

# Repeating Pattern Discovery from Acoustic Musical Signals

Muyuan Wang<sup>1\*</sup>, Lie Lu<sup>2</sup>, Hong-Jiang Zhang<sup>2</sup>

<sup>1</sup>Department of Automation, Tsinghua University, Beijing 100084 China

<sup>2</sup>Microsoft Research Asia, Sigma Center, Haidian District, Beijing 100080 China  
wmy99@mails.tsinghua.edu.cn, llu@microsoft.com

## Abstract

Music pieces are typically repetitive. The automatic extraction of repeating patterns is useful for music summary, indexing and retrieval. In this paper, an effective approach of repeating pattern discovery is proposed. In order to represent the melody similarity more accurately, in our approach, Constant Q transform is used for feature extraction and a novel similarity measure between musical features is proposed. From the self-similarity matrix of the music, an adaptive method is used to extract all the significant repeating patterns. Experiments on pop music indicate the approach is promising.

## 1. Introduction

Music generally shows strong self-similarity, and has some repeating patterns. These repeating patterns are very helpful for further music analysis such as music summary. Some works on repeating pattern detection are for symbol music data, such as [11]. However, few literatures have fully addressed this problem for acoustic musical data. The most related papers are on music summary and music thumbnailing, where repeating patterns are detected as one step. In [1] and [2], a clustering method or HMM was utilized to group the segments with similar characteristics. Similarity matrix was employed in [3] in order to find given-length repetitions. Bartsch [4] proposed a new feature, quantized chromagram, to represent the spectral energy at each twelve pitch classes. Goto [5] also used chroma features to detect chorus sections and further developed a way to detect the modulated repetitions.

To fully investigate the repeating patterns in a music piece, in this paper, we propose an approach to extract all the significant repetitions that have similar melody, with relatively high accuracy. In our approach, acoustic features are extracted based on constant Q transform (CQT) [6], which are more suitable than MFCC and chroma-based features in this music application. A novel distance measure is also proposed to represent the similarity between music clips more properly. Finally, the repeating patterns are extracted from a similarity matrix, based on an adaptive threshold setting method.

The rest of this paper is structured as follows. Section 2 describes how a frame is represented by low-level perceptual musical features. The novel distance measurement is presented in Section 3. In Section 4, the pattern discovery process is described. In Section 5, experiments and evaluations on a corpus of pop songs are given.

\* The work was performed when the first author was a visiting student in Microsoft Research Asia

## 2. Feature Selection

In order to find repeating patterns of a music piece, we prefer to measure melody similarity which is related to a sequence of notes similarity, rather than timbre similarity. That is, we are going to discover melody repetition more than timbre repetition. However, most of the conventional features, such as MFCC, indicate more on timbre characteristics and could not represent note accurately. In order to extract features representing the music notes directly, constant Q transform (CQT) [6] is used in our approach. CQT has the ability to represent musical signals as sequences 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

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

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

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

and  $f_0$  stands for the minimal frequency that we are interested in computing. We choose it 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 get 12 semitones in an octave.  $Q$  is a constant ratio of frequency to resolution,

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

And accordingly, for the  $k$ th filter, the window width  $N_k$  is set as:

$$N_k = \lfloor f_s Q / f_k \rfloor \quad (3)$$

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 [4] 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.

In our implementation, a CQT feature vector of 36-dimension is extracted from each frame of 100ms.

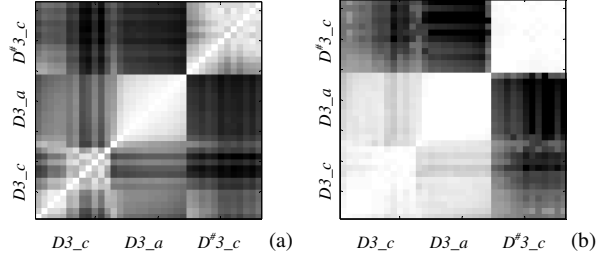
## 3. Distance Measure

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.

One note has several harmonic partials. These partials represent the timbre feature, and will be extracted as components of the CQT feature vector, since they are also at the position of some different notes. Consider two sounds with a same note but played by different instruments, they will have the same fundamental frequency but different timbre and partials. 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 can not represent accurately the actual similarity between them.

Fig.1(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 score are normalized to  $[0, 1]$ , and brighter points represent more similar musical frames. From the matrix, we can find that the similarity between  $D3$  played by cello and  $D3$  by altotrombone is not as high as expected since it considers timbre difference too much. Thus, it may introduce some noise in further repetitions discovery.



**Fig. 1** 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) Our distance measure

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

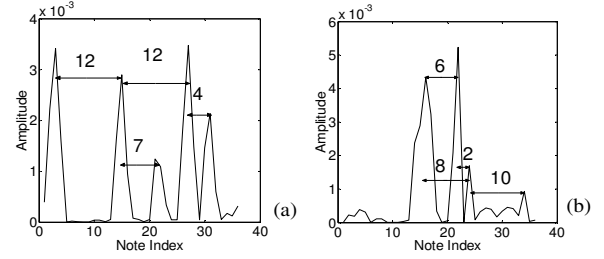
$$\Delta f = f_1 - f_2 = [f_{11} - f_{21}, \dots, f_{1N} - f_{2N}] \quad (4)$$

where  $f_1$  and  $f_2$  are the feature vectors of two notes.

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 octave, perfect fifth or major third. For example,  $2f_0$  is 12 semitones (octave) apart from  $f_0$ , and the  $3f_0$  is 7 semitones (about perfect fifth) apart from  $2f_0$ . These prominent regular intervals appearing in the difference vector of the same notes are called as *harmonic interval* in the later of this paper for simplicity. However, the difference vector between two different notes has not such characteristic. It means the difference vector between same notes has a higher harmonic structure than the one between different notes, as Fig. 2 illustrates.

In Fig. 2, the left is the difference vector between a same note  $D3$  played by cello and by altotrombone, and the right is that of different notes  $D3$  and  $D^{\#}3$  played by cello, where the horizontal axis represents the note index and the vertical axis denotes the magnitude. It is noted that the peaks are mostly octave, perfect-

fifth or major-third (4 semitones apart) spaced 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.



**Fig. 2** 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

### 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 a same note, and the distance should be relatively small; otherwise, the distance should be large.

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

$$r(m) = \sum_{n=0}^{N-m-1} \Delta f_{n+m} \Delta f_n \quad 0 \leq m \leq N-1 \quad (5)$$

where  $N$  is the dimension of the feature vector, 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 whole structure is described as a vector containing all the autocorrelation coefficients,

$$R = [r(0), r(1), \dots, r(N-1)]^T \quad (6)$$

These coefficients are all used in our distance measure. However, different coefficient should have different contribution in distance computation. For example, the coefficients with harmonic intervals, such as  $r(12)$  octave or  $r(7)$  perfect-fifth, represent the possibility that the two sounds are a same note, so they should be de-emphasized 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 frames can be estimated as,

$$d_{ij} = W^T R_{ij} \quad (7)$$

where  $W = [w(0), w(1), \dots, w(N-1)]^T$  is a weighting vector, which is chosen in the next sub-section.

Actually, the above measure only considers the isolated two segments. In order to give a more comprehensive representation of the distance, it is desirable that their neighboring temporal segments are taken into considerations. A better distance is developed as follows.

$$d'_{ij} = \frac{1}{2N} \sum_{k=-N}^{N-1} d_{i+k, j+k} \quad (8)$$

where  $2N$  neighboring frames are also considered.

### 3.2 Weighting Determination

The basic rule in choosing the weightings is that, if the interval index of the autocorrelation 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 [9] established on music perception is utilized in weighting determination. The model maps each musical note onto a helix, and 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 distances between two notes on the helix is smaller, the music interval between these two notes is more possible to be harmonic interval. Thus, weighting of  $r(m)$  can be roughly set as the distance between two notes with corresponding musical interval  $m$ . However, in the helix, the adjacent notes are 7 semitones (perfect-fifth) apart instead of 1 semitone, so we should re-order them to give an appropriate weighting, as

$$w(m) = \frac{1}{A} |P(7m \bmod 12) - P(0)| \quad (9)$$

where  $P(m)$  is the position of  $m$ -th note and set as [9] suggested,

$$P(m) = \left[ \sin \frac{m\pi}{2}, \cos \frac{m\pi}{2}, \frac{m}{2} \right]; \quad (10)$$

and  $A$  is a normalization coefficient to satisfy  $\sum w(m) = 1$ . In Eq(8), the weightings for octave interval are set as 0, in order to further de-emphasize the effect of timbre difference.

Corresponding to Fig 1(a), the similarity matrix based on new distance measure is shown in Fig. 1 (b). It can be seen that the similarity of the same notes are more distinguishable from those of different notes.

### 4. 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. The repeating patterns are represented as the highlighted lines parallel to the diagonal, as Fig 3 (a) shows. The brighter the line, the more similar two segments are; and the longer the line, the more significant the repeating pattern is.

For the convenience of processing, we mapped the matrix into a time-lag matrix  $T$  [5], as

$$T_{i,l} = S_{i,i+l} \quad (11)$$

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 3 (b) shows.

#### 4.1 Erosion and Dilation

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 3 (b). In order to further enhance the significant (relatively long) repetition lines, and remove the short lines which are caused by noises, erosion and dilation [10] 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

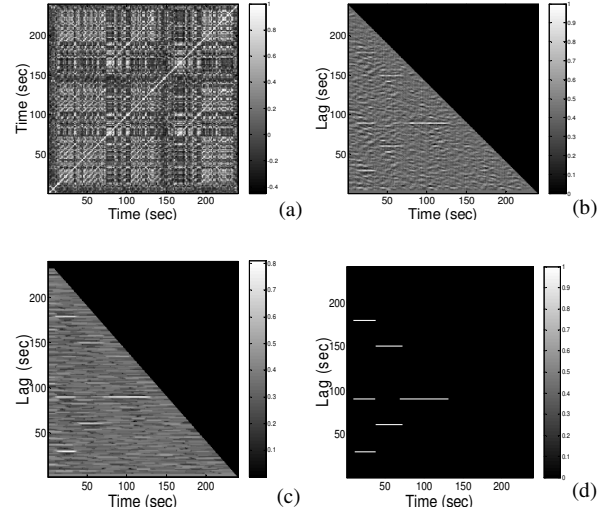
$$T'_{i,j} = \min\{T_{i,j+k} \mid k \in [-L/2, L/2]\} \quad (12)$$

where  $L$  is the minimal length of repetition we want to target.

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

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

Generally, erosion and dilation 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 3 (c) illustrates the time-lag matrix after these operations



**Fig. 3** 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 dilation; (d) Optimal final results

#### 4.2 Adaptive Threshold Setting

To this point, 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 [5], 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. 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.

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  $P^\alpha$ , and  $P^\beta$  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 to be  $[P^{0.99}, P^{0.998}]$ . An exhaustive search is used to find the optimal threshold in this candidate range.

To find an optimal threshold in this range, it is assumed that a good threshold should be robust to small disturbance. That is, the result is assumed have tiny variance when threshold changes

in the neighborhood of the optimal one. Based on it, the optimal threshold  $Th_{opt}$  is chosen as,

$$Th_{opt} = \arg \min_{Th \in [P^\alpha, P^\beta]} \frac{\Delta R(Th)}{\Delta Th} \quad (14)$$

where  $R(Th)$  is the repetition results detected with the threshold  $Th$ . In real implementation, we choose the optimal threshold corresponding to the minimal variance of the detected repetition length.

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 3 (d), from which the repetitions can be easily detected.

## 5. Experiment

In this section, our algorithm is evaluated on a music corpus of 50 pop songs, including 20 English songs and 30 Chinese songs, recorded or alive, performed by male or female singers. All the songs have the sampling frequency of 44100Hz.

The ground truth of the repetition segments is manually labeled. In the ground truth annotation process, we only consider the perceptually identical melodies, with a length longer than a lower limit, which is set as 10 seconds in our experiments. Comparing the ground truth and extracted repetitions, recall, precision and F1 measure are used to evaluate the performance of our repetitions discovery algorithm. F1 measure is defined as the harmonic mean of the recall and precision, as,

$$F1 = 2RP / (R + P) \quad (15)$$

The first experiment compares the performance of different features, including CQT feature, chroma feature and MFCC, using the same cosine distance measure. Table 1 lists the comparison results between CQT and chroma. In the experiments, we find that MFCC hardly finds any repetitions for most of the songs. So we do not list the MFCC results in the table. It can be noted that remarkable improvements are obtained using CQT. The recall is improved 11.6% and precision is improved 7.7%.

**Table 1** Performance comparisons between CQT and chroma using same cosine distance measure

	Recall	Precision	F1-measure
CQT	86.86%	83.55%	84.70%
Chroma	75.22%	76.87%	73.71%

In order to evaluate the proposed distance measure, another experiment is performed. 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 2. It can be seen that the performances of cosine is slightly better than Euclidean distance, while our distance measure can further improve the performance, comparing to them. The recall is improved 3.5%-6.5%, precision is improved about 4.6-5.7% and F1 is improved 4%-5.9%. This is because our method emphasizes more on notes and thus is more robust to the timbre disturbance and accompanying instruments.

To evaluate the language dependence of our approach, experiments are also performed on Chinese and English songs, respectively. The similar performances on both Chinese and

English songs indicate that our approach is independent of the language of the songs, as shown in Table 3.

**Table 2** Performance comparisons among our distance, cosine distance and Euclidean distance using same CQT features

	Recall	Precision	F1
Our Method	90.31%	88.11%	88.72%
Cosine	86.86%	83.55%	84.70%
Euclidean	83.81%	82.37%	82.87%

**Table 3** Performances on Chinese songs and English songs in our approach

	Recall	Precision	F1
English	88.92%	88.60%	88.20%
Chinese	91.06%	87.84%	89.00%

## 6. Conclusion

This paper presents an effective approach to discover repeating patterns from musical 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 also proposed to extract all the significant repeating patterns. Experiments indicate our approach is much better than the conventional approaches which are based on DFT/chroma and cosine/Euclidean distance. Further works are to determine boundaries of the repeating patterns and further discover the music structure. Moreover, the approach need to be further improved to explore modulated repetitions.

## 7. References

- [1] L. Lu, H.-J. Zhang, "Automated extraction of music snippets", *Proc. of ACM Multimedia 2003*, pp.140-147, 2003
- [2] B. Logan, S. Chu, "Music summarization using key phrases", *Proc. of ICASSP 2000*, Vol. II, pp.749-752, 2000
- [3] M. Cooper, J. Foote, "Automatic music summarization via similarity analysis", *Proc. of ISMIR 2002*, pp.81-85, 2002
- [4] M. A. Bartsch and G. H. Wakefield, "To Catch a Chorus: Using Chroma-Based Representation for Audio Thumbnailing". *Proc. WASPAA*, pp 15-19, 2001
- [5] M. Goto, "A chorus-section detecting method for musical audio signals", *Proc. of ICASSP 2003*, Vol. V, pp.437-440, 2003
- [6] J. C. Brown, "Calculation of a constant Q spectral transform", *J. Acoust. Soc. Am.*, Jan. 1990, pp.425-434.
- [7] R. N. Shepard, "Circularity in judgments of relative pitch", *J. Acoust. Soc. Am.*, vol. 36, no. 12, pp. 2346-2353, 1964
- [8] D. William, E. Brown, *Theoretical foundations of music*, Wadsworth Pub. Co., 1978
- [9] E. Chew, "Modeling tonality: applications to music cognition", *Proc. of 23<sup>rd</sup> CogSci*, pp.206-211, 2001
- [10] K. R. Castleman, *Digital image processing*, Prentice-Hall, 1979
- [11] J. L. Hsu, C. C. Liu, and L. P. Chen, "Discovering Non-Trivial Repeating Patterns in Music Data," *IEEE Trans. on Multimedia*, vol. 3, No. 3, pp. 311-325, 2001