Multimed Tools Appl (2015) 74:7921–7942
DOI 10.1007/s11042-014-2031-1

An enhanced direct chord transformation for music retrieval in the AAC transform domain with window switching

Tai-Ming Chang · Chia-Bin Hsieh · Pao-Chi Chang

Received: 29 October 2013 / Revised: 5 March 2014 / Accepted: 14 April 2014 / Published online: 1 June 2014

Abstract With the explosive growth in the number of music albums produced, retrieving music information has become a critical aspect of managing music data. Extracting frequency parameters directly from the compressed files to represent music greatly benefits processing speed when working on a large database. In this study, we focused on advanced audio coding (AAC) files and analyzed the disparity in frequency expression between discrete Fourier transform and discrete cosine transform, considered the frequency resolution to select the appropriate frequency range, and developed a direct chroma feature-transformation method in the AAC transform domain. An added challenge to using AAC files directly is long/short window switching, ignoring which may result in inaccurate frequency mapping and inefficient information retrieval. For a short window in particular, we propose a peak-competition method to enhance the pitch information that does not include ambiguous frequency components when combining eight subframes. Moreover, for chroma feature segmentation, we propose a simple dynamic-segmentation method to replace the complex computation of beat tracking. Our experimental results show that the proposed method increased the accuracy rate by approximately 7 % in Top-1 search results over transform-domain methods described previously and performed nearly as effectively as state-of-the-art waveform-domain approaches did.

Keywords AAC · Transform domain · Chroma feature · Audio coding · Music information retrieval

1 Introduction

Music is an integral part of human entertainment and is created in diverse forms, demonstrating the ability to soothe the emotions of listeners. With the rapid development of multimedia

T.-M. Chang · C.-B. Hsieh · P.-C. Chang (✉)
Department of Communication Engineering, National Central University, Jhong-Li, Taiwan
e-mail: pcchang@ce.ncu.edu.tw

T.-M. Chang
e-mail: tmchang@vaplab.ce.ncu.edu.tw

C.-B. Hsieh
e-mail: cbhsieh@vaplab.ce.ncu.edu.tw

and the explosive growth in the amount of music produced, storing music data has evolved from Gramophone records to recent digital sample compression. Moreover, with the increasing accumulation of digital music data, music management has become a key research area.

Retrieving music information enables users to identify songs quickly that may have unknown melodies or limited music information. The method used to describe music visualization for computers is an interesting topic: Mid-level characteristics such as tone and harmony are generally considered basic features of the music representation used for distinct musical identification tasks. The most widely used feature is the pitch-class profile (PCP), also known as chroma, which forms a 12-dimensional vector to represent the intensity of 12 semitones [10]. By using the PCP extraction method, the input signal is first transformed into discrete time frequencies by using discrete Fourier transform (DFT), and the frequency bins are mapped to a 12-tone equal temperament. All of the pitch contents are folded into a single octave and comprise a chromagram. Several different chroma-based transformations similar to PCP have been proposed, such as chroma DCT-reduced log pitch (CRP) developed using pitch filters and discrete cosine transform (DCT) [16] and Constant-Q transform that generates a 36-bin chromagram [1]; all of these transformations are designed to enhance chroma representation.

Most audio files available on Internet are compressed files. In many of the well-known audio-coding techniques such as MPEG-1 layer 3 (MP3) and advanced audio coding (AAC), transform coding is used to process signals in the frequency domain. The transform coding structure can reduce data rate over ten times more than waveform coding can [12, 13]. The signals are transformed into modified DCT (MDCT) coefficients, and acoustically redundant portions are neglected to reduce date processing rate. In conventional methods used to retrieve music information, the retrieval process requires full decoding in the first step, after which the feature-extraction process operates on the decoded waveform signals to obtain the features that are then matched. Intriguingly, methods used to improve the efficiency of the coding part of the operations in modern audio encoders are similar to content analysis, despite the objectives being distinct. Modern audio coding and music representation share a similar processing step, in which signals are transformed to frequency coefficients, possibly using distinct transforms. Theoretically, if compressed files are used when the retrieval process is started, the amount of information represented in the transform domain could be as much as that in the waveform-domain, indicating that compressed signals provide adequate frequency information for retrieval. Furthermore, psychoacoustic models, which are used in modern audio codecs and can remove redundancy effectively and greatly reduce the data rate, are seldom used in conventional data-retrieval approaches. Therefore, the compression-domain approach can provide chroma features with lower computational complexity than the waveform-domain approach can. Based on this concept, several studies have suggested that the frequency information that already exists in the transform domain can be extracted directly to generate chroma features [5, 17, 19, 25]. Ravelli et al. proposed an effective method for extracting chroma features in the compression domain for MP3, AAC, and 8xMDCT [21]; the method was used at the optimal frequency resolution of MP3 and AAC (i.e., all frames were encoded using a long window), but this condition is not normal for audio codecs used in real-life applications. Furthermore, encoding the audio signal entirely by using long windows can cause the pre-echo problem and noticeable distortion at the transient part of signal volumes [28].

In this work, we investigated the influence of direct extraction of chroma features from the AAC transform domain by focusing on the impact of long/short window switching. The music-retrieval system we examined is shown in Fig. 1. In the proposed method, the frequency information in the transform domain is used directly to generate the chroma feature, avoiding the synthesis-then-analysis procedure used in conventional methods. The proposed method

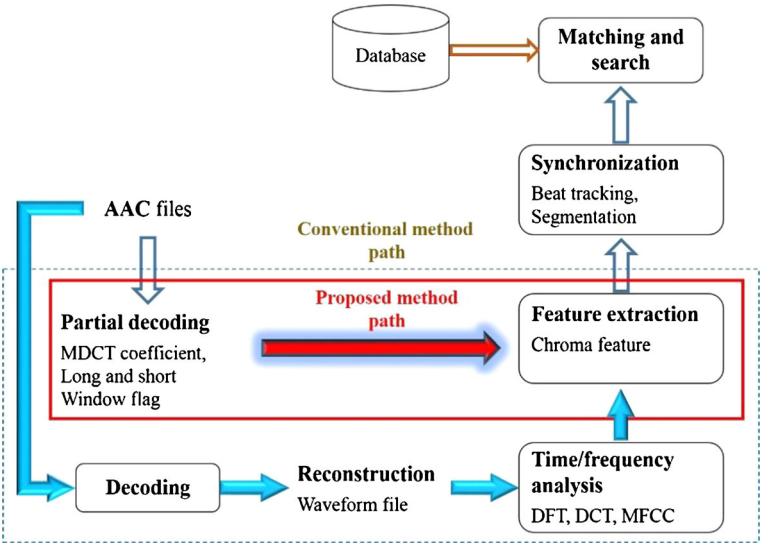

Fig. 1 Schematic diagram of music retrieval system

was designed to perform low-complexity operations and concomitantly retrieving music data as effectively as conventional methods do.

The remainder of this paper is structured as follows: Section 2 provides a brief review of work related to the MPEG-2 AAC codec. Section 3 contains the details of the proposed method, including long/short window processing, chroma mapping, and frequency selection. The criteria for evaluating music-retrieval performance are described in Section 4. Experimental analysis and comparison results are reported in Section 5, and our conclusions are presented in Section 6.

2 Analysis of AAC characteristics

2.1 MPEG-2 AAC codec

MPEG-2 AAC was established as an international standard in 1997 [13]. The aim of this development was to attain "indistinguishable" audio quality at data rates of 320 kbit/s for five full-bandwidth channel audio signals. To this end, AAC integrates the coding efficiency of a high-resolution filter bank, prediction techniques, and Huffman coding to achieve broadcast-quality audio at extremely low data rates. AAC is still considered the state-of-the art scheme both for compression and for quality of audio coding.

For the AAC decoding functions, Tsi and Liu analyzed the hardware; this is summarized in Table 1 [27]. The filter bank consists of the inverse-MDCT (IMDCT) operation, windowing, and overlap, which account for more than 70 % of the computational complexity in the decoding flow. This part can be omitted using a compression-domain approach. By contrast, in a decode-then-extract approach, although the IMDCT operation adopts a fast Fourier transform (FFT) architecture to reduce computational complexity to $O(N\times\log_2 N)$, the total complexity of traditional chroma mapping in frequency transformation is the sum of IMDCT and FFT: $O(N\times\log_2 N)$ plus $O(N/2\times\log_2 N)$. This consumption is a heavy load both for large databases and for multitudinous queries for a music retrieval system.

Table 1 Complexity analysis of AAC decoder

Tools	Complexity
Huffman	19.6 %
IQ	1.6 %
Rescale	1.9 %
Stereo	2.7 %
TNS	0.6 %
Inverse MDCT	73.6 %
Total	100 %

Window switching is a critical factor for audio quality. To consider both coding efficiency and compression quality, the AAC encoder provides long and short windows that contain 2,048 and 8×256 samples, respectively. The window decision is based on the status of the current frame, which may be either steady state or transient. For the transient frame, eight short windows are selected as the optimal compromise between frequency selectivity and pre-echo suppression at low data rates. Figure 2 shows the variation in window shape for a transient condition that consists of long_stop, eight short, and long_start windows.

2.2 Frequency resolution and time resolution

The objective of audio coding is reducing data rates for audio files while preserving audio quality. In addition to the coding technique used, the sampling rate strongly affects audio quality. High sampling rates generally benefit audio performance but may not be necessary for retrieving music information. Chroma feature extraction at high sampling frequency may generate too many features to be matched and redundant high-frequency components. Chroma features are extracted primarily from the fundamental frequency of notes in a short period. However, musicians rarely play notes on musical instruments that are higher than the eighth octave (approximately 8 kHz). Thus, high-frequency components are discarded commonly during data retrieval to eliminate unnecessary noise. Moreover, because the frequency

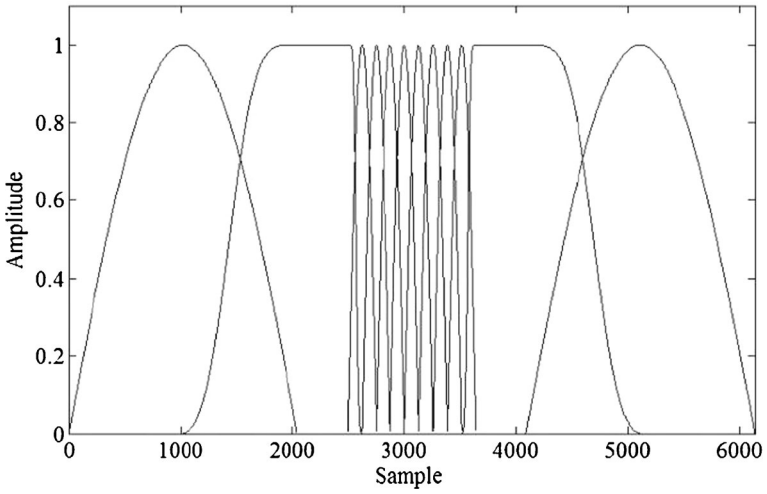

Fig. 2 Dynamic window switching for transient

interval of octaves increases exponentially, researchers must consider frequency resolution when extracting the major frequency from the first few octaves. The details of 12 notes of Western music are shown in Table 2.

Table 3 lists the distinct sampling rates for the AAC codec with window switching and 50 % overlap. The 44.1-kHz sampling rate exhibits the most effective time resolution but shows the poorest frequency resolution. By contrast, the 16-kHz sampling rate exhibits the most effective frequency resolution and sufficient time resolution, and it has to extract less data.

An example of a song clip is shown in Fig. 3, where *Short_ratio* represents the short-window ratio that is defined as (1):

$$Short\_ratio = \frac{number\ of\ short\ window\ frames}{number\ of\ total\ frames} \tag{1}$$

Figure 3 shows that the signal encoded at the 44.1-kHz sampling rate explicitly allocates short windows for the successive transient parts to preserve audio quality. However, for the same signal encoded at a sampling rate of 16 kHz, the AAC codec becomes more sensitive to the transient parts of the signal, which cause the short-window ratio to increase considerably.

2.3 Impact of MDCT and DFT

DFT is the transform used most commonly to transfer discrete signals into the frequency domain. The frequency index of DFT starts at 0 Hz with equal space for latter indices, meaning that the frequency of each index consists of a multiple [7], which is defined as follows:

$$X_{DFT}(k) = \sum\nolimits_{n=0}^{N-1} x(n) \cdot e^{-j\left(\frac{2\pi kn}{N}\right)}, \quad k = 0, 1, \ldots, N-1. \tag{2}$$

where k represents the frequency index, N is the number of sampling points in each frame, and x is the input signal in the time domain. However, unlike with DFT, the frequency of the MDCT coefficient has an $f_s/2N$ shift, and thus the first MDCT coefficient does not start at 0 Hz. With f_s as the sampling frequency, the MDCT is defined as follows:

$$X_{MDCT}(k) = \sum\nolimits_{n=0}^{N-1} x(n)h(n)cos\left(\frac{2\pi}{N}\left(k+\frac{1}{2}\right)\left(n+\frac{1}{2}+\frac{N}{4}\right)\right) \quad k = 0, 1, \ldots, \frac{N}{2}-1. \tag{3}$$

Furthermore,

$$\begin{aligned} X_{MDCT}(\mathrm{k}) &= \mathrm{Re}\left\{\sum_{n=0}^{N-1} x(n)h(n)\cdot e^{-j\left(\frac{2\pi\left(k+\frac{1}{2}\right)\left(n+\frac{1}{2}+\frac{N}{4}\right)}{N}\right)}\right\} \\ &= Re\left\{\sum\nolimits_{n=0}^{N-1} x(n)h(n)\cdot e^{-j\left(\frac{2\pi kn}{N}\right)}\cdot e^{-j\left(\frac{2\pi k}{N}\left(\frac{1}{2}+\frac{N}{4}\right)\right)}\cdot \quad e^{-j\left(\frac{2\pi n}{N}\left(\frac{1}{2}\right)\right)}\cdot e^{-j\left(\frac{2\pi}{N}\left(\frac{1}{4}+\frac{N}{8}\right)\right)}\right\} \end{aligned} \tag{4}$$

where $h(n)$ is a window function that is mentioned in Section 2. The MDCT can be represented as the real part of the Fourier transform with a $1/2+N/4$ temporal delay in the windowing signal, a 1/2 shift in angular frequency, and a phase shift that corresponds to the length of sampling points [6]. The restrictions on the optimal reconstruction of the window function are expressed as follows [15]:

$$\begin{aligned} h\ (n) &= h\ (N-1-k) \\ h^2(n) + h^2\left(n+\frac{N}{2}\right) &= 1 \end{aligned}$$

Table 2 Twelve-tone equal temperament

Note	Octave (Frequency in Hz)										
	0	1	2	3	4	5	6	7	8	9	10
C	16.352	32.703	65.406	130.81	261.63	523.25	1046.5	2093.0	4186.0	8372.0	16744.0
C#	17.324	34.648	69.296	138.59	277.18	554.37	1108.7	2217.5	4434.9	8869.8	17739.7
D	18.354	36.708	73.416	146.83	293.66	587.33	1174.7	2349.3	4698.6	9397.3	18794.5
D#	19.445	38.891	77.782	155.56	311.13	622.25	1244.5	2489.0	4978.0	9956.1	19912.1
E	20.602	41.203	82.407	164.81	329.63	659.26	1318.5	2637.0	5274.0	10548.1	21096.2
F	21.827	43.654	87.307	174.61	349.23	698.46	1396.9	2793.8	5587.7	11175.3	22350.6
F#	23.125	46.249	92.499	185.00	369.99	739.99	1480.0	2960.0	5919.9	11839.8	23679.6
G	24.500	48.999	97.999	196.00	392.00	783.99	1568.0	3136.0	6271.9	12543.9	25087.7
G#	25.957	51.913	103.83	207.65	415.30	830.61	1661.2	3322.4	6644.9	13289.8	26579.5
A	27.500	55.000	110.00	220.00	440.00	880.00	1760.0	3520.0	7040.0	14080.0	28160.0
A#	29.135	58.270	116.54	233.08	466.16	932.33	1864.7	3729.3	7458.6	14917.2	29834.5
B	30.868	61.735	123.47	246.94	493.88	987.77	1975.5	3951.1	7902.1	15804.3	31608.5
Frequency interval	1~1.7	1.9~3.5	3.9~6.9	7.8~13.9	15.6~27.7	31.1~55.4	62.2~110.8	124.5~221.8	248.9~443.5	497.8~887.1	995.7~1774

Table 3 List of frequency resolution and time resolution in different sampling rate

Sampling rate	Long window		Short window	
	Frequency resolution	Time resolution	Frequency resolution	Time resolution
44.1 kHz	21.5 Hz	0.023 s	173.3 Hz	0.006 s
32 kHz	15.6 Hz	0.032 s	125 Hz	0.008 s
16 kHz	7.8 Hz	0.064 s	62.5 Hz	0.016 s

The following example addresses the impact of distinct central frequencies in DFT and MDCT. Two sinusoidal signals of 15.63 and 19.53 Hz are generated at the 16-kHz sampling rate using a Hanning window (2,048 samples). These two signals are the exact central frequencies of the third indices of DFT and MDCT, respectively, and the magnitude responses are shown in Fig. 4. As shown in Fig. 4a, the 15.63-Hz signal reveals a peak in the DFT spectrum, but MDCT distributes the energy to the nearby indices. By contrast, Fig. 4b shows that with the 19.53-Hz signal, a peak appears in the MDCT spectrum, whereas DFT forms a flat energy spectrum between several indices. Accurate frequency mapping is critical for chroma feature extraction.

3 Proposed method

3.1 Feature-extraction algorithm

An overview of the proposed mapping procedure is presented in Fig. 5. Before chroma mapping, MDCT coefficients are extracted from the AAC bitstream and only the magnitude is retained for later use. Long window frames (LWFs) are first mapped to the chroma feature, which is also used to reconstruct part of the chroma feature in short window frames (SWFs). The detailed processing of long and short window frames is described next.

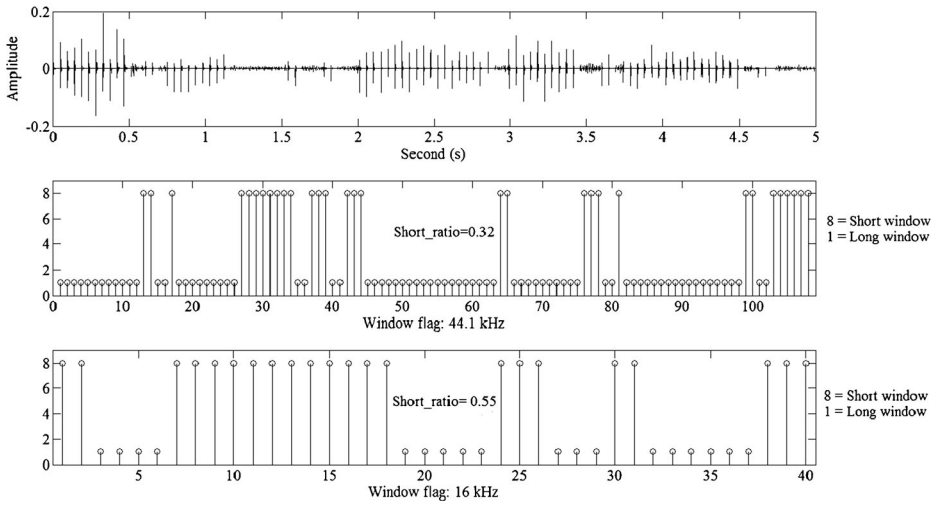

Fig. 3 *Beatles*-Yesterday, *Top*: temporal waveform. *Middle*: window flag at 44.1 kHz sampling rate. *Bottom*: window flag at 16 kHz sampling rate

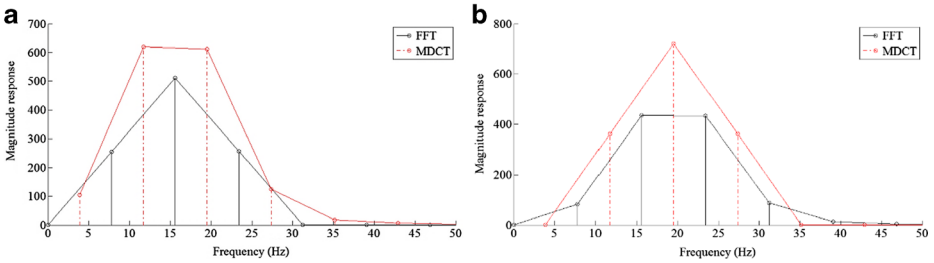

Fig. 4 Magnitude response of single tone **a** 15.63 Hz **b** 19.53 Hz

3.2 Long window frame processing

In the LWF, a magnitude threshold is used to eliminate weak harmonic components and to enhance note representation. The valid MDCT component is defined as follows:

$$X^{'}(k)=\begin{cases} X_{MDCT}(k), & if\ X_{MDCT}(k)>\Omega(X_{MDCT}) \\ 0, & otherwise \end{cases} \tag{5}$$

where

$$\Omega(X_{MDCT})=\frac{\sum_k X_{MDCT}(k)}{U},\quad k=c_1,c_1+1,\ldots,c_2 \quad \text{and}$$

$$c_1=\frac{f_{min}}{resf}+\frac{1}{2},\quad c_2=\frac{f_{max}}{resf}+\frac{1}{2},\quad resf=\frac{f_s}{N}. \tag{6}$$

In (5), k is one of the valid frequency components and U is the number of k; f_{min} and f_{max} are the minimal frequency and maximal frequency that are used to determine the bandwidth; $resf$ is the frequency resolution of each MDCT coefficient; and and are the ceiling and floor mathematical operations, respectively. For the chroma-mapping method, the mapping algorithm proposed by Ravelli was modified to render it suitable for MDCT [21], as shown below:

$$Bin(b)=\mathrm{mod}\left(round\left(12log_2\left(\frac{resf\cdot\left(k+\frac{1}{2}\right)}{f_0}\right)\right),12\right),\quad b=k-c_1\Leftrightarrow b=0,1,\ldots,c_2-c_1. \tag{7}$$

$$Ch(Bin(b),i)=\begin{cases} skip, & if\ X^{'}(c_1+b)=0 \\ Ch(Bin(b),i)+X^{'}(c_1+b), & otherwise \end{cases} \tag{8}$$

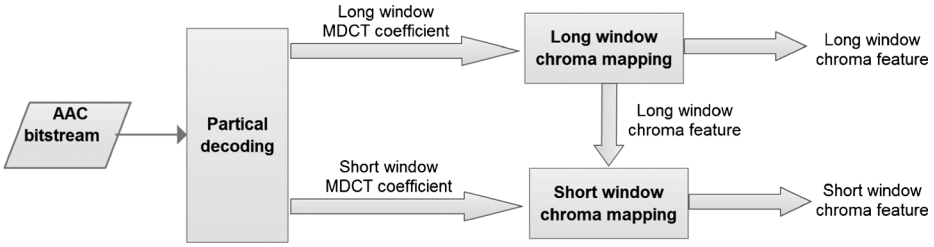

Fig. 5 Chroma feature mapping block diagram

where i is the frame index, f_0 is the lowest note C0 in the 12-tone equal temperament, which is 16.352 Hz, and *Bin* represents the current frequency index corresponding to the semitone. Finally, depending on (7), the energy of the major notes in a frame is added appropriately to 12 semitones in (8).

3.3 Short window frame processing

SWF processing is depicted in Fig. 6. It is observed that that the short-window flag decision occurs not only in the true transient parts of the signal but also from the noise or erroneous judgment. To reduce the impact from incorrect transient, consecutive SWFs that are longer than a threshold are interpolated by neighboring long-window chroma features which have the advantage of clear frequency information. A small threshold does not solve the problem from incorrect transient, while a large threshold might lead to incorrect interpolation. A suitable threshold for the number of consecutive SWFs was obtained by running experiments. The results showed that four was the most suitable number. For other consecutive SWFs that are longer than four frames, SWFs are processed using more sophisticate methods that are described in the next section.

3.3.1 Subframe combination

As mentioned in Section 2, the transient part of the signal is divided into eight subframes by selecting short windows. However, these subframes might be highly correlated in a short window. Here, two methods for combining subframes are suggested, and the details of the combination methods are the following:

Intuitional method: *Direct combination*

An intuitive method for subframe combination that is used conventionally is the summation of all of the subframes in an SWF into one frame. Theoretically, this

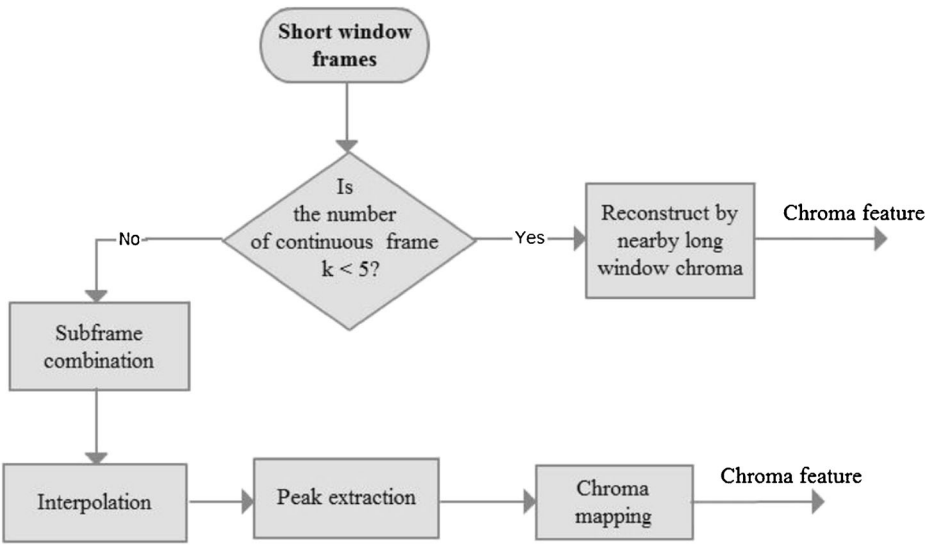

Fig. 6 Short window frame processing flowchart

method can perform the spectral distribution of an SWF because all of the characteristics of the subframes are preserved. Direct combination is defined as (9).

$$X_{comb}(k) = \sum_{r=0}^{7} X_s^{'}(r,k) \tag{9}$$

Where r is the index of a subframe in an SWF $X_s^{'}$ and k represents the MDCT coefficient index from c_1 to c_2.

Proposed Method: *Peak competition*

The proposed method was developed based on assuming that the audio files uploaded on the internet may suffer lossy compression when received by file users or managers. Although the human ear might not detect any deterioration in audio quality, the frequency of the signal might be shifted substantially because of having passed through many filters or analysis/synthesis processing steps. Consequently, a frequency component that is critical for feature extraction might not be identified in a noisy spectrum. Hence, we propose a peak-competition method to improve the accuracy of subframe combination. An example of a frequency shift of an SWF is shown in Fig. 7. First, we define an "abrupt peak" (Fig. 7, red circle) and a "subduction zone" (red arrow). An "abrupt peak" is a peak in which least one side falls to zero, and which has a frequency magnitude higher than a specified threshold value. The "subduction zone" refers to a peak that is hidden in a flat area. Figure 7 shows five candidate abrupt peaks that present notes with index values of 3, 4, 5, 6, and 9; the figure also shows subduction zones in subframes 3, 4, 6, and 8. Our aim is to not only identify the most representative peak from adjacent abrupt peaks, but to also reveal peaks hiding in subduction zones.

The next step is to combine valid MDCT coefficients in the subframes of an SWF. The method proposed is to assign a set of weights to each MDCT coefficient to rapidly attenuate competition failure components. The variation of the weight vector W depends on the frequency magnitude of the subframes and a threshold value T_s, which is defined as follows:

$$T_s(r) = G_t \cdot \Omega\left(X_s^{'}(r,k)\right) \tag{10}$$

where G_t is a threshold gain; the weight W is calculated using (11).

$$W(k) = \prod_{r=0}^{7} \alpha(r,k) \tag{11}$$

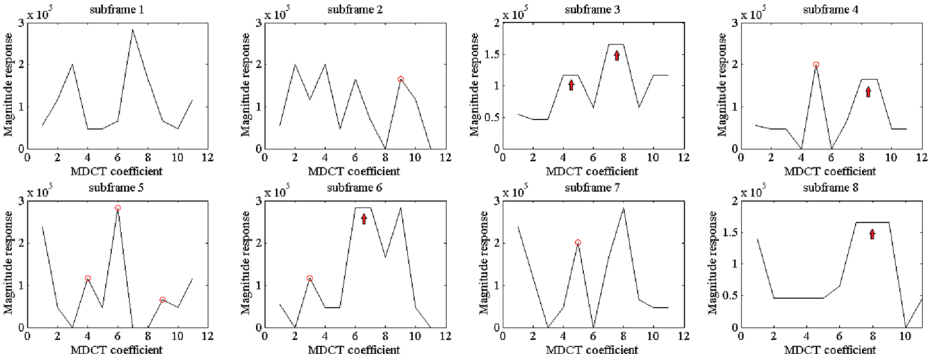

Fig. 7 Frequency energy distribution of a short window frame

where

$$\alpha(r,k) = \begin{cases} G_d & if\ X_s'(r,k) \le T_s(r) \\ 1 & otherwise \end{cases} \tag{12}$$

The attenuation factor G_d is similar to an exponent decay, which controls the slope of a curve. As the curves with distinct attenuation factors in Fig. 8 show, when the value of G_d is small, the curve is steep, whereas when the value of G_d high, the curve falls more gently. Finally, the combined frame X_{comb} is given as

$$X_{comb}(k) = W(k)\sum_{r=0}^{7} X_s'(r,k) \tag{13}$$

Based on experiments, G_t and G_d are set to 0.1 and 0.645, values with which optimal performance can be obtained with the proposed method; therefore, these setting of G_t and G_d were used in all of the experiments described in this article. Figure 9 shows the combination results of Fig. 7 obtained using direct combination and the peak-competition method. Direct combination exhibits highly ambiguous pitches and therefore loses the prime information of an SWF. By contrast, the peak-competition method not only maintains the main pitches but also reduces the magnitude of ambiguous peaks substantially. The experimental results show that the proposed method can retain pitch information effectively when combining subframes in an SWF.

3.3.2 Interpolation

Because of the low frequency resolution in SWFs, pitch information may be lost due to the incorrect assignment of the frequency magnitude to neighboring points. Here, an interpolation is used to reconstruct the peaks that occur between frequency intervals. A set of MDCT coefficients of a subframe is used as an example to analyze the effect of the interpolators. At low frequency resolution, many pitches or harmonics are distributed to nearby indices, which are marked by arrows in Fig. 10a. For these divergent

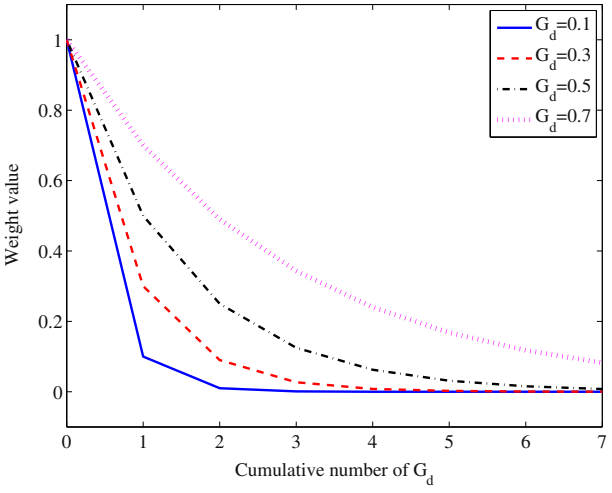

Fig. 8 Decay curves of different attenuation factor G_d

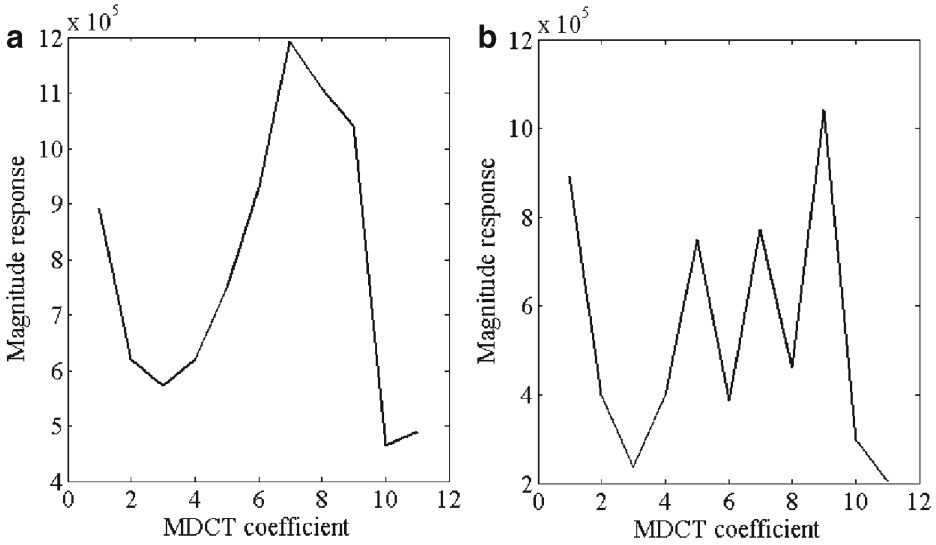

Fig. 9 Results of subframe combination **a** Direct combination **b** Peak competition

areas, a high order interpolator (such as a cubic spline or a Lagrange polynomial) [9, 11] is required to smooth the curve. Based on our experiments, Oetken et al. proposed an interpolator method [18, 20] that performed optimally in interpolating two samples to smooth the coefficients and reveal latent peaks. In this method, an optimal low-pass filter is designed that allows the original signal to pass through unchanged and interpolates signals in between by minimizing the mean-square error.

At the sampling rate at 16 kHz, the frequency resolution of each MDCT coefficient of the SWF is improved from 62.5 to 20.83 Hz when interpolating two samples.

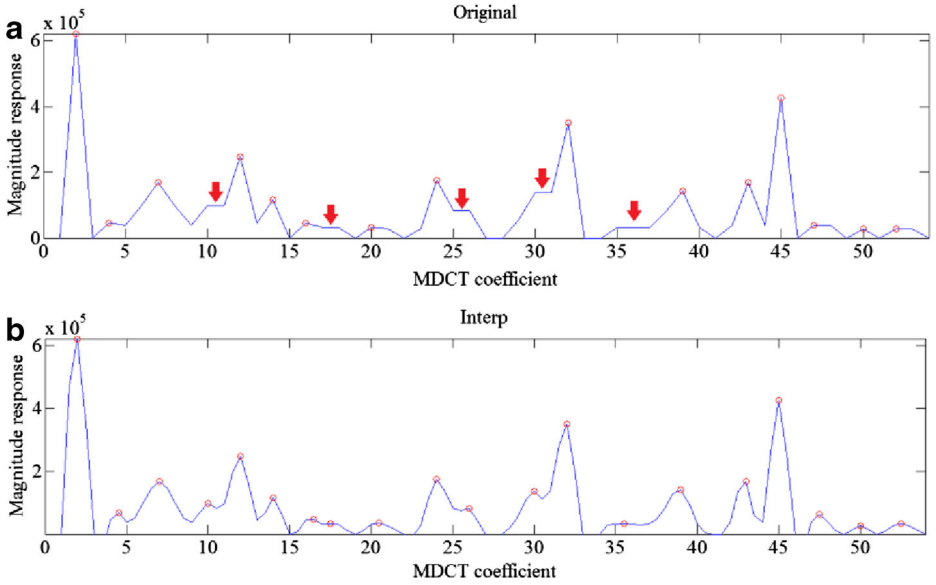

Fig. 10 Peaks of MDCT coefficient. **a** Original magnitude **b** After interpolation

Figure 10b presents the interpolation signal obtained using Oetken's method, with the peaks marked by red circles; the figure shows that several areas with flat energy are reshaped as sine trends and form peaks. These peaks are extracted using (7) to map into chroma bins and are processed using normalization.

3.4 Frequency of notes: analysis and selection

Selecting a frequency range is critical for chroma feature mapping. In "pop" music, the percussion often maintains a song's beat and its frequency is below 100 Hz; by contrast, most other instruments are played at frequencies from 130 Hz to 1 kHz, spanning three octaves. Moreover, most musical components higher than 1 kHz are harmonics that may influence the chroma mapping because of the multiples of pitch components. For example, the third and fifth harmonics of C3 may map to G4 and E5, respectively, which are in incorrect bins. Because frequency selection considers the characteristics of musical instruments, it provides a useful method to enhance chroma in our experiment.

Table 3 shows that the frequency resolution of the LWF at a sampling rate of 16 kHz is 7.8 Hz, which matches exactly the frequency interval at the third octave. Thus, for the LWF, the frequency range from 124 Hz to 1 kHz was selected. However, for the SWF, pitch information cannot be extracted readily because of low frequency resolution. Therefore, we focus on the multiples of pitch components and interpolate two samples to obtain 20.83-Hz frequency resolutions. Finally, the frequency selection used in the SWF ranged from 468 Hz to 2 kHz.

3.5 Complexity analysis

Comparing to the decode-then-extract approach that needs full decoding including inverse MDCT, this compression domain approach only needs partial decoding that consumes less than one quarter of the decoding computational complexity, as shown in Fig. 1. Our proposed method mainly aims at improving the SWF performance with minor computational overhead. The cover80 dataset was used to evaluate the extra processing time in addition to the partial decoding. Experiments showed that Ravelli's method in which feature extraction were obtained based on all long-windows increased 0.1 % computation complexity, while our proposed method yielded 1.4 % computation complexity overhead. Considering the time saving from skipping the inverse MDCT, the overheads from either Ravelli's method or our proposed method were insignificant.

4 Evaluation functions

The preceding sections describe how to process MDCT coefficients effectively for long windows and short windows. However, the extent to which performance is improved in a chromagram cannot be verified. Thus, a system for identifying "cover" songs was used to evaluate chromagram representation. Compared with the original song, the cover version may vary considerably in tempo, structure, timbre, key, or language of the vocals [23]. When an ineffective chroma-representation method is used, the accuracy of finding the original or the cover song is low. In the matching procedure used in this study, we adopted the algorithm proposed by Serra et al. which contains mainly an optimal transposition index (OTI), a binary similarity matrix (BSM), and dynamic programming local alignment (DPLA) [22, 24]. An

additional modification, a dynamic segmentation, was used to substitute for the complex beat-tracking operation; the proposed segmentation strategy was incorporated in this work to improve performance.

4.1 Assessment methodology

A large dataset is appropriate for comparing music-retrieval methods objectively. SecondHandSongs (SHS) is currently the largest dataset of cover song tasks [2], containing 12,960 training sets and 5,236 test sets. All of the songs of SHS are part of the Million Song Dataset (MSD) [3]. However, SHS does not provide audio files that enable custom feature extraction. In addition, most of cover songs retrieval tasks working on files at 16 kHz sampling rate. Therefore, to compare waveform-domain retrieval methods objectively, in our experiments, Cover80 and EA50 were used as evaluation datasets, in which all audio files are converted to the AAC format at a sampling rate of 16 kHz. The Cover80 dataset has 80 song sets of Western pop music stored in the MP3 format [8]. The EA50 dataset was obtained from a personal collection with 50 song sets of East Asian pop music. Each song set represents an original song and a cover song. Another dataset, DB130, combines Cover80 and EA50 to complicate the retrieval task.

To evaluate music retrieval across experiments, a mean reciprocal rank (MRR) was used, which is defined as

$$MRR = \frac{1}{Q}\sum_{q=1}^{Q}\begin{cases}\frac{1}{rank_z}, & rank_z \le p\\ 0, & rank_z > p\end{cases} \tag{14}$$

where Q is the number of queries and $rank_z$ is the rank of the first correct answer in the list of answer candidates z. In MRR, the reciprocal ranks of top-p are counted and averaged; in our experiments, p was fixed as 10 [4, 14].

4.2 A simple dynamic-segmentation method

In the MPEG-2 AAC standard, the input signal is encoded frame-by-frame [13]. An average 3.5-min song has approximately 3,000 frames at a sampling rate of 16 kHz. Audio is a continuous signal in which a preceding frame and a following frame are highly correlated. Serra [24] reported that frame combination performed better than beat tracking when dynamic time warping (DTW) was used for alignment. Serra's matching method allowed deviations of the double or half tempo, and the best performance was obtained with segment size between 0.7 and 1.16 s; however, for chroma transformation, the 36-bin harmonic pitch-class profile (HPCP) method was used. Under distinct conditions of chroma number and chroma transformation, we propose, in addition to a fixed segment size, a dynamic-segmentation method, with which the segment size is calculated based on the length of a song, as defined in (15).

$$Seg_{size} = round\left(\frac{8\cdot M}{2048}\right) \tag{15}$$

Seg_{size} is limited as follows:

$$Seg'_{size} = \max(\min(Seg_{size}, Seg_{max}), Seg_{min})$$

where M is the number of frames of a song. The maximal segment size Seg_{max} and minimal segment size Seg_{min} were set to be 19 and 9 frames, respectively. A song with dynamic segmentation for a fragment was limited between 0.63 and 1.28 s. The reason for using (15) is

to allow a long adagio song and a short light song, which are common in pop music, to be segmented with appropriate adjustments. Table 4 lists the average number of features of dynamic segmentation and the beat tracking used by Ellis for Cover80 [8]. The statistical data show that with dynamic segmentation, feature data were nearly 3.7-times less than the data with beat tracking.

4.3 Optimal transposition index and binary similarity matrix

Because cover songs may be performed in a key distinct from that of the original song, transposing the key of a chroma feature of one song to that of the other would help similarity measurements considerably. For transposing, OTI is used to calculate the global key difference between two songs; OTI is defined as

$$T_r = \text{argmax}_{id=0,1,\ldots,11}\{mean(Ch_A)\cdot mean(circshift(Ch_B, id))\} \tag{16}$$

where " " indicates a dot product, and *circshift*() is a function that rotates the vector Ch_B with *id* positions. According to the key difference T_r, a BSM is generated:

$$BSM(\rho,\sigma) = \begin{cases} 1, & if\ OTI_s(Ch_{A,\rho}, Ch_{B,\sigma}) = T_r \\ 0, & otherwise \end{cases} \tag{17}$$

where ρ and σ represent the segment index of Chroma A and Chroma B, respectively. OTI_s, which calculates the note change for all of the segments between the two songs, is defined as follows:

$$OTI_s(\rho,\sigma) = \text{argmax}_{id=0,1,\ldots,11}\left\{Ch_{A,\rho}\cdot circshift\left(Ch_{B,\sigma}, id\right)\right\} \tag{18}$$

Figure 11 presents examples of BSMs of reference/cover and reference/non-cover songs, which show that similar songs have clear diagonal white lines in the BSM; this prominent characteristic of the BSM can be used by the DPLA algorithm to evaluate the similarity score.

4.4 Dynamic programming local alignment

The Smith-Waterman (*SW*) algorithm is used for identifying local matches in genetics [26]. In the matching step of the proposed method, this algorithm is used to grade the similarity between the original and cover songs [24]. First, we assume that the original song contains L_1 segments and the cover song contains L_2 segments, and then the $(L_1+1)\times(L_2+1)$ matrix SW is created using the following recursive formula:

$$SW(\rho,\sigma) = \max\begin{cases} SW(\rho-1,\sigma-1) + BSM(\rho-1,\sigma-1) \\ SW(\rho-2,\sigma-1) + BSM(\rho-1,\sigma-1) - \alpha \\ SW(\rho-1,\sigma-2) + BSM(\rho-1,\sigma-1) - \beta \\ 0 \end{cases} \tag{19}$$

for $\rho = 3,4,\ldots,L_1+1, \quad \sigma = 3,4,\ldots,L_2+1$

Table 4 The average number of features for Cover80: Ellis' beat tracking method and dynamic segmentation

	Beat tracking	Dynamic segmentation
Number of features	953.7	260.3

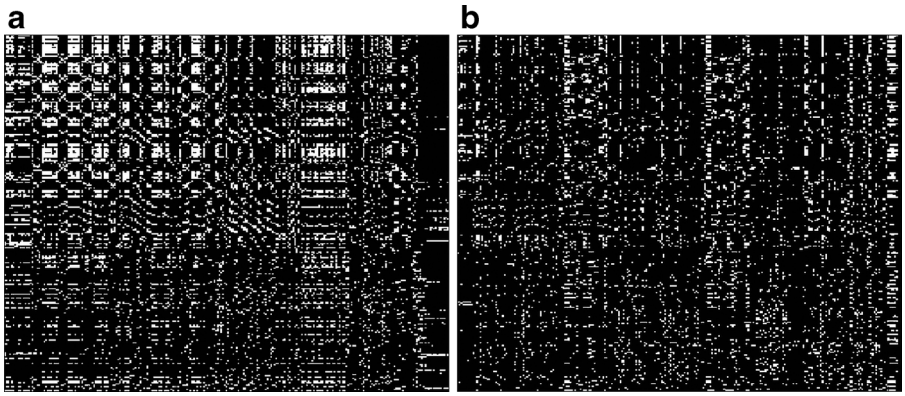

Fig. 11 BSM matrix: **a** Similarly songs **b** Non-similarly songs

In this formula, the BSM is used as the input to update the *SW* matrix recursively. The constants α and β are used as follows:

$$\begin{cases} \alpha, \beta = 0, & if\ BSM(\rho{-}1, \sigma{-}1) = 1 \\ \alpha = 0.5, \quad \beta = 0.6, & otherwise \end{cases} \tag{20}$$

Every input query creates an *SW* matrix with each original song in the database. Figure 12 shows examples of matching trends of similar and non-similar songs in the SW matrix. The maximal value in the *SW* matrix is used as the similarity score. Finally, the retrieval result is returned based on the ranking of the similarity scores of all songs in the database.

5 Experimental analysis

5.1 Frequency range and segmentation

To assess the impact of frequency selection and fragment size, four frequency modules with six distinct segment sizes (Tables 5 and 6) were used for cross-experimental analyses. The frequency ranges of Modes 1 to 3 were those used in notable previous studies, and Mode 4 used for Ravelli's method was only processed using the long window. For preliminary

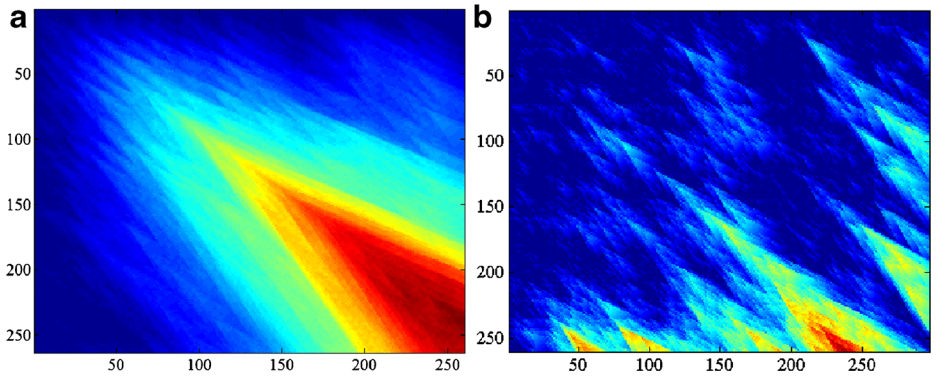

Fig. 12 SW matrix: **a** Similarly songs **b** Non-similarly songs

Table 5 Different frequency range modules

	Long window	Short window
Mode 1 [8]	100~1 k Hz	
Mode 2	100~2 k Hz	
Mode 3 [24]	40~5 k Hz	
Mode 4	124~1 k Hz	468~2 k Hz

experiments, the Cover80 dataset was employed, and the MRRs calculated for various combinations are shown in Table 7.

The results in Table 7 show that Ravelli's method and the intuitional method performed optimally in Mode 4. Except for dynamic segmentation, most of the optimal results of the combination factor are between 7 and 15, which is consistent with previous work [24]. In our experiments, we noted that harmonics at high frequencies affected the chroma feature markedly and led to poor matching results, which may be because of the mapping errors that occur at high frequency harmonics (explained in Section 3.4). Furthermore, Ravelli's method performed effectively because the SWF corresponding to the LWF has a relationship that equals 8-times the frequency, which is merely the harmonic component of the pitch in the first subframe. Hence, Ravelli's method performed effectively when operating in Modes 1 and 4.

5.2 Chroma transformation evaluation for SWF

In this study, we evaluated the accuracy of chroma transformation in the SWF by assigning distinct weights to SWFs. Because a song contains many SWFs, when most of the chroma in SWFs are accurate, performance improves steadily when the weighting value W_s increases. Conversely, performance deteriorates when most of the chroma are inaccurate. The experimental setting is described in Table 8 and the results of varying the weighting are shown in Fig. 13.

Table 6 Different combinations in time

Combination factor (Frame count)	3	7	11	15	19	Dynamic
Segment length (second)	0.256	0.512	0.768	1.024	1.280	Variable

Table 7 MRR within Top-10 retrieved songs in Cover80 dataset

	Combination factor					
	3	7	11	15	19	Dynamic
Ravelli mode 1	0.4474	0.4333	0.4808	**0.5176**	0.4838	0.5031
Ravelli mode 2	0.2458	0.3386	**0.3628**	0.3592	0.3409	0.3433
Ravelli mode 3	0.3112	**0.3974**	0.3423	0.3320	0.3058	0.3441
Ravelli mode 4	0.4319	**0.5530**	0.5252	0.5252	0.4732	0.5040
Intuitional mode 1	0.4061	0.4739	0.4659	**0.5101**	0.5087	**0.5264**
Intuitional mode 2	0.4875	0.5247	0.5476	0.5362	**0.5508**	0.5424
Intuitional mode 3	0.3860	0.4471	0.4374	**0.4727**	0.4377	0.4466
Intuitional mode 4	0.4548	0.5938	**0.5983**	0.5702	0.5603	**0.6158**

Numbers in boldface indicate the highest MRR in a row

Table 8 Description of experiment setting

Dataset	Segmentation	Frequency selection	Weighting W_s
Cover80	Dynamic	Mode 4	0.1~1

Figure 13 shows that Ravelli's method exhibits the most marked decreasing trend because SWF processing is not considered in the method. Thus, with Ravelli's method, an increase in weighting leads to an increase in the impact of incorrect chroma information. In the intuitional method with interpolation, ambiguous frequency components are deemphasized (Section 3.3.1) and low weighting can suppresses the influence of ambiguous frequency components; therefore, this method yields better results than the proposed method with interpolation when W_s is <0.6. However, the performance curve of the proposed method with interpolation shows an upswing when weighting is increased, which indicates that using interpolation contributes substantially to enhancing the chroma feature in SWFs.

5.3 Comparison of waveform-domain music-retrieval methods

We compared our method with state-of-the-art waveform-domain methods for retrieving music as shown in Tables 9. In the reference method [22], the method names were represented as Feat *u v* bpm, where *u* was one of the three sets of feature extraction and similarity matching (1: chroma + cross-correlation, 2: chroma with high pass filter + corss-correlation, 3: harmonic pitch class profile + DPLA), and *v* represented the beats that were tracked from three distinct tempos. For "Top-1" experiments, the proposed method was as accurate as methods presented in waveform-domain studies. However, in our method, the decode-then-encode procedure is not required and the complex computation of beat tracking is replaced by simple dynamic segmentation.

Figure 14 shows the accuracy rates in "Top-10" experiments for Ravelli's method, the intuitional method, and the proposed method, all used with the optimal parameter setting for

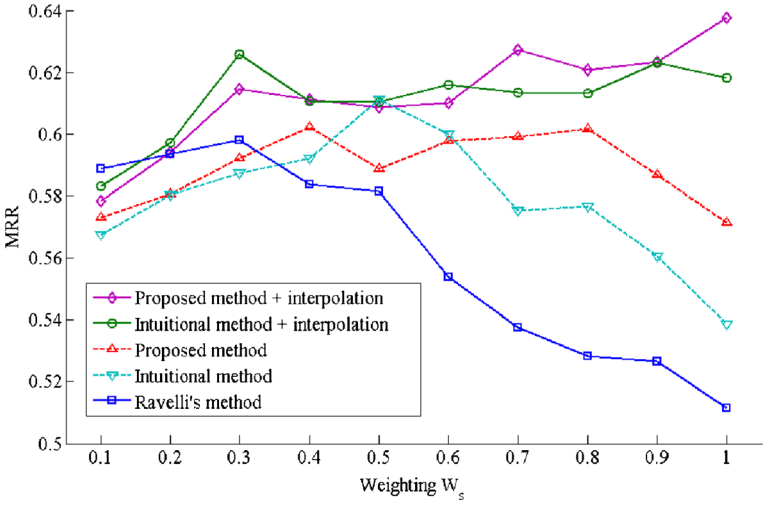

Fig. 13 Results of different weights to SWFs

Table 9 Modern technique for Single tempo chroma feature comparison in Cover80 [22]

Method name	Segmentation	Similarity computation	Correct (Top-1)
Feat 1 240 bpm	240 bpm	Cross-correlation	46/80=57.50 %
Feat 1 120 bpm	120 bpm	Cross-correlation	49/80=61.25 %
Feat 1 60 bpm	60 bpm	Cross-correlation	45/80=56.25 %
Feat 2 240 bpm	240 bpm	Cross-correlation	50/80=62.50 %
Feat 2 120 bpm	120 bpm	Cross-correlation	50/80=62.50 %
Feat 2 60 bpm	60 bpm	Cross-correlation	54/80=67.50 %
Feat 3 240 bpm	240 bpm	DPLA	48/80=60.00 %
Feat 3 120 bpm	120 bpm	DPLA	49/80=61.25 %
Feat 3 60 bpm	60 bpm	DPLA	51/80=63.75 %
Ravelli's method	Frame count=7	DPLA	42/80=52.50 %
Intuitional method	Dynamic	DPLA	47/80=58.75 %
Proposed method	Dynamic	DPLA	50/80=62.50 %

evaluation with the DB130 dataset. The proposed method exhibits greater than 70 % accuracy in Top-1 music-retrieval performance and is approximately 7 % more accurate than Ravelli's method.

6 Conclusion

This paper proposes a chroma-transformation method based directly on the AAC transform domain. Unlike previous studies, we considered the impact of sampling rate, frequency resolution, frequency range selection, and window switching to propose a chroma-enhancement method that moderately processes the problem of frequency mapping in window switching. Specifically, for the short window frame, using the proposed method reduces the

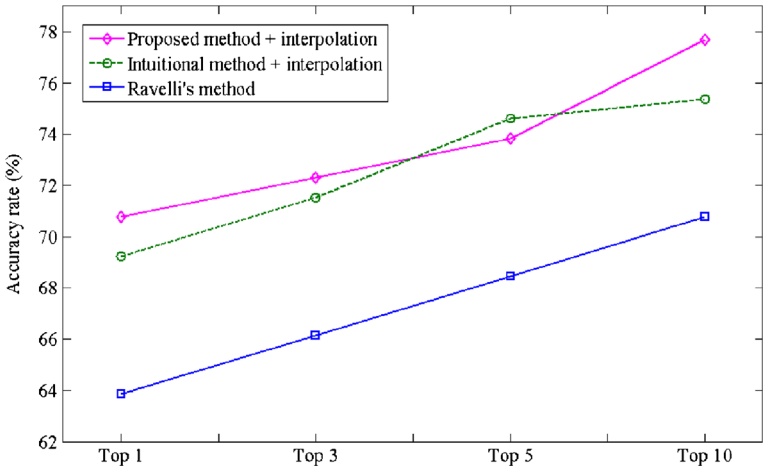

Fig. 14 Top-10 retrieval accuracy rate in DB130

distortion of frequency of MDCT caused by multiple encoding, and using an interpolation with the proposed method increases frequency resolution and reveals latent peaks. This frequency-enhancement procedure helps to raise the pitch exactitude when frequency is ambiguous. We also propose a simple dynamic-segmentation method to adjust segment size, which increases the accuracy rate slightly more than using fixed fragment size. In conclusion, our method provides a simpler chroma-transformation operation but retrieves music nearly as accurately as waveform-domain methods.

References

1. Bello JP, Pickens J (2005) A robust mid-level representation for harmonic content in music signals. In Proc. Int. Conf. Music Inf. Retrieval, pp 304–311
2. Bertin-Mahieux T, Ellis DPW (2011) Large-scale cover song recognition using hashed chroma landmarks. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp 117–120, 2011
3. Bertin-Mahieux T, Ellis DPW, Whitman B, Lamere P (2011) The million song dataset. In Proceedings of the 12th International Society for Music Information Retrieval Conference
4. Chakrabarti S, Khanna R, Sawant U, Bhattacharyya C (2008) Structured learning for non-smooth ranking losses. Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 88–96
5. Chang TM, Chen ET, Hsieh CB, Chang PC (2013) Cover song identification with direct chroma feature extraction from AAC files. IEEE 2nd Global Conference on Consumer Electronics, in press
6. Chen S, Xiong N, Park J, Chen M, Hu R (2010) Spatial parameters for audio coding: MDCT domain analysis and synthesis. Multimed Tools Appl 48(2):225–246
7. Cooley JW, Tukey JW (1965) An algorithm for the machine calculation of complex Fourier series. Math Comput 19:297–301
8. Ellis DPW, Poliner GE (2007) Identifying cover songs with chroma features and dynamic programming beat tracking. In Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Honolulu, Hawaii, U. S. A, pp 1429–1432
9. Fan J, Yao Q (2005) Nonlinear time series: nonparametric and parametric methods. Springer, New York
10. Fujishima T (1999) Realtime chord recognition of musical sound: a system using common lisp music. In Proc. Int. Comput. Music Conf., pp 464–467
11. Hinsen G, Klösters D (1993) The sampling series as a limiting case of Lagrange interpolation. Appl Anal 49(1–2):49–60
12. ISO/IEC 11172–3 (F) (1999) Information technology—coding of moving picture and associated audio for digital storage media at up to about 1.5Mbits/s Part3: Audio
13. ISO/IEC 13818–7 (1997) Information technology—generic coding of moving pictures and associated audio, Part7: Advance Audio Coding
14. Lee MH, Rho S, Choi EI (2013) Ontology based user query interpretation for semantic multimedia contents retrieval. Multimed Tools Appl. doi:10.1007/s11042-013-1383-2
15. Malvar H (1992) Signal processing with lapped transforms. Artech House, Inc.
16. Müller M, Ewert S (2010) Towards timbre-invariant audio features for harmony-based music. IEEE Trans Audio Speech Signal Proc 18:649–662
17. Nakajima Y, Lu Y, Sugano M, Yoneyama A, Yamagihara H, Kurematsu A (1999) A fast audio classification from MPEG coded data. Proc IEEE Int Conf Acoust, Speech Signal Process 6:3005–3008
18. Oetken G, Parks TW, Schussler HW (1975) New results in the design of digital interpolators. IEEE Trans Acoust Speech, Signal Process 23:301–309
19. Patel N, Sethi I (1996) Audio characterization for video indexing. In Proc. SPIE, pp 373–384
20. Programs for digital signal processing (1979) IEEE Press
21. Ravelli E, Richard G, Daudet L (2010) Audio signal representations for indexing in the transform domain. IEEE Trans Audio, Speech, Lang Process 18(3):434–446
22. Ravuri S, Ellis DPW (2009) The hydra system of unstructured cover song detection. Ext. Abstract for the MIREX Audio Cover Song Identification task submission, Kobe, Japan

23. Serra J, Emilia G, Perfecto H (2010) Advances in music information retrieval. Springer, Berlin
24. Serra J, Gomez E, Herrera P, Serra X (2008) Chroma binary similarity and local alignment applied to cover song identification. IEEE Trans Audio, Speech, Lang Process 16(6):1138–1151
25. Shao X, Xu C, Wang Y, Kankanhalli M (2004) Automatic music summarization in compressed domain. Proc IEEE Int Conf Acoust, Speech Signal Process 4:261–264
26. Smith TF, Waterman MS (1981) Identification of common molecular subsequences. J Mol Biol 147:195–197
27. Tsai TH, Liu C (2007) A configurable common filterbank processor for multi-standard audio decoder. IEICE Trans Fundam Electron Commun Comput Sci 90(9):1913–1923
28. Yu CH, You SD (2002) On the possibility of only using long windows in MPEG-2 AAC coding. IEEE Pacific Rim Conference on Multimedia, pp 663–670

Tai Ming Chang received the B.S. degree in electrical engineering and the M.S. degree in computer and communication from the Southern Taiwan University and Shu Te University, Taiwan, in 2005 and 2007, respectively. He is currently pursuing the Ph. D degree at the Video-Audio Processing Laboratory in communication engineering, National Central University, Taiwan. His research interests include speech/audio processing, audio compression, multi-channel audio signal processing, and music information retrieval.

Chia-Bin Hsieh received the B.S. degree in communication engineering and the M.S. degree in communication engineering from the Yuan Ze University and National Central University, Taiwan, in 2011 and 2012, respectively. His research interests include audio coding, music information retrieval, and so on.

Pao-Chi Chang received the B.S. and M.S. degrees from National Chiao Tung University, Taiwan, in 1977 and 1979, respectively, and the Ph. D. degree from Stanford University, California, 1986, all in electrical engineering. From 1986 to 1993, he was a research staff member of the department of communications at IBM T. J. Watson Research Center, Hawthorne, New York. At Watson, his work centered on high speed switching systems, efficient network design algorithms, and multimedia conferencing. In 1993, he joined the faculty of National Central University (NCU), Taiwan, where he is presently a Professor in the Department of Communication Engineering and the Department of Electrical Engineering, as well as the Advisor of the Video-Audio Processing Laboratory (VAPLab). He was a visiting professor at Stanford University in 2000 and 2004, respectively. His main research interests include speech/audio coding, video/image compression, scalable coding, error resilient coding, digital watermarking and data hiding, and multimedia delivery over packet and wireless networks.