

Repeating Pattern Discovery and Structure Analysis from Acoustic Music Data

Lie Lu¹, Muyuan Wang², Hong-Jiang Zhang¹

¹Microsoft Research Asia
Beijing, P.R. China, 100080

{llu, hjzhang}@microsoft.com

²Department of Automation, Tsinghua University
Beijing, P.R.China, 100084

wmy99@mails.tsinghua.edu.cn

ABSTRACT

Music and songs usually have repeating patterns and prominent structure. The automatic extraction of such repeating patterns and structure is useful for further music summarization, indexing and retrieval. In this paper, an effective approach of repeating pattern discovery and structure analysis of acoustic music data is proposed. In order to represent the melody similarity more accurately, in our approach, Constant Q transform is utilized in feature extraction and a novel similarity measure between musical features is proposed. From the self-similarity matrix of the music, an adaptive method is then presented to extract all significant repeating patterns. Based on the obtained repetitions, musical structure is further analyzed using a few heuristic rules. Finally, an optimization-based approach is proposed to determine the accurate boundary of each musical section. Evaluations on various music pieces indicate our approach is promising.

Categories and Subject Descriptors

H.5.5 [Information Interfaces and Presentation]: Sound and Music Computing - *signal analysis, synthesis and processing; systems*; H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing - *indexing methods*.

General Terms

Algorithms, Management, Design, Experimentation

Keywords

Music structure, repeating pattern, CQT, structure-based distance measure

1. INTRODUCTION

Music generally shows strong self-similarity, and thus has some repeating patterns and prominently repetitive structure. These repeating patterns and structure are very helpful for further music analysis such as music snippet [1] or music thumbnail [6], music summarization [2][5], and music retrieval. However, few

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MIR'04, October 15-16, 2004, New York, New York, USA

Copyright 2004 ACM 1-58113-940-3/04/0010...\$5.00

literatures have fully addressed this issue from acoustic musical data. Several published works on repeating pattern discovery are all for MIDI data [3][4], which are not practical in real acoustic music processing. Some works relevant to repeating pattern analysis from acoustic data can be found in works on music summarization and music thumbnail, as one step towards the objective. In works [1] and [2], a clustering method or Hidden Markov Model (HMM) is utilized to group the segments with similar characteristics. Cooper [5] also presents a method to find given-length repetitions, by employing a 2D similarity matrix. In [6], Bartsch proposes an approach to catch chorus, by using a new feature set, quantized chromagram, to represent the spectral energy at each twelve pitch classes. Goto [7] also uses chroma features to detect chorus sections for musical audio signal and further developed a way to detect the modulated repetitions.

However, most of the above algorithms are designed to extract one segment of chorus or thumbnail, they did not fully investigate all the repeating patterns in a music piece. In this paper, a new approach is proposed to extract all the significant repetitions that have similar melody. In order to represent the melody similarity more accurately, Constant Q transform (CQT) [9] is utilized for feature extraction and a novel distance measure is proposed. CQT features represent the spectral energy at each exact note, so that it contains more information than chroma-based features and MFCC and thus is more suitable in our application. The proposed distance measure emphasizes more on melody similarity and suppresses timbre similarity. Thus it facilitates to find the repetition between two similar melodies played with different instruments.

Based on the results of repeating pattern analysis [14], we further design an algorithm to discover the structural information of a music piece, such as AABABB, which indicates the first music section is repeated at the second and fourth section while the third one is repeated at the fifth and sixth section. Chai [8] presents a preliminary approach to structural analysis. In this paper, a more complete investigation is presented. Besides repetitive structures, we also propose an optimization-based approach to determine the boundary of each section of the music structure.

The proposed approach to repeating pattern and music structure analysis is illustrated in the Fig. 1. First, each feature set is extracted from the acoustic data, including temporal feature, spectral feature and CQT feature. Temporal features are used to estimate tempo period and the length of a musical phrase, which is used as the minimum length of a significant repetition in repeating patterns discovery and boundary determination. Spectral features

are used for vocal and instrumental sounds discrimination in order to identify the *intro*, *interlude* and *coda* [15] of a popular song in final music structure analysis. CQT features are used to represent the note and melody information, based on which a self-similarity matrix of the music is obtained, using our novel distance measure. The significant repeating patterns are then detected from the similarity matrix with an adaptive threshold setting method. Finally, the boundaries of repeating patterns are roughly aligned to facilitate music structure inference; and the obtained structure is utilized correspondingly to refine the boundary of each musical section, with an optimization-based approach.

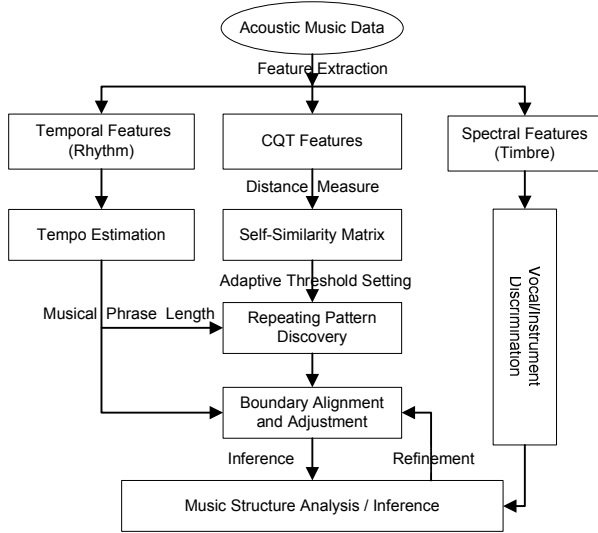


Fig. 1 A system framework of repeating pattern discovery and structure analysis from acoustic music data

The rest of the paper is organized as follows. Section 2 discusses the CQT features used in the algorithm. Section 3 presents our novel distance measure which emphasizes more on melody similarity and suppresses timbre similarity. Section 4 describes the approach to musical repeating pattern discovery, and Section 5 addresses the problem of musical structure analysis. Evaluations and discussions are presented in the Section 6.

2. CQT FEATURES

Human perception of repetitions in popular song is generally based on melody similarity but not timbre similarity. That is, we are going to discover melody repetition more than timbre repetition. Therefore, the extracted features and corresponding similarity measure should focus on melody similarity which is related to a sequence of note similarity, rather than timbre similarity. Ideally, music is converted into note sequence by multi-pitch analysis, and then melody similarity can be easily measured based on the explicit note sequence. However, music transcription is not feasible currently and most of the conventional features, such as Mel-Frequency Cepstral Coefficient (MFCC) [13], indicate more on timbre properties and could not represent note accurately. In order to extract acoustic features representing the music notes more accurately, constant Q transform (CQT) [9] is used in our approach. CQT has the ability to represent musical signal as a spectral sequence of exact musical notes, with a bank

of filters whose center frequencies are geometrically spaced. In our approach, the musical notes in 3 octaves, i. e. 36 semi-tones are extracted, as

$$X(k) = \frac{1}{N_k} \sum_{n=0}^{N_k-1} x(n) e^{-\frac{j2\pi Qn}{N_k}} \quad (1)$$

where $X(k)$ represents the spectral energy of the k -th note with the center frequency f_k ,

$$f_k = f_0 \cdot 2^{k/b}, \quad k = 0, 1, 2, \dots, 36 \quad (2)$$

and f_0 stands for the minimal frequency that we are interested in computing. It is chosen to be 130.8Hz as the pitch of C3, since most pitches in pop music are larger than it. b is set as 12 in order to obtain 12 semitones in an octave. Q is a constant ratio of frequency to resolution,

$$Q = f_k / (f_{k+1} - f_k) = 1 / (2^{1/12} - 1) \quad (3)$$

and accordingly, for the k -th filter, the window width N_k is set as:

$$N_k = \lfloor f_s Q / f_k \rfloor \quad (4)$$

where f_s denotes the sampling rate.

Compared to Discrete Fourier Transform (DFT), CQT uses geometrically spaced center frequencies, which are related to exact musical notes. Moreover, CQT has a finer resolution, and thus gives a better representation of music signals. The chroma algorithm [6][7] also has a similar idea as CQT and gives the spectral energy of 12 pitch classes. However, it is derived from DFT directly and ignores the difference between octaves. Thus, it does not have finer resolution and is not as accurate as the features obtained by CQT. Experiments also indicate that the CQT features perform better than MFCC and chroma features which are based on DFT.

Based on CQT, a feature vector of 36-dimension is extracted. In our approach, the feature vector is further normalized to be unit-norm in order to compensate for the effect of the amplitude variations.

3. DISTANCE MEASURE

As mentioned above, we are trying to measure the melody similarity rather than timbre similarity. Although the extracted features are more related to musical note and melody, we would also design a distance measure algorithm to focus more on note difference than timbre difference, in case that the same melody is played by different instruments in two different sections.

The timbre feature of a note is generally represented by the spectral energy at each of its harmonic partials which are the components of CQT feature vector. Consider two sounds with the same note but played by different instruments, they will have the same fundamental frequency but different timbre. However, conventional Euclidean distance or cosine distance considers the absolute value of the partial difference, and makes the distance between the same notes relatively large and thus cannot represent accurately the actual similarity between them.

Fig. 2(a) illustrates a self-similarity matrix based on the Euclidean distance among three notes, which includes D3 played by cello,

D3 by altotrombone, and D[#]3 by cello. The similarity scores are normalized to [0, 1], and brighter points represent more similar musical frames. From the matrix, it is noted that the similarity between two D3s played by different instruments is not prominently higher than that between D3 and D[#]3 played by cello, since the timbre difference is over-considered. Thus, it may introduce some noise in further repetition discovery.

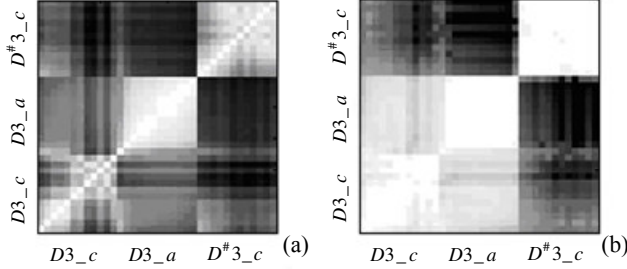


Fig. 2 Self-similarity matrices of three notes, which includes D3 played by cello(D3_c), D3 by altotrombone(D3_a), and D[#]3 by cello(D[#]3_c), using difference distance measure (a) Euclidean distance (b) Structure-based distance measure

In order to discriminate the note property from timbre property, the difference vector ΔV between two notes is examined, which is defined as follows,

$$\Delta V = V_1 - V_2 = [|v_{11} - v_{21}|, \dots, |v_{1N} - v_{2N}|] \quad (5)$$

where V_1 and V_2 are the feature vectors of two notes, and N is the dimension of the feature vector.

It is noted that the difference vectors have different structure properties in the case of timbre variation and note variation. For a difference vector between the same notes with different timbres, its spectral components are mostly placed at the positions of f_0 , $2f_0$, $3f_0$, etc, assuming f_0 is the fundamental frequency. Thus, the spectral peaks are mostly spaced with some prominent regular intervals, such as 12 semitones (octave), 7 semitones (perfect fifth) or 4 semitones (major third). For example, $2f_0$ is 12 semitones apart from f_0 , and the $3f_0$ is about 7 semitones apart from $2f_0$. These prominent regular intervals appearing in the difference vector of the same notes are called *harmonic interval* in the later of this paper for simplicity. However, the difference vector between two different notes has not such characteristic, as Fig. 3 illustrates.

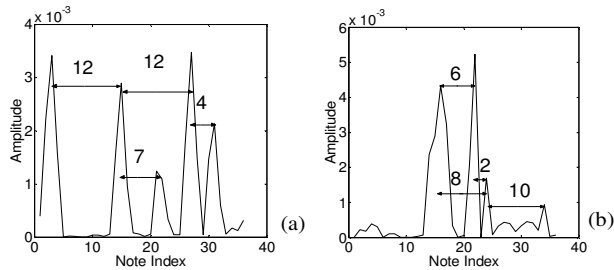


Fig. 3 Different structures of the difference vectors, which are between (a) D3 played by cello and by altotrombone; (b) D3 and D[#]3 played by cello

In Fig. 3, the left is the difference vector between the same note D3 played by cello and by altotrombone, and the right is that of different notes D3 and D[#]3 played by cello. It is noted that the peaks are mostly spaced by 12, 7 or 4 semitones in the left figure, while they are not in the right. However, the norms of these two vectors, which are the corresponding Euclidean distances, are almost the same, although the structures of these two vectors are completely different.

Although the above descriptions are for single notes, the difference vector between two chords also has similar property more or less, especially when the notes of a chord are perfect-fifth or major-third spaced.

3.1 Structure-based Distance Definition

From above section, it is clear that, in order to focus more on note difference than timbre difference, the distance measure had better be dependent on the structure of the difference vector but not just the norm of it. That is, if the spectral peaks in the difference vector are mostly apart with *harmonic intervals*, the two sounds are more likely from the same note, and the distance should be relatively small; otherwise, the distance should be large.

In order to describe the structure, i.e. the peak intervals in the difference vector, the autocorrelation is used as follows,

$$r(m) = \frac{\sum_{n=0}^{N-m-1} \Delta v_{n+m} \Delta v_n}{\sum_{n=0}^{N-m-1} \Delta v_n^2} \quad 0 \leq m \leq N-1 \quad (6)$$

where Δv_i is the i -th component of ΔV , and m is the interval index. $r(m)$ is the autocorrelation coefficient and can roughly represent the likelihood that the peaks in difference vector has a period of m . For example, the magnitude of $r(12)$ reflects the degree that the peaks are octave-spaced. Thus the structure is described as a vector containing all the coefficients,

$$R = [r(0), r(1), \dots, r(N-1)]^T \quad (7)$$

However, different coefficient should have different contribution in distance computation. For example, the coefficients with *harmonic intervals*, such as $r(12)$ or $r(7)$, represent the possibility that the two sounds are the same note, so they should be suppressed in the distance measure, in order to make timbre difference less important. Therefore, to reflect the contribution of various intervals, different weightings are given to different autocorrelation coefficients. Thus, the distance between the i -th and j -th musical frame can be estimated as,

$$d_{ij} = W^T R_{ij} \quad (8)$$

where R_{ij} is the corresponding structure between two frames, and $W = [w(0), w(1), \dots, w(N-1)]^T$ is a weighting vector, which is chosen in the next sub-section.

Actually, the above measure only considers the isolated two frames. In order to give a more comprehensive representation of the distance, it is desirable that their neighboring temporal frames in a window are taken into considerations, as the following ,

$$d'_{ij} = \frac{1}{2N_w} \sum_{k=-N_w}^{N_w-1} d_{i+k, j+k} \quad (9)$$

where $2N_w$ neighboring frames are also considered.

3.2 Weighting Determination

The basic rule in choosing the weightings is that, if the interval index of a coefficient is more possible to be a *harmonic interval*, the corresponding weighting should be smaller. For example, the weighting of $r(12)$ or $r(7)$ should be relatively small.

Although various weightings can be chosen, in our application, the spiral array model [10] established on music perception is utilized in weighting determination. The model maps each musical note onto a 3D helix, where adjacent notes are perfect-fifth (7 semitones) apart. Thus the order of notes on the spiral is: C, G, D, A, E, B, F#, C#, G#, D#, A#, F. It is noted that if the music interval between two notes is more possible to be *harmonic interval*, the distance between these two notes is smaller on the helix. Thus, the distance between notes with interval m can be utilized as the weighting of $r(m)$. However, on the helix, the adjacent notes are 7 semitones apart instead of 1 semitone, so we should re-order them to give an appropriate weighting, as

$$w(m) = \frac{1}{A} |P(7m \bmod 12) - P(0)| \quad (10)$$

where $P(m)$ is the position of m -th note and set as [10] suggested,

$$P(m) = \left[\sin \frac{m\pi}{2}, \cos \frac{m\pi}{2}, \frac{m}{2} \right] \quad (11)$$

and A is a normalization coefficient to satisfy $\sum w(m) = 1$. It is noted that the weighting for octave interval is set as 0, in order to further de-emphasize the effect of timbre difference.

Integrating these weightings into Eq(8) and Eq(9) obtains structure-based distance measure. Corresponding to Fig. 2(a), the similarity matrix based on new distance is shown in Fig. 2(b). It can be seen that the similarity between the same notes are more distinguish-able from those between different notes now.

4. REPEATING PATTERN DISCOVERY IN SIMILARITY MATRIX

Once the distance measure is given, a self-similarity matrix $S = \{S_{ij}\}$ can be computed from the whole music, with each S_{ij} is simply set as $1/d_{ij}$ in our approach. The repeating patterns are represented as the highlighted lines parallel to the diagonal, as Fig. 4 (a) shows. The brighter the line, the more similar two segments are; and the longer the line, the more significant the repeating pattern is.

In order not to trivialize the repetition detection, we assume that a significant repeating pattern at least has the length of a musical phrase. It is reasonable since most of the songs satisfy such an assumption. Based on some music theories, a musical phrase usually contains four or eight bars. Thus, tempo, which measures the duration of two contiguous beats, can be used to estimate the length of a musical phrase. In our approach, a similar algorithm as the work presented in [1] is employed for tempo estimation and musical phrase length estimation.

After the minimum length is given, the significant repetitions are enhanced and then all repeating patterns are explored with an adaptive threshold.

4.1 Erosion and Dilatation

For the convenience of processing, we map the similarity matrix into a *time-lag matrix* T [7], as

$$T_{i,l} = S_{i,i+l} \quad (12)$$

where $T_{i,l}$ represents the similarity between frame i and the frame $i+l$ which has lag l . Thus, the repeating patterns are converted to be parallel to the horizontal lines in the lower triangular time-lag matrix, as Fig. 4 (b) shows.

However, in the time-lag matrix, an actual repetition lines may be broken into several lines; and meanwhile, some short horizontal lines may also be introduced due to the noise, as illustrated in Fig. 4 (b). In order to further enhance the significant repetition lines, and remove the short lines which may be caused by noises, erosion and dilatation [11] which are common operations in grayscale image processing, are applied in our approach.

The erosion operation is used to replace a point with the minimum value in a range around it, as

$$T'_{i,j} = \min\{T_{i,j+k} \mid k \in [-L/2, L/2]\} \quad (13)$$

where L is the minimal length of repetition we want to target, which is adaptively set as the length of a musical phrase.

Correspondingly, the dilatation operation is used to replace a point with the maximum value in the range of L as

$$T'_{i,j} = \max\{T_{i,j+k} \mid k \in [-L/2, L/2]\} \quad (14)$$

Generally, erosion and dilatation is used sequentially to remove the short lines whose length is shorter than L . After these operations, the significant repetitions are enhanced and the short lines are weakened. Fig. 4 (c) illustrates the time-lag matrix after these operations.

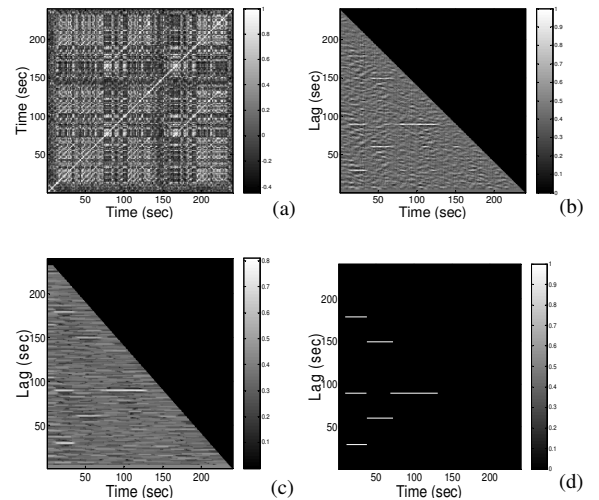


Fig. 4 Repeating pattern discovery of an example music clip. (a) The self-similarity matrix; (b) Corresponding time-lag matrix (c) Time-lag matrix after erosion and dilatation; (d) Optimal final results

4.2 Adaptive Threshold Setting

To this end, a threshold should be determined to discriminate the repetitions from non-repetitions. However, experiments indicate that the threshold is strongly dependent on the samples. It is not appropriate to use a constant threshold for all music pieces. Instead, we should determine it adaptively. In [7], a threshold is chosen by maximizing intra-class distance while minimizing inner-class distance. However, we found this method causes many false repetitions when dealing with our time-lag matrix, if the threshold is allowed to be chosen from the whole value domain of similarity levels. This is because, in our cases, the two classes are extremely unbalanced. The repetitions lines generally occupy less than 1% points of the whole matrix. Thus, the threshold should be chosen in a constrained range.

To solve this issue, we firstly estimate the probability distribution of similarity levels in the time-lag matrix. Considering the repetitions almost have the largest value but with a small number, a range of $[P^\alpha, P^\beta]$ in which a reasonable threshold may exist is estimated, where P^α and P^β stands for the *percentile* of probability distribution. For instance, $P^{0.99}$ represents a threshold classify 1% of points as repetitions. In our implementation, the range is experimentally chosen as $[P^{0.99}, P^{0.998}]$. Then, the optimal threshold is chosen in this range, based on the criterion that maximizes intra-class distance while minimizes inner-class distance.

After the threshold is determined, the time-lag matrix can be easily quantized to binary value (0, 1). Since the quantization will also cause some breaks in the repetition line, dilation and then erosion are used sequentially to remove the short breaks. The final time-lag matrix is shown in Fig. 4 (d), from which the repetitions can be easily detected.

Moreover, in our approach, if segment A is a repetition of segment B, while B is a repetition of C, it is assumed that A is also a repetition of C. Such assumption is utilized in case that not all of repetition pairs are completely detected.

5. MUSIC STRUCTURE ANALYSIS

After repeating patterns are obtained, musical structure can be correspondingly inferred from them. However, in previous processing, the boundary of obtained repetitions may be not aligned with each other, due to the errors introduced by erosion/dilation processing and binarization. Two examples are illustrated in Fig. 5, where each line shows a pair of repeating segments with a same color. Fig. 5(a) shows an example of start time shift between two segments, while in Fig. 5(b), the end time of two segments and the start time of another segment are overlapped. It is intuitively obvious that the segments in these two cases share the same boundary, if the shift or overlapping between the boundaries is short enough, for example, less than half of a musical phrase in our implementation. It should be noted that if the shift or overlapping is long enough, it will be identified as an individual section of a subtle structure (in Section 5.1) but not the one introduced by boundary misalignment.

In general, the optimal boundary of those misaligned segments can be selected from *uncertain area* determined by the boundary shift or boundary overlapping between them, as the Fig. 5 illustrates, where the uncertain area is marked with slash lines, such as $[T1, T2]$ in case (a) and $[T3, T4]$ in case (b).

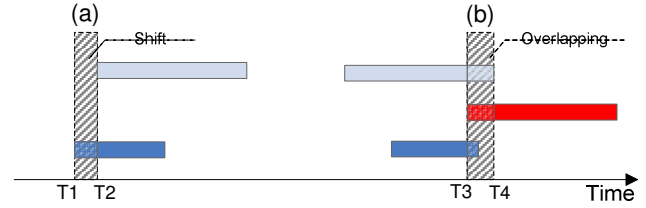


Fig. 5 An illustration on boundary misalignment. The region with slash lines is the uncertain area, from which the optimal boundary can be selected

It is better to align the boundary of the extracted repeating segments to facilitate further processing. However, in boundary alignment, the adjustment of one segment's boundary also affects the boundaries of its repetitions. It is difficult to find a global boundary optimization method without any overall structure information. In our approach, we firstly identify the uncertain area which includes the potential boundary, and roughly align the boundary of each segment with the boundary of corresponding uncertain area in order to facilitate further structure analysis, without considering the effects among one another. Then, the music structure is analyzed with some heuristic rules. After the music structure is obtained, the boundary of each repetition or section is refined with an optimization-based algorithm. And finally, the instrumental sections, including *intro*, *interlude* and *coda*, are identified to obtain a more comprehensive structure.

5.1 Structure Inference with Heuristic Rules

After the repeating patterns are detected and the boundary is preliminary aligned, we can label each repeating segments to obtain the musical structure. The basic rule is to give a same label to the segments which are repetitions of each other, from the beginning to the end of a song. This process is iteratively processed until all the repeating segments are labeled. If the all repeating segments are not overlapped with each other, the above process can be smoothly finished. However, some obtained segments are usually overlapped, due to the repetitive property of the music structure or the effect of a subtle structure. Fig. 6 shows two fundamental cases on segments overlapping, where case (a) shows two overlapped segments which are not repetitive with each other, while case (b) shows two overlapped repetitions. It indicates that the segments may be not an individual section in the structure but contain a more subtle structure. In these cases, some heuristic rules are utilized in our approach to label the structure.

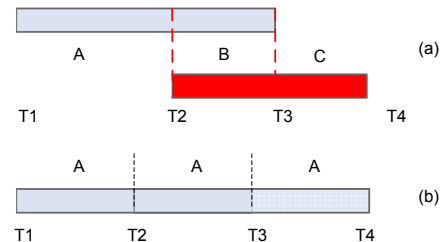


Fig. 6 Structure inference with segments overlapping (a) overlapped between two segments which are not repetitive (b) overlapped between two repetitions

5.1.1 Overlapped Non-Repetitions

Fig. 6 (a) shows two segments [T1, T3] and [T2, T4] overlaps, while these two segments are not repetitions of each other. It indicates that each segment is not an individual section in the structure, but may contain a more subtle structure and thus be composed of two sections. In this case, we will split the segments at point T2 and T3 and take segment [T2, T3] as an individual section. Thus the first segment is labeled as AB while the second segment is labeled as BC.

It is noted that the same rule is also feasible in more complex cases, such as more than two segments are overlapped or one segment is included in another segment (e.g., when T4 = T3).

5.1.2 Overlapped Repetitions

Fig. 6 (b) illustrates another case that two repeating segments are overlapped, where segment [T1, T3] is a repetition of [T2, T4] while they are overlapped at [T2, T3]. It indicates there is an internal repetition in each segment. For example, if the length of [T2, T3] is roughly equal to [T1, T2], each segment is actually composed of two repetitions of a subtle section such as AA.

To be more general, if the length of the repeating segment is multiples of the overlapped length, the segment is generally composed of multiple repetitions of a subtle section. The repetition number can be roughly estimated as,

$$N_r = \left\lceil \frac{T3-T1}{T3-T2} + 0.5 \right\rceil \quad (15)$$

5.2 Boundary Refinement

After the music structure is obtained, the accurate boundary of each section can be determined. Fig. 7 (a) illustrates an example result of structure analysis and the uncertain areas of boundary, where A and B represent the marked label of repeating sections, @ represents a section that only appears once and does not have any repetition, and the gray area with slash lines is the uncertain area from which the candidate boundary of each section can be selected. Suppose there are N sections in the music, there will be $N+1$ boundaries to be determined. Fig. 7 (a) also illustrates a candidate boundary sequence, which could be represented as,

$$\mathbf{B} = \langle b_1, b_2, \dots, b_{N+1} \rangle$$

where \mathbf{B} indicates a candidate boundary set, and b_i is the boundary between the $(i-1)$ -th and i -th section.

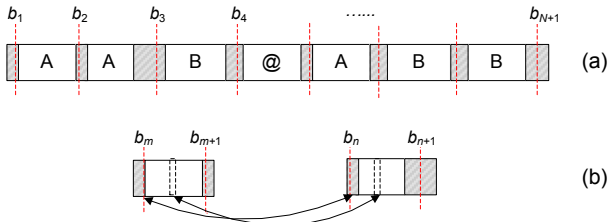


Fig. 7. Optimal boundary determination (a) an example result of structure and the uncertain boundary areas (b) similarity measure between two segments

Intuitively, an optimal boundary set should satisfy the following two conditions,

- 1) The optimal boundary set maximizes the similarity between every two sections with the same label.
- 2) The length of each section with the same label is roughly equal to each other.

To measure the similarity of two sections, the similarities between the corresponding points in these two sections are considered, as the Fig. 7 (b) shows. The similarity can be denoted as,

$$S(m, n) = \frac{1}{L} \sum_{i=0}^{L-1} S_{b_m+i, b_n+i} \quad (16)$$

where $L = \min\{L_m, L_n\}$, L_m and L_n is the length of the m -th and n -th section, with $L_m = b_{m+1} - b_m$ and $L_n = b_{n+1} - b_n$, and usually $L_m = L_n$.

Thus, given the candidate boundary set, the objective function for selecting the optimal boundary set could be obtained, as

$$F(\mathbf{B}) = \sum_{i=1}^{N(G)} \left(\frac{1}{N_{G_i}} \sum_{\substack{m \in G_i \\ n \in G_i \\ n \neq m}} S(m, n) \right) \quad (17)$$

subject to the constrains $L_m = L_n$, $m, n \in G_i$, $1 \leq i \leq N(G)$, where G_i is the section group with the i -th label, N_{G_i} is the total number of section pairs in this group, and $N(G)$ is the number of the groups or corresponding different labels.

The constraints can also be integrated into the objective function, by considering the cost C introduced by length difference, as

$$F'(\mathbf{B}) = \sum_{i=1}^{N(G)} \left\{ \frac{1}{N_{G_i}} \left(\sum_{\substack{m \in G_i \\ n \in G_i \\ n \neq m}} S(m, n) - C |L_m - L_n| \right) \right\} \quad (18)$$

Thus the optimal boundary set can be chosen to maximize the objective function, as

$$\mathbf{B}_{opt} = \arg \max F'(\mathbf{B}) \quad (19)$$

Many optimization methods can be used to solve such problem. However, for implementation simplicity, in our approach, the length of section with the same label is imperatively set to be equal to each other, thus, the section boundary is correlated with each other and the search space is dramatically decreased. An exhaustive search is used to find the optimal boundary set.

5.3 Identifying Intro, Interlude and Coda

In the structure analysis, we still have some blank sections left to be labeled, such as the one marked as '@' in Fig. 7 (a). Such sections may be from the vocal section which only appears once, or from the instrumental section such as *intro*, *interlude* and *coda*, especially in popular music. Identifying these sections makes the structure analysis more comprehensive, especially for pop songs.

To identify the instrumental sections, the first step is to discriminate the instrumental sounds from the vocals. Following previous researches on speech ad audio processing, Mel-Frequency Cepstral Coefficient (MFCC) [13] is extracted as frame features in our approach, and delta MFCC is also used to represent the temporal variation. However, MFCC averages the spectral distribution in each sub-band, thus loses the relative spectral information. To complement this feature, octave-based spectral contrast described in [1][12] is also utilized. It can also roughly reflect the relative distribution of the harmonic and non-harmonic components in the spectrum.

These two feature sets are then concatenated into a combined feature vector for each frame. Their statistics (mean and standard variation) are used to represent the characteristics of half-second sliding window. Boosting algorithm (with native Bayes as weak classifier) is then used to classify each window into two classes.

In our approach, it is assumed that each blank section belongs to either vocal section or instrumental section. If it happens to be a mixture of above two, the dominant one is detected. Thus, the identification of each section is simply achieved by voting, based on the results of each sliding window. If the section is a vocal section, it is given a new label and integrated into the music structure. If it is an instrumental section, it can be further identified as *intro*, *interlude (bridge)* or *coda* based on its position, since *intro* and *coda* are always at the beginning and ending of the music while the *interludes* are in the middle.

6. EVALUATION AND DISCUSSION

The evaluation of the proposed algorithm has been performed on a test database composed of 100 general popular songs, performed by both male and female singers. Most of the songs are with 44.1KHz or 48KHz, stereo and 16 bits per sample.

Two subjects with music experiences are asked to annotate the ground truth of the repetitions and the music structure. In the repeating pattern annotation, they are asked to consider only the perceptually similar melodies, with a length longer than a minimum. The music structure annotation is based on the labeled repeating patterns; and the boundary of each section is usually set at the time with a local energy valley. When the subjects are confused on a song or cannot have a compromise on the annotation, the music is discarded and a substitute song is used.

In our implementation, the audio data is firstly divided into frames of 100ms long. Each frame is normalized and hamming windowed, and then feature vectors are extracted from it. In the similarity matrix calculation, the basic unit is a 1 second segment with 0.5s overlapping. It means that the resolution of the matrix is 0.5s. It is easy to improve the resolution in the cost of memory and computations. From the similarity matrix, the repetitions are detected and the structure is analyzed accordingly.

6.1 Repeating Pattern Discovery

To evaluate the extracted repetitions against the ground truth, recall, precision and F1 measure are used in our experiments. The recall and precision of each repeating pattern are calculated based on frame numbers, and then average recall and precision are used to measure the whole song. F1 measure is defined as the harmonic mean of the average recall and precision, and represents the overall performance, as,

$$F1 = 2RP / (R + P) \quad (20)$$

The first experiment compares the performance of different features, including CQT feature, chroma feature and MFCC, using the conventional Cosine distance. Since the conventional chroma feature is 12-dimension, while CQT has 36 dimensions, to explore more information and make the dimension same, in experiments, we also introduce another feature set by unpacking the 12D chroma to 36D, without integrating the components which are in the same pitch class but in different octave, just as CQT does. Correspondingly, 18D MFCC with 18D delta MFCC are used for dimension balance. Table I lists the comparison results among

CQT, chroma_36, chroma_12 and MFCC. In the experiments, we find that MFCC always finds few repetitions for most of the songs. It also indicates that remarkable improvements are obtained using CQT. Comparing with chroma_12, the recall is improved by 10.7% and precision is improved by 13.2%. CQT also has about 3% improvement from chroma_36.

Table I Performance comparisons among CQT, chroma and MFCC, using the same Cosine distance

	Recall	Precision	F1-measure
CQT	79.48%	75.14%	77.25%
Chroma_36	75.67%	73.93%	74.79%
Chroma_12	71.76%	66.35%	68.95%
MFCC	57.41%	43.61%	49.37%

In order to evaluate the proposed structure-based distance measure, we compare the performance of our distance measure with Cosine distance and Euclidean distance measure, when using the same CQT features. The detail results are shown in Table II. It can be seen that the performance of cosine distance is similar to that of Euclidean distance, while our distance measure can further improve the performance. The recall is improved 2.7%-3.5%, precision is improved 4.3-9.0% and F1 is improved 3.5-6.3%. This is because our method emphasizes more on notes and thus is more robust to the timbre disturbance.

Table II Performance comparisons among our distance, cosine distance and Euclidean distance using same CQT features

	Recall	Precision	F1
Our Method	82.92%	84.17%	83.54%
Cosine	79.48%	75.14%	77.25%
Euclidean	80.20%	79.86%	80.03%

Above evaluations are focused on pop music. Another small dataset composed of Jazz, Rock and light music is also tried in order to investigate the performance of the proposed algorithm on different music genres.

From the preliminary results, we find that our algorithm can greatly work on pop and light music. However, the performance on jazz and rock is not as good. This is because pop and light music in the test database usually have clear structure and relatively strict repetition, while most of the rocks have much percussion which disturbs the repetition detection, and sometimes rock and jazz songs even don't have distinct melody repetitions. In general, our algorithms work well for the songs with explicit structure and distinct melody repetitions.

In our experiments, we also find our method usually is not able to catch the modulated melody [7], although the modulated melody usually appears less frequency in our database. This is because our distance measure is based on the exact note, but not the melody contour.

6.2 Structure Analysis

Actually, the above evaluations on repeating patterns can roughly represent the performance of the obtained structure. In order to evaluate the structure analysis more comprehensively, the evaluations on symbol musical section and boundary bias are both investigated in the experiments.

A method similar to *edit distance* [8] is used to measure the difference between the actual structure and the obtained structure. It indicates how many detected sections are wrong, missed or inserted, compared with ground truth sections.

Table III Average Edit Distance on the obtained structure

	Error	Miss	Insert
Average Section	0.35	0.41	1.77

Table III lists the average section errors, misses and inserts of the detected structure of each song. It can be seen that only 0.35 sections are wrong and 0.41 sections are missed in each song. The most cases are inserts, where one section is usually divided into two sections. This is because our approach usually detects some subtle structures which are not labeled in ground truth. Although the obtained structure has some inserts, it actually is also an acceptable representation of actual structure, based on our informal subjective surveys.

In order to represent the detail boundary information of each musical section, another experiment is performed to show the boundary bias between the obtained section boundary and the actual boundary. The detail results are shown in the Fig. 8.

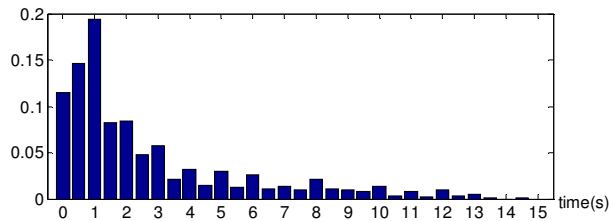


Fig. 8 Histogram of the shift between the obtained section boundary and the actual boundary

From the Fig. 8, it can be seen that the nearly 55% of the obtained boundaries are in less than 2 seconds away from the actual ones, and 75% in less than 4 seconds. It indicates our optimization-based boundary refinement algorithm performs very well. In general application, such boundary is sufficiently accurate, since there are usually some instrumental sounds between two musical sections, and it is both reasonable to classify it into either section. Moreover, it is also difficult for humans to determine the accurate section boundaries.

The final experiment is implemented to evaluate the performance of instrumental sections identification. The detailed result is listed in Table IV, comparing the performance of vocal and instrumental sounds discrimination on half-second window and music section.

Table IV. Vocal and instrumental discrimination on half-second sliding window and musical section

	On Window	On Section
Accuracy	75.6%	87.3%

Discriminating vocal from instrumental sounds is a difficult task, since the vocal sounds are usually accompanied with instrument sounds in the music. Although experiment shows that the accuracy is only about 75% in classifying each half-second window, however, it can correctly discriminate 87% of the sections. This is reasonable since sections contain more information so that the identification accuracy is improved.

7. CONCLUSIONS

This paper presents an effective approach to discover repeating patterns and musical structure from acoustic signals. Constant Q transform is used to extract notes information, and a novel distance measurement is proposed to measure the melody/note similarity more accurately. An adaptive threshold setting method is utilized to extract all the significant repeating patterns. Based on the obtained repetitions, the musical structure is further analyzed with some heuristic rules, and the optimal boundary of each music section is determined from the uncertain area with an optimization-based approach. Experiments indicate our approach is better than the conventional approaches which are based on DFT/chroma and cosine/Euclidean distance. Most of the music can get correct repetitions and structure; and most of the detected boundaries have little bias.

There are still rooms to improve the proposed approach. For example, more effective distance measure is expected in the case of the chord or concurrent multi-notes. How to suppress the effects of percussions, and how to detect the repetitions of modulated melody, are also left difficult issues in future works.

8. REFERENCES

- [1] L. Lu and H.-J. Zhang, "Automated extraction of music snippets", *Proc. of ACM Multimedia 2003*, pp.140-147, 2003
- [2] B. Logan and S. Chu. "Music Summarization Using Key Phrases" *Proc. ICASSP*, Vol. II, pp 749-752, 2000
- [3] J.-L. Hsu, C.-C. Liu and L.P. Chen. "Discovering Non-Trivial Repeating Patterns in Music Data", *IEEE Transactions on Multimedia*, Vol.3, No.3, pp.311-325, 2001
- [4] H.-H. Shih, S. S. Narayanan, and C.-C. J. Kuo, "Automatic main melody extraction from MIDI files with a modified Lempel-Ziv algorithm", *ISIMVSP*, 2001.
- [5] M. Cooper and J. Foote "Automatic Music Summarization via Similarity Analysis" *Proc.ISMIR*, pp. 81-85, 2002
- [6] M. A. Bartsch and G. H. Wakefield, "To Catch a Chorus: Using Chroma-Based Representation for Audio Thumbnailing". *Proc. Int. Workshop on applications of Signal Processing to Audio and Acoustics*, pp 15-19, 2001
- [7] M. Goto, "A chorus-section detecting method for musical audio signals", *Proc. ICASSP*, Vol. V, pp.437-440, 2003
- [8] W. Chai. "Structural Analysis of Musical Signals via Pattern Matching" *Proc. ICASSP*, Vol. V, pp 549-552, 2003
- [9] J. C. Brown, "Calculation of a constant Q spectral transform", *J. Acoust. Soc. Am*, 89(1), pp.425-434, Jan. 1990.
- [10] E. Chew, "Modeling tonality: applications to music cognition", *Proc. of 23rd CogSci*, pp.206-211, 2001
- [11] K. Castleman, *Digital image processing*, Prentice-Hall, 1979
- [12] D. N. Jiang, L. Lu, H.-J. Zhang, J. H. Tao and L. H. Cai. "Music Type Classification by Spectral Contrast Features", *Proc. ICME*, Vol. I, pp.113 -116, 2002.
- [13] L. Rabiner and B.H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, 1993.
- [14] M.Y. Wang, L. Lu, H.-J. Zhang. "Repeating Pattern Discovery from Acoustic Musical Signals", *Proc. ICME 2004*.
- [15] Glossary of Musical Terms. <http://www.classicalworks.com/html/glossary.html>