Smith, L. A. "Evaluation of a melody transcription system," *IEEE International Conference on Multimedia and Expo*, 2, 819-822, 2000.
4. Lee, T. W. and Ziehe A., "Combining time-delayed decorrelation and ICA: towards solving the cocktail party problem," *ICASSP Prog.*, 1249-1252, 1998.
5. Rossing, Thomas D., The science of sound, Addison-Wesley, 1982.
6. Su, B., and Jeng, S. K., "Multi-timbre chord classification using wavelet transform and self-organized MAP neural networks," *ICASSP Proc.*, 3377-3380, 2001.
7. Nishi, K., Ando, S., and Aida, S., "Optimum harmonics tracking filter for auditory scene analysis," *ICASSP Proc.*, 573-576, 1996.
8. Mallat, S., Zhang, Z., "Matching pursuits with time-frequency dictionaries," *IEEE-SP*, 41 (12), 3397-3415, Dec. 1993.
9. Goodwin, M., "Matching pursuits with damped sinusoids," *ICASSP Proc.*, 2037-2040, 1997.

[Profile]

- Do-Hyoung Kim

The Journal of the Acoustical Society of Korea, Vol. 22, No. 7, 2003.

- Jae-Ho Chung

The Journal of the Acoustical Society of Korea, Vol. 22, No. 7, 2003.