Automatic Structure Detection for Popular Music

Namunu C. Maddage
Institute for Infocomm Research

Our proposed approach detects music structures by looking at beat-space segmentation, chords, singing-voice boundaries, and melody- and content-based similarity regions. Experiments illustrate that the proposed approach is capable of extracting useful information for music applications.

Music structure information is important for music semantic understanding. It consists of time information (beats, meter), the melody/harmony line (chords), music regions (instrumental, vocal), song structure, and music similarities. The components of song structure—such as the introduction (intro), verse, chorus, bridge, instrumental, and ending (outro)—can be identified by determining the melody- and content-based similarity regions in a song. (For a detailed discussion of some of the basics of music and how it pertains to this article, see the sidebar "Music Knowledge" on the next page) We define melody-based similarity regions as the regions that have similar pitch contours constructed from the chord patterns and content-based similarity regions as the regions which have both similar vocal content and similar melody. For example, the verse sections in a song are melody-based similarity regions while chorus sections are content-based similarity regions.

This article presents information based on earlier work with more explanations.[1] Our proposed framework for music structure detection combines both high-level music structure knowledge and low-level audio signal processing techniques. The content-based similarity regions in the music are important for many applications, such as music summarization, music transcription, automatic lyrics recognition, music information retrieval, and music streaming. We describe our proposed approach for music structure detection step by step.

1. Our system first analyzes' the music's rhythm and structure by detecting note onsets and the beats. The music is segmented into frames with the size proportional to the interbeat time interval of the song. We refer to this segmentation method as beat space segmentation.

2. A statistical learning method then identifies the melody transition via detection of chord patterns in the music and of singing voice boundaries.

3. With the help of repeated chord pattern analysis and vocal content analysis, the system detects the song structure.

4. The information (timing, melody/harmony, vocal instrumental regions, music similarities) extracted in our system, including song structure, describes the music structure.

Of course, other research exists on music structure analysis, which we list in the "Related Work" sidebar (see p. 68). The limitation of other methods is that most of the methods have not exploited music knowledge and have not addressed the following issues of the music structure analysis:

- The estimation of the boundaries of repeating sections is difficult if the time information (time signature and meter), and melody of the song are unknown. Note that the time signature (TS) is the number of beats per bar; a TS of 4/4 implies there are four crotchet beats in the bar. If the TS is 4/4 (the most common TS in popular music) then the tempo indicates how many crotchet beats there are per minute. The key is the set of chords by which the piece is built.

- In some song structures, the chorus and verses either have the same melody (pitch contour) or a tone/semitone-shifted melody (different music scale). In such cases, we can't guarantee that we can correctly identify the verse and chorus without analyzing the music's vocal content.

Rhythm extraction and BSS

As we explain in the "Music Knowledge" sidebar, the melody transition, music phases, seman-

1070-986X/06/$20.00 © 2006 IEEE

Music Knowledge

Popular song structure

From a music composition point of view, all the measures of music event changes are based on the discrete step size of music notes. In the following sections, we introduce time alignments between music notes and phrases. This information is directly embedded with music segmentation. Music chords, keys, and scales reveal how such information can be used to correctly measure melody fluctuations in a song. We use general composition knowledge for song writing and incorporate it for high-level music structure formulation.

Music notes

For readers who may not have a background in music, we provide a brief overview. A music note's duration is characterized by a note's onset and offset times. Figure A shows the correlation of the music notes' length, symbols, identities, and their relationships with the silences (rests). The song's duration is measured as a number of bars.[1] While listening to music, the steady throb to which a person could clap is called the pulse or beat and the accents are the beats that are stronger than the others. The numbers of beats from one accent to adjacent accents are equal and it divides the music into equal measures. Thus, the equal measure of the number of beats from one accent to another is called the bar.

In a song, the words or syllables in the sentence fall on beats to construct a music phrase. Figure B illustrates how the words "Baa, baa, black sheep, have you any wool?" form themselves into a rhythm and its music notation. The first and second bars are formed with two quarter notes each. Four eighth notes and a half note are placed in the third and fourth bars, respectively, to represent the words rhythmically.

A music phrase is commonly two or four bars in length. The incomplete bars are filled with notes, rests, or humming (the duration of humming is equal to the length of a music note).

Music scale, chords, and key of a piece

The eight basic notes (C, D, E, F, G, A, B, C), which are the white notes on a piano keyboard, can be arranged in an alphabetical succession of sounds ascending or descending from the starting note. This note arrangement is known as a music scale. Figure C1 shows the note progression in the G scale. In a music scale, the pitch progression of one note to the other is either a half step (a semitone S) or whole step (a tone T). Thus, it expands the eight basic notes into 12 pitch classes. The first note in the scale is known as the tonic and it is the key note (tone note) from which the scale takes the name. Depending on the pitch progression pattern, a music scale is divided into

Note	Shape	Rest	Value in Terms of a Semibreve	Corresponding Names Commonly Used in the US and Canada
Semibreve	𝅝	𝄻	1	Whole note
Minim	𝅗𝅥	𝄼	1/2	Half note
Crotchet	♩	𝄽 or 𝄾	1/4	Quarter note
Quaver	♪	𝄾	1/8	Eighth note
Semiquaver	𝅘𝅥𝅯	𝄿	1/16	Sixteenth note
Demisemiquaver	𝅘𝅥𝅰	𝅀	1/32	Thirty-second note

Figure A. Correlation between different music notes and their time alignment.

tic events (verse, chorus, and so on) occur in interbeat time proportional intervals. Here we narrow down our scope to English-language songs with a 4/4 time signature, which is the commonly used TS. In music composition, smaller notes such as eighth, sixteenth, or thirty-second notes are played along with music phrases to align instrumental melody with vocal pitch contours. Therefore, in our proposed music segmentation approach, we segment the music into the smallest note length frames. This is called beat space segmentation (BSS).

To calculate the duration of the smallest note, we first detect the note onsets and beats of the song according to the steps described in Figure 1. Because the harmonics structure of music signals are in octaves, we decompose the music signal into eight subbands whose frequency ranges we show in Figure 2 (on p. 69).

The subband signals are segmented into 60-ms frames with 50 percent overlap. Both the frequency and energy transients are analyzed using a method similar to Duxburg et al.'s.[2] The fundamentals and harmonics of the music notes in popular music are strong in subbands 01 to 04. Thus, we measure the frequency transients in terms of progressive distances between the spectrums in these subbands. To reduce the effect of strong frequencies generated from percussion instruments and bass-clef music notes (usually generated by bass guitar and piano), the spectrums computed from subbands 03 and 04 are one major scale and three minor scales (natural, harmonic, and melodic). The major and natural minor scales follow the patterns of "T-T-S-T-T-T-S" and "T-S-T-T-S-T-T," respectively. Figure C2 lists the notes that are present in major and minor scales for the C pitch class.

Music chords are constructed by selecting notes from the corresponding scales. Types of chords are major, minor, diminished, and augmented. The first note of the chord is the key note in the scale.

The set of notes on which the piece is built is known as the key. A major key (the chords that can be derived from the major scale) and minor key (the chords that can be derived from three minor scales) are two possible kinds of keys in a scale.

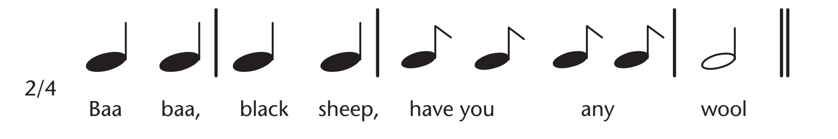
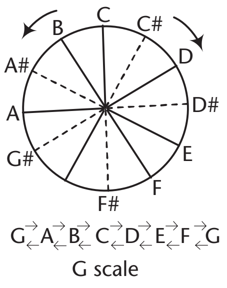


Figure B. Rhythmic groups of words.



(1)

C scale	Notes used in the C scale							
	I	II	III	IV	V	VI	VII	I
Major	C	D	E	F	G	A	B	C
Natural minor	C	D	D#	F	G	G#	A#	C
Harmonic minor	C	D	D#	F	G	G#	B	C
Melodic minor	C	D	D#	F	G	A	B	C

(2)

Figure C. Succession of (1) musical notes and (2) a music scale.

Popular song structure

Popular music's structure[2] often contains the intro, verse, chorus, bridge, middle eight, and outro. The intro may be 2, 4, 8, or 16 bars long; occasionally, there's no intro in a song. The intro is usually instrumental music. Both the verse and chorus are 8 or 16 bars long. Typically the verse is not melodically as strong as the chorus, but in some songs the verse is equally strong and most people can easily hum or sing it.

The gap between the verse and chorus is linked by a bridge. Silence may act as a bridge between the verse and chorus of a song, but such cases are rare. The middle eight—which is 4 or 16 bars long—is an alternative version of the verse with a new chord progression possibly modulated with different keys.

The instrumental sections in the song can be instrumental versions of the chorus or verse or entirely different tunes with a set of chords together. The outro is the fade-out of the last phrases of the chorus. These parts of the song are commonly arranged simply—verse, chorus, and so on in a repeated pattern. Three variations of this theme are discussed in the "Music structure detection" section in the main text.

References

1. The Associated Board of the Royal Schools of Music, *Rudiments and Theory of Music*, 1949.
2. "Ten Minute Master No. 18: Song Structure," *Music Tech*, Oct. 2003, pp. 62-63; http://www.musictechmag.co.uk.

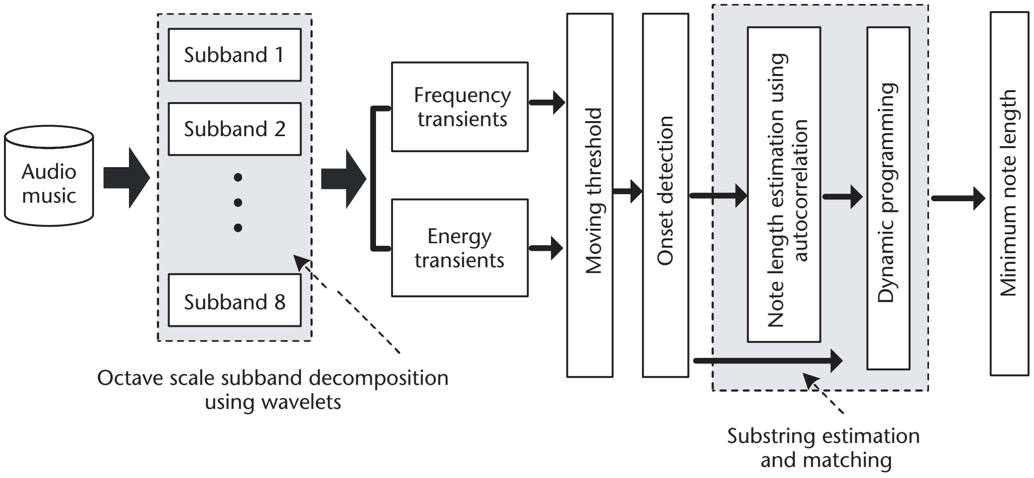


Figure 1. Rhythm tracking and extraction.

locally normalized before measuring the distances between the spectrums. The energy transients are computed for subbands 05 to 08.

Final onsets are computed by taking the weighted sum of onsets detected in each subband. We took weighted summations over subbands' onsets to find the final series of onsets. Music theories describe metrical structure as alternating strong and weak beats over time. Both strong and weak beats indicate the bar and note level time information. We estimated the initial interbeat length by taking the autocorrelation over the detected onsets. We employed a dynamic programming approach to check for patterns of equally spaced strong and weak beats among the computed onsets. Because our main purpose of onset detection was to calculate the interbeat timing and note level timing, we didn't need to detect all of the song's onsets. Figures 3a, 3b, and 3c show detected onsets, autocorrelation over detected onsets, and both sixteenth-note –level sementation and a bar measure of a clip.

After BSS, we detect the silence frames and remove them. Silence is defined as a segment of imperceptible music, including unnoticeable noise and short clicks. Short-time energy analysis over frames is employed for detecting silence frames. We further analyze the nonsilent beat-space-segmented frames in the following sections for chord and singing-voice boundary detection.

Chord detection

As we previously discussed, a chord is constructed by playing more than two music notes simultaneously. Thus, detecting the fundamental frequencies (F0s) of notes that comprise a chord is the key idea to identify the chord.

Chord detection is essential to identify melody-based similarity regions that have similar chord patterns. The vocal content in these regions may be different. Therefore, in some songs, both the verse and chorus have a similar melody.

The pitch class profile (PCP) features[3]—which are highly sensitive to the F0s of notes—are extracted from training samples to model the chord with a hidden Markov model (HMM). The polyphonic music contains signals of different music notes played at lower and higher octaves. Some music instruments (such as string instruments) have a strong third harmonic component[4] that nearly overlaps with the eighth semitone of the next high octave. This will lead to the wrong chord detection. For example, the third harmonic of note C in (C3 ~ B3) and F0 of

Related Work

Many researchers have attempted music structure analysis, with varying degrees of success. Cooper[1] analyzed how rhythm is perceived and established in the mind. Dennenberg[2] proposed chroma- and autocorrelation-based techniques to detect the melody line in the music. Repeated segments in the music are identified using Euclidean-distance similarity matching and clustering the music segments.

Goto[3] and Bartsch[4] constructed vectors from extracted pitch-sensitive, chroma-based features and measured the similarities between these vectors to find the repeating sections (the chorus) of the music. Foote and Cooper[5] extracted mel-frequency cepstral coefficients (MFCCs) and constructed a similarity matrix to compute the most salient sections in the music. Cooper[6] extracted MFCCs from the music content and reduced the vector dimensions using singular value decomposition techniques. Then he defined a global similarity function to find the most salient music section.

Logan[7] used clustering and hidden Markov models (HMMs) to detect the key phrases that were the most repetitive sections in the song. For automatic music summarization, Lu[8] extracted octave-based spectral contrast and MFCCs to characterize the music signals; the music's most salient segment was detected based on its occurrence frequency. Then the music signal was filtered using the band-pass filter in the frequency range of 125 ~ 1,000 Hz to find the music phrase boundary. He used these boundaries to ensure that an extracted summary didn't break the music phrase.

Xu[9] analyzed the signal in both time and frequency domains using linear prediction coefficients and MFCCs. An adaptive clustering method was introduced to find the salient sections in the music. Chai[10] generated music thumbnails by representing music signals with pitch-, spectral-, and chroma-based features and then matching their similarities using dynamic programming.

References

1. G. Cooper and L.B. Meyer, *The Rhythmic Structure of Music*, Univ. of Chicago Press, 1960.
2. R.B. Dannenberg and N. Hu, "Discovering Music Structure in Audio Recording," *Proc. 2nd Int'l Conf. Music and Artificial Intelligence*, 2002, pp. 43-57.
3. M.A. Goto, "Chorus-Section Detecting Method for Musical Audio Signals," *Proc. IEEE Int'l Conf. Acoustics Speech and Signal Processing*, IEEE CS Press, 2003.
4. M.A. Bartsch and G.H. Wakefield, "To Catch a Chorus: Using Chroma-Based Representations for Audio Thumbnailing," *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, IEEE CS Press, 2001.
5. J. Foote, M. Cooper, and A. Girgensohn, "Creating Music Videos Using Automatic Media Analysis," *Proc. ACM Multimedia*, ACM Press, 2002, pp. 553-560.
6. J.R. Deller, J.H.L. Hansen, and H.J.G. Proakis, *Discrete-Time Processing of Speech Signals*, IEEE Press, 1999.
7. B. Logan and S. Chu, "Music Summarization Using Key Phrases," *Proc. IEEE Int'l Conf. Acoustics Speech and Signal Processing*, 2000.
8. L. Lu and H. Zhang, "Automated Extraction of Music Snippets," *Proc. ACM Multimedia*, ACM Press, 2003, pp. 140-147.
9. C.S. Xu, N.C. Maddage, and X. Shao, "Automatic Music Classification and Summarization," *IEEE Trans. Speech and Audio Processing*, vol. 13, 2005, pp. 441-450.
10. W. Chai and B. Vercoe, "Music Thumbnailing via Structural Analysis," *Proc. ACM Multimedia*, ACM Press, 2003, pp. 223-226.

Subband		01		02	03	04	05	06	07	08
Octave scale		~B1	C2 ~ B2	C3 ~ B3	C4 ~ B4	C5 ~ B5	C6 ~ B6	C7 ~ B7	C8 ~ B8	All the higher octaves in the 8,192 ~ 22,050 Hz frequency range
Frequency			64 ~128	128~256	256~512	512~1024	1024~2048	2048~4096	4096~8192	
12 pitch-class notes	C	0 ~ 64	65.406	130.813	261.626	523.251	1046.502	2093.004	4186.008	
	C#		69.296	138.591	277.183	554.365	1108.730	2217.460	4434.920	
	D		73.416	146.832	293.665	587.330	1174.659	2349.318	4698.636	
	D#		77.782	155.563	311.127	622.254	1244.508	2489.016	4978.032	
	E		82.407	164.814	329.628	659.255	1318.510	2637.02	5274.04	
	F		87.307	174.614	349.228	698.456	1396.913	2793.826	5587.652	
	F#		92.499	184.997	369.994	739.989	1479.987	2959.956	5919.912	
	G		97.999	195.998	391.995	793.991	1567.982	3135.964	6271.928	
	G#		103.826	207.652	415.305	830.609	1661.219	3322.438	6644.876	
	A		110.000	220.000	440.00	880.000	1760.000	3520.000	7040.000	
	A#		116.541	233.082	466.164	932.328	1864.655	3729.310	7458.62	
	B		123.471	246.942	493.883	987.767	1975.533	395.066	7902.132	

Figure 2. Fundamental frequencies (F0) of music notes and their placement in the octave scale subbands.

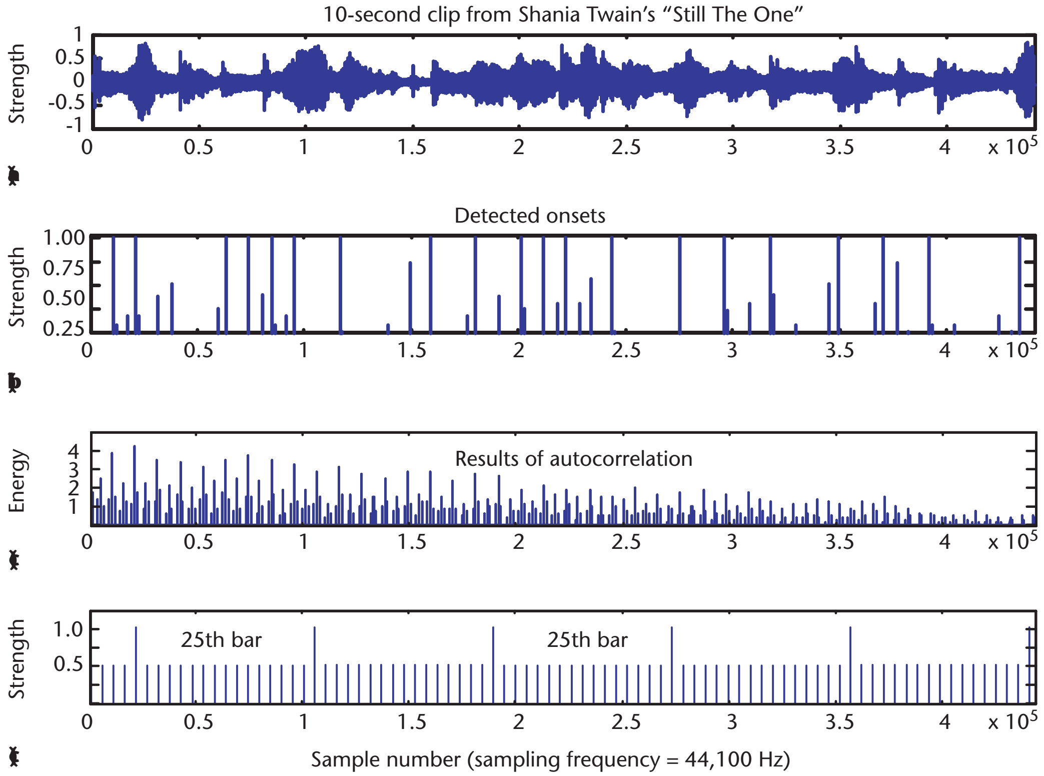


Figure 3. Clips lasting 44.39 ~ 54.39 seconds of Shania Twain's song, "Still the One." A sixteenth note's length is 112.10625 milliseconds (ms).

note G in (C4 ~ B4) nearly overlap (see Figure 2).

To overcome this, in our implementation BSS frames are represented in the frequency domain with a 2-Hz frequency resolution. Then the linear frequency is mapped into the octave scale, where the pitch of each semitone is represented with as high a resolution as 100 cents.

We consider the 128 ~ 8,192 Hz frequency range (subband 02 ~ 07 in Figure 2) to construct the PCP feature vectors to avoid percussion noise. We use 48 HMMs to model 12 major, 12 minor, 12 diminished, and 12 augmented chords. Each model has five states, including entry, exit, and three Gaussian mixtures (GM) for each hidden state.

We can see from Table 1 (next page) that the pitch difference between the notes of chord pairs is small. In our experiments, sometimes we find that the observed final-state probabilities of HMMs corresponding to these chord pairs are high and close to each other. This may lead to an incorrect chord detection. Thus we apply the rule-based method (key determination) to correct the detected chords and apply heuristic rules based on popular music composition to further correct the time alignment (transition) of the chords.

Songwriters use relative major and minor key combinations in different sections—perhaps a minor key for the middle eight and major key for the rest—which would break up the monotony of the song. Therefore, a 16-bar length with a 14-bar overlap window is run over the detected chords to determine the key of that section. The majority of chords that belong to a key are assigned as the key of that section. The 16-bar length window is sufficient to identify the key.[5] If the middle eight is present, we can estimate the region where it appears in the song by detecting the key change. Once the key is determined, the error chord is corrected as follows:

- Normalize the observations of the 48 HMMs that represent 48 chords according to the highest probability observed from the error chord.
- If the observation is above a certain threshold and it's the highest observation among all the chords in a key, the error chord is replaced by the next highest observed chord that belongs to the same key.
- If there are no observations belonging to the key above the threshold, assign the previous chord.

The information carried by the music signal can be considered quasistationary between the interbeat intervals, because the melody transition occurs on the beat time. Thus, we apply the following chord knowledge[6] to correct the chord transition within the window:

- Chords are more likely to change on beat times than on other positions.
- Chords are more likely to change on half-note times than on other positions of beat times.
- Chords are more likely to change at the beginning of the measures (bars) than at other positions of half-note times.

Table 1. Correct classification in percentage for vocal and instrumental classes.

Feature	Filters	Coefficients	Pure Instrumental Vocals	Instrumental Mixed and Pure Vocals
OSCC	64	12	82.94	79.93
MFCC	36	24	75.56	74.81

Singing-voice boundary detection

For the similar melodies in the choruses, they may have different instrumentals set up to break the monotony in the song. For example, the first chorus may contain snare drums with piano music and the second chorus may progress with a bass, snare drums, and rhythm guitar. After detecting melody-based similarity regions, it's important to decide which regions have similar vocal contents. Therefore, singing-voice boundary detection is the first step to analyze the vocal content.

In previous works[7,8,15] related to singing-voice detection, researchers used fixed-length signal sementation and characterized the signal frame with speech-related features such as mel frequency Cepstral coefficients (MFCCs), energies, zero crossing, spectral flux, and modeled the features with statistical learning techniques (such as HMM, *K*-nearest neighbors, and thresholding). However, none of these methods used music knowledge.

In our method, we further analyze the BSS frames to detect the vocal and instrumental frames. The analysis of harmonic structures of music signals indicates that the frequency components are enveloped in octaves. However, the similar spectral envelopes can't be seen in the speech signal's spectrum.

Thus, we use a frequency-scaling called the octave scale instead of the mel scale to calculate Cepstral coefficients to represent the music content. Sung vocal lines always follow the instrumental line so that both pitch and harmonic structure variations are also in the octave scale.

In our approach, we divide the whole frequency band into eight subbands (the first row in Figure 2) corresponding to the octaves in the music. We considered the entire audible spectrum to accommodate the harmonics (overtones) of the high tones. The range demanded of a voice's fundamental frequency in classical opera is from ~80 to 1,200 Hz, corresponding to the low end of the bass voice and the high end of the soprano voice. We empirically found that the number of octal-spaced triangular filters in each subband are {6, 8, 12, 12, 8, 8, 6, 4} respectively. As we can see, the number of filters are maximum in the bands

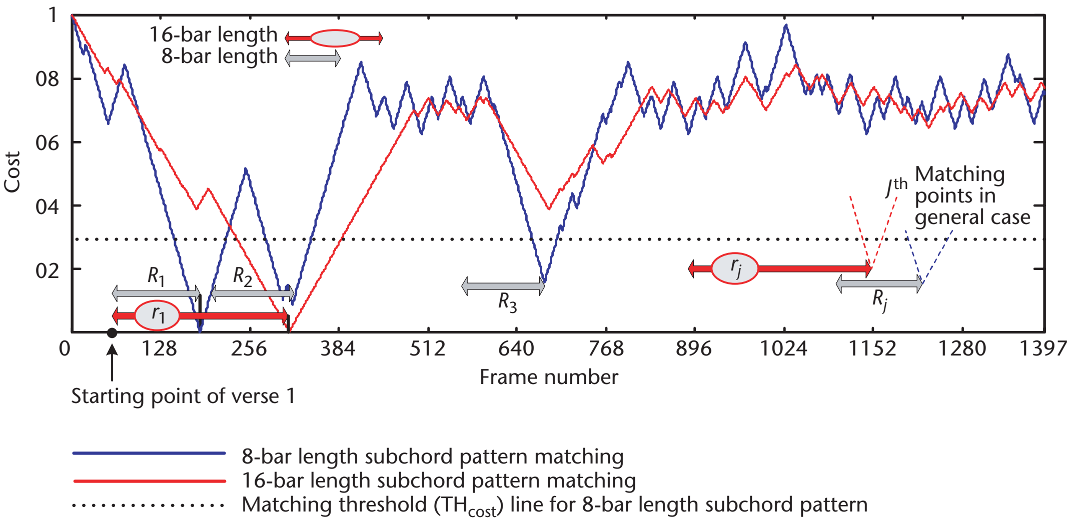


Figure 4. Both 8- and 16-bar-length chord patterns match in MLTR's song, "Twenty-Five Minutes." The notations R_1, R_2, ... R_j are further used as general melody-based similarity regions to explain our structure detection algorithm explicitly.

where the majority of the singing voice is present for better signal resolution in that range.

Cepstral coefficients are then extracted from the octave scale to characterize music content. These Cepstral coefficients are called octave scale Cepstral coefficients (OSCCs). Singular values indicate the variance of the corresponding structure. Comparatively high singular values describe the number of dimensions in which the structure can be represented orthogonally, while smaller singular values indicate the correlated information in the structure.

When the structure changes, these singular values also vary accordingly. However, we found that singular value variation is smaller in OSCCs than in MFCCs for both pure vocal music and vocal-mixed instrumental music. This implies that OSCCs are more sensitive to vocals than vocal-mixed instrumental music.

We applied singular value decomposition to find the uncorrelated Cepstral coefficients for the octave scale. We used the order range of 10 to 16 coefficients. Then we trained the support vector machine to identify the pure instrumental (PI) and instrumental mixed vocal (IMV) frames. Our earlier experimental results show that the radial-based kernel function in Equation 1 with c = 0.65, performs better in vocal/instrumental boundary detection:

$$K(x, y) = \exp(-|x - y|^2/c) \quad (1)$$

Song structure detection

We extract the high-level song structure based on melody-based similarity regions detected according to chord transition patterns and content-based similarity regions detected according to singing voice boundaries. Later, we explain how to detect melody- and content-based similarity regions in the music. Then we apply the song composition knowledge to detect the song structure.

Melody-based similarity region detection

The repeating chord patterns form the melody-based similarity regions. We employ a chord pattern-matching technique using dynamic programming to find the melody-based similarity regions. In Figure 4, regions R_2, ..., R_3, have the same chord pattern (similar melody) as R_1. Since it's difficult to detect all the chords correctly, the matching cost is not zero. Thus, we normalized the costs and set a threshold (TH_{cost}) to find the local matching points closer to zero (see Figure 4). $TH_{cost} = 0.3825$ gives good results in our experiments. By counting the same number of frames as in the subpattern backward from the matching point, we detect the melody-based similarity regions.

Figure 4 illustrates the matching of both 8- and 16-bar length chord patterns extracted from the beginning of verse 1 in MLTR's song, "Twenty-Five Minutes." The Y-axis is the normalized cost of matching the pattern and the X-axis is the frame number. We set the threshold TH_{cost} and analyzed the matching cost below the threshold to find the pattern-matching points in the song. The 8-bar-length regions ($R_2 \sim R_3$) have the same chord pattern as the first 8-bar chord pattern (R_1) in verse 1. When the matching pattern was extended to 16 bars (that is, the R_1 region), we weren't able to find a 16-bar-length region with the same chord pattern as the R_1 region.

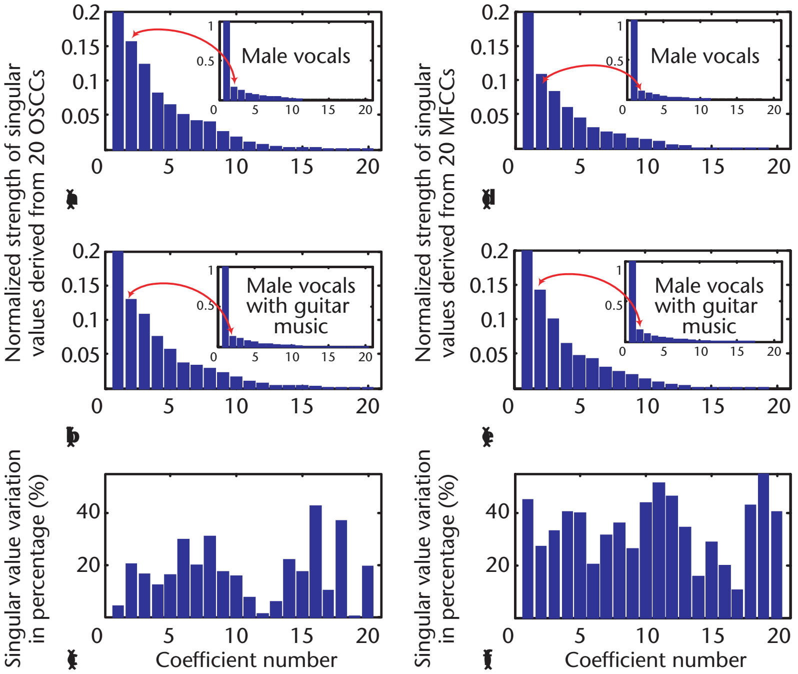


Figure 5. Analysis of singular values derived for octave scale Cepstral coefficients (OSCCs) and mel-frequency Cepstral coefficients (MFCCs).

Content-based similarity region detection

For the melody-based similarity regions R_i and R_j, we used the following steps to further analyze them for content-based similarity region detection.

Step 1. The BSS vocal frames of two regions are first subsegmented into 30-ms subframes with 50 percent overlap. Although two choruses have similar vocal content, they may have the same melody with a different instrumental setup. Therefore, we extracted 20 coefficients of the OSCC feature per subframe, since OSCCs are highly sensitive to vocal content and not to the instrumental melody changes. Figure 5 illustrates singular values derived from analyzing the OSCCs and MFCCs extracted from both the solo male track and guitar mixed male vocals of a Sri Lankan song “Ma Bala Kale (මා බාල කාලේ).”

The quarter-note length is 662 ms and the subframe size is 30 ms with a 50-percent overlap. Figures 5a, 5b, 5d, and 5e show the singular value variation of 20 OSCCs and 20 MFCCs for both pure vocals and the vocals mixed with guitar. Figures 5c and 5f show the percentage variation of the singular values of each OSCC and MFCC when guitar music is mixed with respect to their values for solo vocals.

When all 20 coefficients are considered, the average singular value variation for OSCC and MFCC are 17.18 and 34.35 percent, respectively. When the first 10 coefficients are considered, they are 18.16 percent and 34.25 percent. We can see that even when the guitar music is mixed with vocals, the variation of OSCCs is much lower than the variation of MFCCs. Thus, compared with MFCCs, OSCCs are more sensitive to the vocal line than to the instrumental music.

Step 2. The distance and dissimilarity between feature vectors of R_i and R_j are calculated using Equations 2 and 3. The dissimilarity *(R_i R_j)* gives low value for the content-based similarity region pairs.

$$\text{dist}_{R_iR_j}(k)=\frac{\left|V_i(k)-V_j(k)\right|}{\left|V_i(k)*\left|V_j(k)\right|\right|}\quad i\neq j \tag{2}$$

$$\text{dissimilarity}\left(R_i,R_j\right)=\sum_{k=1}^{n}\frac{dist_{RiRj}(k)}{n} \tag{3}$$

Step 3. To overcome the pattern-matching errors due to detected error chords, we shift the regions back and forth by four bars with two bars overlapping and repeat steps 1 and 2 to find the positions of the regions that give the minimum value for dissimilarity (R_i R_j).

Step 4. Calculate dissimilarity (R_i R_j) in all region pairs and normalize them. By setting a threshold (TH_{smlr}), the region pairs below the TH_{smlr} are detected as content-based similarity regions. This indicates that they belong to chorus regions. Based on our experiments, a value of $\text{TH}_{\text{smlr}} = 0.389$ works well. Figure 6 illustrates the content-based similarity region detection based on melody-based similarity region pairs.

Structure detection

We apply the following heuristics, which agree with most of the English-language songs we used to detect music structure.

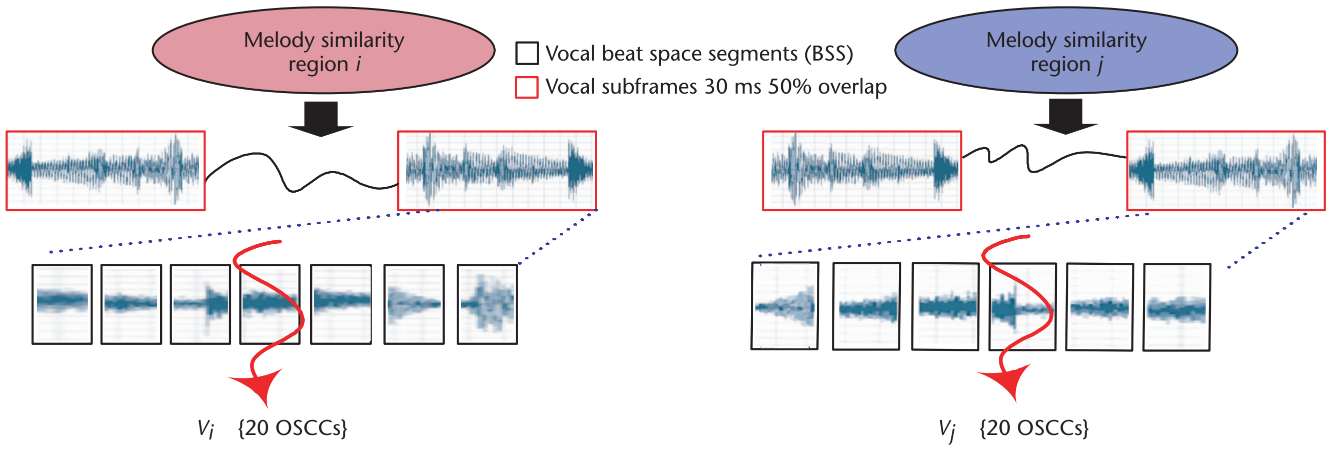
1. A typical song structure more or less uses one of the following verse–chorus patterns:[10]

   a. Intro, verse 1, chorus, verse 2, chorus, chorus, outro.

   b. Intro, verse 1, verse 2, chorus, verse 3, chorus, middle eight, chorus, chorus, outro.

   c. Intro, verse 1, verse 2, chorus, verse 3, middle eight, chorus, chorus, outro.

2. The minimum number of verses and choruses is two and three, respectively.



Melody similarity region *i*
Vocal beat space segments (BSS)
Vocal subframes 30 ms 50% overlap
Melody similarity region *j*
V_i {20 OSCCs}
V_j {20 OSCCs}

Figure 6. Content-based similarity region detection steps. Melody-based similarity regions R_i and R_j are framed to check for the vocal similarities between the regions.

3. The verse and chorus are 8 or 16 bars long.

4. The middle eight is 8 or 16 bars long.

The set of notes on which the piece is built is defined as the key. For example, the C major key is derived from the chords in the C major scale. Based on our data set—which only includes songs in English—the statement of "songs with multiple keys are rare" is true. But if this is extended to other language songs (such as Japanese songs), this statement may not be true. Therefore, we now avoid giving a false impression of generality by explicitly stating that the techniques and results presented here apply only to English-language pop songs.

Intro detection. According to the song structure, the intro section is located before verse 1. Thus we extract the instrumental section until the first vocal frame and detect this section as the intro. If silent frames are detected at the beginning, they aren't considered as part of the intro because they don't carry a melody.

Verses and chorus detection. Because the end of the intro is the beginning of verse 1, we assume the length of verse 1 is 8 or 16 bars and use this length-chord sequence to find the melody-based similarity regions in a song.

If only two or three melody-based similarity regions exist, they are the verses. Then we can conclude that the chorus doesn't have the same chord pattern as the verses. Cases 1 and 2 explain the detection of choruses and verses.

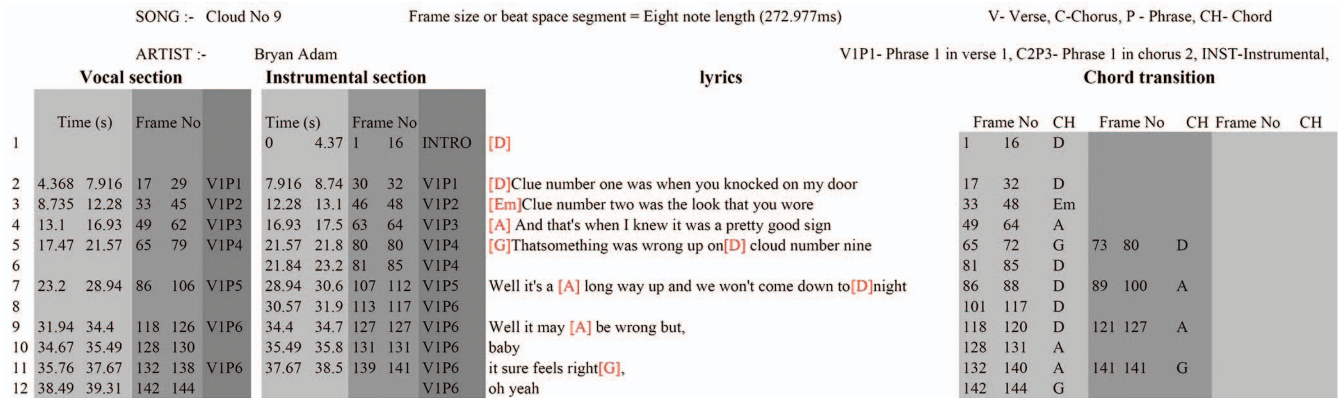
Case 1. The system finds two melody-based similarity regions. In this case, the song has the structure described in item 1a. If the gap between verses 1 and 2 is equal and more than 24 bars, both the verse and chorus are 16 bars long each. If the gap is less than 16 bars, both the verse and chorus are 8 bars long. Using the chord pattern of the first chorus between verses 1 and 2, we can detect other chorus regions. Because a bridge may appear between a verse and chorus or vice versa, we align the chorus by comparing the vocal similarities of the detected chorus regions.

Case 2. The system finds three melody similarity regions. In this case, the song follows the pattern either in item 1b or 1c. Thus, the first chorus appears between verses 2 and 3 and we can find other chorus sections using a procedure similar to that described in case 1.

If there are more than three melody-based similarity regions ($j > 3$ in Figure 4), it implies that the chorus chord pattern is partially or fully similar to the verse chord pattern. Thus we detect the 8-bar length chorus sections (which may not be the full length of the chorus) by analyzing the vocal similarities in the melody-based similarity regions. Cases 3 and 4 illustrate the detection of verse and chorus.

Case 3. If R_2 (Figure 4) is found to be a part of the chorus, the song follows the 1a pattern. If the gaps between R_1 and R_2 and R_2 and R_3 are more than 8 bars, the verse and chorus are 16 bars long. Thus we increase the subchord pattern length to 16 bars and detect the verse sections. After the verse sections are found, we can detect the chorus sections using a way similar to that in case 1.

Case 4. If R_2 is found to be a verse, the song follows the 1b or 1c pattern. The chorus appears after R_2 regions. By checking the gaps between R_1 and R_2 and R_2 and R_3, the length of the verse and chorus is similar to case 3. We can find the verse



SONG :- Cloud No 9 Frame size or beat space segment = Eight note length (272.977ms) V- Verse, C-Chorus, P - Phrase, CH- Chord

ARTIST :- Bryan Adam V1P1- Phrase 1 in verse 1, C2P3- Phrase 1 in chorus 2, INST-Instrumental,

	Vocal section					Instrumental section					lyrics	Chord transition							
	Time (s)		Frame No			Time (s)		Frame No				Frame No		CH	Frame No		CH	Frame No	CH
1						0	4.37	1	16	INTRO	[D]	1	16	D					
2	4.368	7.916	17	29	V1P1	7.916	8.74	30	32	V1P1	[D]Clue number one was when you knocked on my door	17	32	D					
3	8.735	12.28	33	45	V1P2	12.28	13.1	46	48	V1P2	[Em]Clue number two was the look that you wore	33	48	Em					
4	13.1	16.93	49	62	V1P3	16.93	17.5	63	64	V1P3	[A] And that's when I knew it was a pretty good sign	49	64	A					
5	17.47	21.57	65	79	V1P4	21.57	21.8	80	80	V1P4	[G]Thatsomething was wrong up on[D] cloud number nine	65	72	G	73	80	D		
6						21.84	23.2	81	85	V1P4		81	85	D					
7	23.2	28.94	86	106	V1P5	28.94	30.6	107	112	V1P5	Well it's a [A] long way up and we won't come down to[D]night	86	88	D	89	100	A		
8						30.57	31.9	113	117	V1P6		101	117	D					
9	31.94	34.4	118	126	V1P6	34.4	34.7	127	127	V1P6	Well it may [A] be wrong but,	118	120	D	121	127	A		
10	34.67	35.49	128	130		35.49	35.8	131	131	V1P6	baby	128	131	A					
11	35.76	37.67	132	138	V1P6	37.67	38.5	139	141	V1P6	it sure feels right[G],	132	140	A	141	141	G		
12	38.49	39.31	142	144						V1P6	oh yeah	142	144	G					

Figure 7. Manual annotation of the intro and verse 1 of Bryan Adams' song, "Cloud No. 9."

and chorus regions by applying procedures similar to those described in cases 3 and 1.

Instrumental sections (INSTs) detection. The Instrumental section may have a similar melody to the chorus or verse. Therefore, the melody-based similarity regions that have only instrumental music are detected as INSTs. However some INSTs have a different melody. In this case, we use a window of four bars to find regions that have INSTs.

Middle-eight and bridge detection. The middle eighth is 8 or 16 bars long and it has a different key from the main key. If a different key from the main key of the song is detected at any point, we further check whether the key changed area has a 16- or 8-bar length.

Once the boundaries of verses, choruses, INSTs and middle eight are defined, the appearance of the bridge can be found by checking the gaps between these regions.

Outro detection. From the song patterns in items 1a, 1b, and 1c we can see that before the outro there's a chorus. Thus, we detect the outro based on the length between the end of the last chorus and the song.

Experimental results

We used 50 popular English-language songs (by the following artists: MLTR, Bryan Adams, Westlife, the Backstreet Boys, the Beatles, and Shania Twain) for the experiments in chord detection, singing-voice boundary detection, and song-structure detection. We first sampled the songs at 44.1 kHz with 16 bits per sample and stereo format from commercial music CDs. We manually annotated the songs by conducting listening tests with the aid of commercially available music sheets to identify the timing of vocal/instrumental boundaries, chord transitions, the key, and song structure in terms of BSS units (the number of frames).

Figure 7 shows one example of a manually annotated song section, explaining how the music phrases and the chords change with inter-beat length. This annotation describes the time information of the intro, verse, chorus, instrumental, and outro in terms of 272.977-ms frames.

The frame length is equal to an eighth-note's length and it's the smallest note length found in the song. The beat-space measures of vocal and instrumental parts in the respective phrases (in the Lyrics column) are described in the Vocal and Instrumental columns. Then the system detected the silence frames (rest notes)—which may contain unnoticeable noise—by the frames' characteristic lower short-time energies.

Chord detection

We model 48 chords with HMMs. We use the first 35 songs (1 to 35) for training and the last 15 songs (36 to 50) for testing. Then we repeat the training and testing with different circular combinations, such as songs 16 to 50 for training and songs 1 to 15 for testing.

Because we don't have enough training chord samples in the songs for training the chord models we use the additional training data from the chord database. Thus, we have more than 10 minutes for each chord sample data for training each HMM. Our chord database consists of different sets of chords generated from original instruments (the piano, bass guitar, rhythm guitar, and so on), synthetic instruments (Roland RS-70 synthesizer or Cakewalk's software), and the system synthetically mixes instrumental notes by changing the time delay of the corresponding notes. It also synthetically mixes male

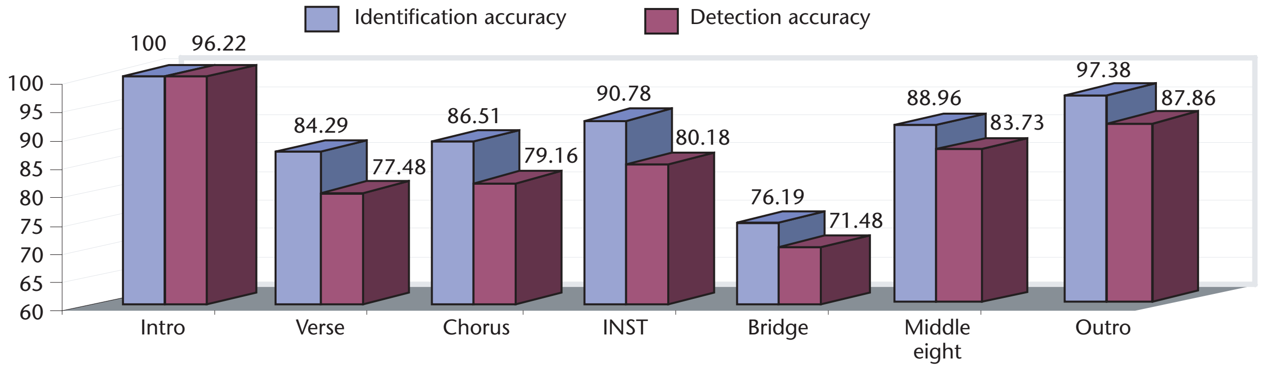


Figure 8. Average detection accuracies of different sections.

and female vocal notes. The recorded instrumental chords span from C3 to B6, comprising four octaves.

The average frame-based accuracy of chord detection is 80.87 percent. We can also determine the correct key of all the songs. After error correction with key information, we can achieve 85.49 percent frame-based accuracy.

Singing voice boundary detection

We use the Support Vector Machine (SVM) to classify frames into a vocal or instrumental class. The support vectors are trained with 12 OSCCs extracted from each nonoverlapping BSS.

The system uses the radial-based function in the SVM kernel. The parameters used to tune OSCCs are the number of filters and their distribution in the octave frequency scale. The 30 songs for SVM training and 20 songs for testing are employed with four different song combinations to evaluate the accuracy. Table 1 illustrates the comparison of the average frame-based classification accuracy of OSCCs and MFCCs. We empirically found that both the number of filters and coefficients of the feature give the best performance in classifying instrumental frames. OSCCs achieve better accuracy in this task.

We further applied music knowledge and heuristic rules[10] to correct the errors of misclassified vocal/instrument frames. With rules, the classification accuracy is significantly improved by 2.5 ~ 5.0 percent for both vocal and instrumental frames after applying rule-based error corrections.

Intro/verse/chorus/bridge/outro detection

We used two criteria to evaluate the results of the detected music structures:

- First, we used the accuracy of all the parts in the music identified. For example, if two-thirds of the choruses are identified in a song, the accuracy of identifying the choruses is 66.66 percent.

- Second, we used the accuracy of the sections detected. We illustrate the detection accuracy of a section in Equation 4. For example, if the detection accuracies of three chorus sections are 80.0 percent, 89.0, percent, and 0.0 percent, the average detection accuracy of the chorus section is $(80 + 89 + 0)/3 = 56.33$ percent.

$$\text{Detection accuracy of a section } (\%)\} = \frac{\text{length of correctly detected section}}{\text{correct length}} * 100 \quad (4)$$

Figure 8 illustrates our experimental results for the average accuracy detection of different sections. We can see that the the system detects the intro (I) and outro (O) with high accuracy. But detection accuracy for the bridge (B) sections is the lowest.

We compared our chorus-detection method with an earlier method[11] using our testing data set. Using the previous method, we reported identification and detection accuracies of 67.83 and 70.18 percent, respectively.

Applications

People can use music structure analysis for many applications, whether it involves music handling (such as music transcription), summarization, information retrieval, or streaming.

Music transcription and lyrics identification

Both rhythm extraction and vocal-/instrumental boundary detection are the preliminary steps toward lyric identification and music transcription applications. Because music phrases are constructed with rhythmically spoken lyrics,[12] we could use rhythm analysis and BSS to identify the word boundary in the polyphonic music signal. Along with signal separation techniques, this can reduce the complexity of identifying the voiced/unvoiced regions within the signal and make the lyric-identification process simpler. In addition, the chord detection extracts the pitch/melody contour in the music. Further analysis of BSS music signals will help to estimate the signal source mixture, which is the breaking point of music transcription.

Music summarization

The creation of a concise and informative extraction that accurately summarizes original digital content is extremely important in large-scale information organization and processing. Today, most music summaries used commercially are manually produced.

Music summaries are created based on the most repeated section, which is the most memorable or distinguishable part in a song. Based on successful music structure analysis, we can generate music summaries efficiently. For example, when we consider what we know about music, the chorus sections are usually the most repeated sections in popular music. Therefore, if we can accurately detect the chorus in each song, it is likely that we've also identified a good music summary.

Music information retrieval

Ever-increasing music collections require efficient and intuitive methods of searching and browsing. Music information retrieval (MIR) explores how a music database might best be searched by providing input queries in some music form. For people who aren't trained or educated with music theory, humming is the most natural way to formulate music queries. In most MIR systems, a fundamental frequency tracking algorithm parses a sung query for melody content.[13] The resulting melodic information searches a music database using either string-matching techniques or other models such as HMMs.

However, a problem for query by humming is that the hummed melody can correspond to any part of the target melody (not just at the beginning), which makes it difficult to find the matched starting point in the target melody. If we can detect the chorus accurately in a song, the location problem can be simpler. Because the choruses of popular songs are typically prominent and are generally sections that are readily recognized or remembered, the users are most likely to hum a fragment of the chorus. Furthermore, since the chord sequences are a description that captures much of the character of a song, and the chord pattern changes periodically for a certain song, we can match the chords with our input humming, which will facilitate the retrieval process.

Music streaming

Continuous media streaming over unreliable networks like the Internet and wireless networks may encounter packet losses because of mismatches between the source coding and channel characteristics. The objective of packet-loss recovery in music streaming is to reconstruct a lost packet so that it's perceptually indistinguishable or sufficiently similar to the original one.

Existing error-concealment schemes[14] mainly employ either packet-repetition or signal-restoration techniques. The most recently proposed content-based unequal error-protection technique[14] effectively repairs the lost packets that have percussion signals. However, this method is inefficient in repairing lost packets that contain signals other than percussion sounds (such as vocal signals and string, bowing, and blowing types of instrumental signals). Therefore, we need to be able to identify the music structure to construct an efficient packet-loss recovery scheme. The instrumental- and vocal-boundary detection simplifies the signal content analysis at the sender's end.

Such analysis along with pitch information (the melody contour) is helpful for better signal restoration at the receiver's side. We can construe a content-based similarity region identification to be a type of a music signal compression scheme. Because structure analysis helps identify content-based similarity regions such as the chorus and instrumental music sections, we can avoid retransmitting packets from similar regions and reduce the bandwidth consumption. Compared to conventional audio compression techniques such as MP3s (which can attain a 5:1 compression ratio), using music structure analysis we can potentially increase the compression ratio to 10:1.

Concluding remarks

By combining high-level music knowledge with existing audio-processing techniques, our system provides an efficient structural analysis approach for popular music. Our approach aims to extract the basic ingredients of music structures that can immensely simplify the development of many applications. In fact, a colleague at our lab is looking at polyphonic content-based audio retrieval based on our structural analysis. The initial results are promising.

Based on our current work, we plan to extend structure analysis to other music genres (such as classical or jazz) to come up with a broader music structure analysis approach. We also plan to explore more applications using music structure information, such as music genre classification and digital music watermarking. **MM**

Acknowledgments

I would like to thank Changsheng Xu, Mohan S. Kankanhalli, and Xi Shao for their comments and suggestions for this article.

References

1. N.C. Maddage et al., “Content-Based Music Structure Analysis with Applications to Music Semantic Understanding,” *Proc. ACM Multimedia*, ACM Press, 2004, pp. 112-119.
2. C. Duxburg, M. Sandle, and M. Davies, “A Hybrid Approach to Musical Note Onset Detection,” *Proc. Int’l Conf. Digital Audio Effects*, 2002.
3. A. Sheh and D.P.W. Ellis, “Chord Segmentation and Recognition Using EM-Trained Hidden Markov Models,” *Proc. Int’l Conf. Music Information Retrieval*, 2003.
4. T.D. Rossing, F.R. Moore, and P.A. Wheeler, *Science of Sound*, 3rd ed., Addison Wesley, 2001.
5. A. Shenoy, R. Mohapatra, and Y. Wang, “Key Detection of Acoustic Musical Signals,” *Proc. IEEE Int’l Conf. Multimedia and Expo*, IEEE CS Press, 2004.
6. M. Goto, “An Audio-Based Real-Time Beat Tracking System for Music with or without Drum-Sounds,” *J. New Music Research*, vol. 30, no. 2, 2001, pp. 159-171.
7. Y.K. Kim and Y. Brian, “Singer Identification in Popular Music Recordings Using Voice Coding Features,” *Proc. Int’l Conf. Music Information Retrieval*, 2002.
8. T. Zhang, “Automatic Singer Identification,” *Proc. IEEE Int’l Conf. Multimedia and Expo*, IEEE CS Press, 2003.
9. C.S. Xu, N.C. Maddage, and X. Shao, “Automatic Music Classification and Summarization,” *IEEE Trans. Speech and Audio Processing*, IEEE CS Press, vol. 13, 2005, pp. 441-450.
10. “Ten Minute Master No. 18: Song Structure,” *Music Tech*, Oct. 2003, pp. 62-63; http://www.musictechmag.co.uk.
11. M.A. Goto, “Chorus-Section Detecting Method for Musical Audio Signals,” *Proc. IEEE Int’l Conf. Acoustics Speech and Signal Processing*, IEEE CS Press, 2003.
12. The Associated Board of the Royal Schools of Music, *Rudiments and Theory of Music*, 1949.
13. A. Ghias et al., “Query by Humming: Musical Information Retrieval in an Audio Database,” *Proc. ACM Multimedia*, ACM Press, 1995, pp. 231-236.
14. Y. Wang et al., “Content-Based UEP: A New Scheme for Packet Loss Recovery in Music Streaming,” *Proc. ACM Multimedia*, ACM Press, 2003.
15. A.L. Berenzweig and D.P.W. Ellis, “Location Singing Voice Segments within Music Signals,” *Proc. IEEE Workshop on Application of Signal Processing to Audio and Acoustics*, IEEE CS Press, 2001, pp. 119-122.

Namunu C. Maddage is an associate scientist in the Speech and Dialog Lab at the Institute for Infocomm Research of Singapore. His current research interests are in music content analysis and audio data mining. Maddage received his PhD in computing from the National University of Singapore.

Readers may contact Namunu C. Maddage at maddage@i2r.a-star.edu.sg.