行政院國家科學委員會專題研究計畫 成果報告

以內容為基礎之音樂擷取

計畫類別：個別型計畫
計畫編號：NSC92-2213-E-032-021-
執行期間：92年08月01日至93年07月31日
執行單位：淡江大學資訊工程研究所

計畫主持人：林慧珍
共同主持人：顏淑惠

報告類型：精簡報告

處理方式：本計畫可公開查詢

中 華 民 國 93年11月2日

Main Melody Extraction for Polyphonic Music Based on Approximate Matching

Hwei-Jen Lin and Hung-Hsuan Wu
Department of Computer Sciences and Information Engineering
Tamkang University, Tamsui, Taipei, Taiwan, R.O.C.
E-mail：hjlin@cs.tku.edu.tw

Abstract

We propose a method of main melody extraction from multi-track polyphonic MIDI files. In each track of a MIDI file, the system traces the pitch contour of the polyphonic music, and describes it in a monophonic form. In each of these contours, maximal approximate repeating patterns are found with the help of a modified correlative matrix, so that exact and approximate repeating patterns. Not only the exact repeating patterns but also the approximate repeating patterns can be extracted. All of these extracted patterns are then collected in a dictionary, with which we can find the set of all maximal repeating patterns with no redundancy, called the main melody. The main melody extraction results and how well these extracted main melodies can improve the work on content-based music retrieval are given and described.

Keywords：content-based music retrieval, repeating pattern, main melody extraction, correlative matrix, polyphonic music.

1. Introduction

The main melody is the set of certain patterns or phrases that the composer stresses and is always repeating throughout the entire piece of music [1][2]. The main melody or theme varies in nature, depending on the type of music. In the Baroque period, contrapuntal composition was very popular, a single idea, or theme, continued throughout the piece with scarcely a moment's letup [7]. Thus for main melody extraction, we are interested in finding the frequent occurring patterns in the music object.

Most of the published papers dealing with main melody extraction work on monophonic music [1][2][3]. In the recent years, there have been many papers presenting various methods of extracting main melodies in polyphonic music [4][5] [6].

Medina et al. [5] flatted a polyphonic MIDI file into a text sequence and used a self-similarity matrix to find the repeating patterns in the given text sequence. In consequence, redundancy and proper substrings of other patterns can be removed. Then they also remove the closely related patterns with significant length of partial matched patterns. The shortcoming of this method is that repeating themes with slight change cannot be detected. To overcome this problem, we adopt a modified correlative matrix to extract approximately repeating patterns.

Ren et al. [6] described how to find the retrograde themes and inverted themes based on an approximate matrix method. Such retrograde themes and inverted themes frequently occur in contrapuntal compositions in the Baroque period, but seldom appear in those produced in the later period.

In this paper we propose a method for finding the maximal approximate repeating patterns in not only monophonic but also polyphonic music stored in multi-track MIDI files. The MIDI file is first pre-processed to obtain the contour of each track. A modified correlative matrix is then utilized to determine all maximal approximate repeating patterns in each track.

A dictionary is then used to collect the patterns extracted from each track. Finally, some patterns are removed from the dictionary, such as redundant patterns, patterns that are proper sub-patterns of other patterns, and patterns that are part of the accompaniment. This paper is organized as follows. In

Section 2, the proposed main melody extraction procedure is described. The experimental results are shown in Section 3. Section 4 concludes the paper and discusses some future work.

2. Main Melody Extraction

In this section, we propose a procedure of main melody extraction from polyphonic music. In each MIDI file track, the system traces out the pitch contour of the polyphonic music, which is described in a monophonic form. We construct a correlative matrix to find out all maximal repeating patterns in each track and use a dictionary to collect all repeating patterns with the numbers of their occurrences [5]. Finally, inappropriate candidates from the collection are discarded according to some general properties of the main melody. The set of remaining patterns in the dictionary is the so-called main melody.

2.1. Pitch Contour Tracing

The outer voice is usually perceptually significant, while the inner voice is hard to recognize [8]. In this research, the highest pitch contour will be traced for main melody extraction; that is, if there are more than one notes playing simultaneously, the highest pitch is extracted. This process is called the pitch contour tracing. Thus, the pitch contour of a piece of polyphonic music is the sequence of the highest pitches along the music sequence.

In this sub-section, we propose Algorithm PCTRACE to trace out the pitch contour for a piece of polyphonic music. For a given sequence of MIDI events along a time axis, PCTRACE utilizes a priority queue, denoted by PQ, to extract the highest pitch at every time point shown in the time axis. The sequence of these extracted highest pitches along the time axis is called the contour of the polyphonic music and denoted by CS in the Algorithm PCTRACE. The pitch contours of all tracks are extracted and then stored in a monophonic form.

Algorithm PCTRACE

The contour sequence *CS* is constructed as follows:

0. Initially, for time point 0, add a dummy pitch with value 0 into the empty sequences.
1. Along the time axis, all events at a time point *t* are read. For each note-off event, the system removes its corresponding note-on event from *PQ*. If this removed pitch is the largest value in *PQ*, the second largest pitch is extracted (not remove) and is added into *CS* for time point *t*.
2. Each note-on event is inserted into *PQ*.
3. The highest pitch, denoted by *hp*, is extracted from *PQ*, if *hp* is higher than the last pitch, denoted by *lp*, in *CS* then pitch *hp* is added into *CS* for time point *t*.
4. Proceed on the next time point along the time axis and go to step 1.

2.2. Maximal Approximate Repeating Pattern Extraction

Consider the example of the opening of the Brahms Waltz in A flat, shown in Figure 1. It can be represented by the pitch string C6–Ab5–Ab5–C6–C6–Ab5–Ab5–C6–Db6–C6–Bb5–C6. We can easily see that the pattern C6–Ab5–Ab5–C6 occurs twice in the pitch string and is not a proper sub-pattern of any other repeating pattern. Such a pattern is a so-called maximal repeating pattern.

Figure 1. The score of the opening of the Brahms Waltz in A flat.

To automatically determine the maximal repeating patterns in a pitch string *S*, one may construct a correlative matrix *T* of size $n \times n$, where *n* is the length of the pitch string *S*. Let S_i denote the *i*-th character of S and $S_{i...j}$ denote the sub-string of S from the i-th to j-th characters. Initially, the value of each entry of T is set to zeros and then the correlative matrix is constructed row by row, from left to right as follows. If $S_i = S_j$, the value for $T_{i,j}$ is set to that of $T_{i-1,j-1}+1$; otherwise, set to zero. Because the matrix T is symmetric, we only need to work on the portion above the main diagonal of the matrix. The value of each entry $T_{i,j}$ denotes the length of the repeating pattern $S_{h...j}$, called the pattern corresponding to $T_{i,j}$, and thus, $j-h+1= T_{i,j}$ or $h=j+1-T_{i,j}$. The construction of the correlative matrix is performed by Algorithm CoMatrix given as follows.

Algorithm CoMatrix

//initialization
for $i \leftarrow 1$ to n do
    for $j \leftarrow i+1$ to n do
        $T_{i,j} \leftarrow 0$
//construct the entries of the correlative matrix
for $i \leftarrow 1$ to n do
    for $j \leftarrow i+1$ to n do
        if $(S_i = S_j)$ then $T_{i,j} \leftarrow T_{i-1,j-1}+1$

The correlative matrix for the Brahms Waltz opening shown in Figure 1 constructed by Algorithm CoMatrix is shown in Figure 2.

	C6	A^b5	A^b5	C6	C6	A^b5	A^b5	C6	D^b6	C6	B^b5	C6
C6	—			1	1			1		1		1
A^b5		—	1			2	1					
A^b5			—			1	3					
C6				—	1			4		1		1
C6					—			1		1		1
A^b5						—	1					
A^b5							—					
C6								—		1		1
D^b6									—			
C6										—		1
B^b5											—	
C6												—

Figure 2. The correlative matrix of the opening pitch string of the Brahms Waltz in A flat.

We find that some short patterns tend to occur frequently. We simply discard these short repeating patterns by checking their lengths. That is, we discard a repeating pattern if its length is less than a given threshold value t.

Because the main melody or theme varies in nature, a correlative matrix coupled with mismatch penalty forms a modified version that we shall use in this research. With the help of such a modified correlative matrix we may extract all maximal approximate repeating patterns.

The modified correlative matrix T for a pitch string S of length n, is constructed as follows. Let S_i denote the i-th character of S and $S_{i...j}$ denote the sub-string of S from the i-th to j-th characters. Initially, the value of each entry of T is set to zero and then the modified correlative matrix is constructed row by row, from left to right as follows. If $S_i = S_j$, the value for $T_{i,j}$ is set to $T_{i-1,j-1}+1$; otherwise, set to $T_{i-1,j-1}-c$, where c is a penalty value. Because the matrix T is symmetric, we only need to work on the portion above the main diagonal of the matrix.

To retrieve the approximate repeating patterns, we trace along each line parallel to the main diagonal of the matrix to find the maximal value $T_{i,j}$ between two consecutive zeros. Suppose that these two consecutive zeros are $T_{a,b}$ and $T_{c,d}$, where $a<c$ and $b<d$, then we extract $S_{a+1\ldots i}$ as an maximal approximate repeating patterns.

The construction of the modified correlative matrix is performed by Algorithm Modi-CoMatrix given as follows.

Algorithm Modi-CoMatrix

//initialization
for $i \leftarrow 1$ to n do
    for $j \leftarrow i+1$ to n do
        $T_{i,j} \leftarrow 0$
//construct the entries of the correlative matrix
for $i \leftarrow 1$ to n do
    for $j \leftarrow i+1$ to n do
        if $(S_i = S_j)$ then $T_{i,j} \leftarrow T_{i-1,j-1}+1$
                                  else $T_{i,j} \leftarrow T_{i-1,j-1}-c$
//retrieve the approximate repeating pattern
for $dist \leftarrow 1$ to $n-1$ do
    for $i \leftarrow 1$ to $n-dist$ do
        $j = i + dist;$
        if ($T_{i,j}$ *is maximal of* ($T_{a,b}$, $T_{c,d}$))
            then *the maximal approximate repeating pattern is* $S_{a+1\ldots i}$

The modified correlative matrix for a given string shown in Figure 3, is constructed by Algorithm Modi-CoMatrix is shown in Figure 4. We also set a threshold value t to discard short repeating patterns.

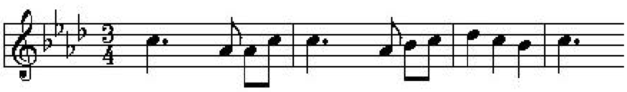

Figure 3. The score of C6–Ab5–Ab5–C6–C6–Ab5–Bb5–C6–Db6–C6–Bb5–C6.

	C6	A^b5	A^b5	C6	C6	A^b5	B^b5	C6	D^b6	C6	B^b5	C6
				$T_{a,b}=0$								
C6	–			1	1			1				1
A^b5		–	1		0.001	2			0.001			
A^b5			–	0.001		1.001	1.001					
C6				–			0.002	2.001		1		1
C6					–			1	1.001	1	0.001	1
A^b5						–	1		0.001	0.001	0.001	
B^b5							–	0.001			$T_{c,d}=0$	
C6								–		1		
D^b6									–		0.001	
C6										–		1
B^b5											–	
C6												–

Figure 4. The modified correlative matrix of the given string in figure 3.

2.3. Removing Improper Candidates

Most melody lines contain a preponderance of conjunct motion, but the inclusion of few leaps or disjunction motion will help greatly in adding interest and variety to the melody line [6]. When extraction is performed on multi-track polyphonic music, the candidates might be extracted from the accompaniment part. The accompaniment might consist of many disjunctions, frequently repeating tones, or long scales, which are rarely included in the main melody. Thus, we ought to discard these candidates. According to the fact that the main melody consists of much more conjunction intervals than disjunction intervals, we may detect disjunctions in a candidate by checking its disjunction ratio R_{disj}, which is the ratio of the number of the disjunctions to the total number of intervals; that is, R_{disj} = *No. of disjunction intervals / No. of all intervals*. If the value of for a candidate is out of a given range, it has too many disjunctions or is lack of disjunctions, and thus, it possibly comes from the accompaniment and should be removed. Frequently repeating tones much more likely come from the accompaniment than the main melody. To detect frequently repeating tones occurring in a candidate, we check the zero interval ratio R_{zero}, which is the ratio of the number of the zero intervals to the total number of intervals; that is, R_{zero} = *No. of zero intervals / No. of all intervals*. If the valve of is greater than a given threshold value, we consider the candidate as part of the accompaniment and remove it. In this research we assume a disjunction interval is of more than 6 half tones (the perfect 4th).

3. Experimental Results

No matter whether a MIDI file contains polyphonic or monophonic music, and the music is in multiple tracks or in a single track, Algorithm PCTRACE can trace out correct pitch contours. Figure 5(a) shows the opening 6 measures of the *Bach's 2-Invention No.13 in a minor* in a single-track MIDI file and its result of the pitch contour tracing is shown in Figure 5(b).

(a) (b)

Figure 5.(a) The opening 6 measures of the Bach's 2-Invention No.13 and (b) The result of PCTRACE.

The time complexity of contour tracing PCTRACE module is *O(n)*, where *n* is the number of notes in a MIDI file, which is approximately proportional to the size of the MIDI file. Figure 6 shows the time cost of the main melody extraction versus the number of notes. The work of main melody extraction includes contour tracing, constructing the modified correlative matrix, extracting all maximal approximate repeating patterns and inserting them into in a dictionary, removing the proper sub-patterns, and discarding some improper candidates.

Among the extracted maximal approximate repeating patterns, there might exist some overlapping patterns. In order to eliminate these overlapping patterns, we can use a binary index array to store the position corresponding to the position of all extracted patterns. If one pattern occurs in some position of a given string *S*, we could set the position of the index array to be true. All these non-repeating position will be false. In such way, all patterns are mapping to a binary index string, we can trace the index array to merge all overlapping patterns and extract all non-overlapping patterns as our candidates.

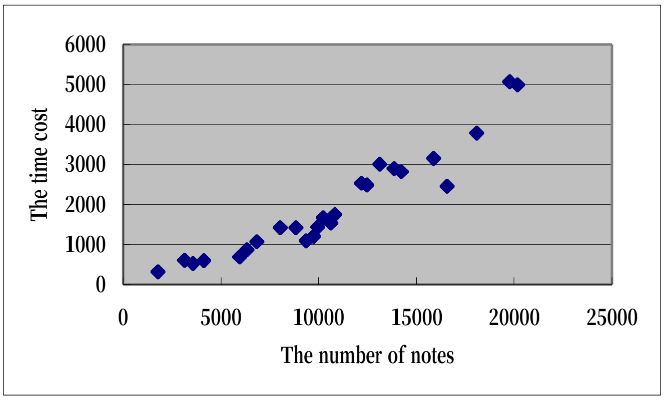

Figure 6. The time cost of the main melody extraction versus the number of notes in pitch contour.

In our experiment on removing the candidates coming from the accompaniment, the threshold value R_{zero} is set to 0.8. If the value of R_{zero} for a candidate is greater than 0.8, we think this candidate includes too many repeating notes, and so we discard it. The given range for R_{disj} is from 0.05 to 0.5. If the value of R_{disj} for a candidate is out of the range, we discard it.

To evaluate the performance of our main melody extracting method, we compare the extracted result with the Barlow and Morgenstern's Dictionary of Music Themes [9]. The data set tested in our experiment includes 25 orchestral or sonata movements from Baroque, Classical and Romance period. The Composers involved in the test data set are Bach, Haydn, Mozart, Beethoven, Brahms, Schubert. The entire data set includes approximately 265,000 notes. On average, each composition contains 10627 notes.

The rate that the theme extraction algorithm successfully captures the primary theme is 96%. Many of the orchestra works in the test data set have secondary, third, or even fourth themes listed in the theme dictionary. There are 22 works containing second theme, and we found 16 of them; that is about 72%. Additionally, we also found 5 of the 9 third themes and 2 of 3 fourth themes.

The extracted main melody data size is about 32.3% of the origin data set by using our method, and about 16.9% by using our previous exact matching method. In our experiments of extraction base on the exact matching, there are two music works that we cannot capture the first theme because their themes are not exactly matched. Thus the approximate approach is necessary although the size of be extracted main melody is about twice of that extracted by the exactly matching method.

4. Conclusions and Future Work

In this paper, we propose an effective method for main melody extraction in multi-track polyphonic music. Approximate matching approach and approximate modified correlative matrices are utilized to detect all maximal approximate repeating patterns, and then a dictionary-based approach is applied to remove redundancy. According to some melody line properties inappropriate candidates are identified and removed.

We have defined main melody using the approximate repeating property. However, there are several factors that can be taken into account such as the duration of notes and the phrases of the music.

If we can design a phrase tracking mechanism, we may treat a phrase as a pattern and use a dictionary to accumulate the number of occurrences of each phrase without using the correlative matrix. Unfortunately, automatic segmentation of musical sentences or phrases is not easy. Some research found that the rhythm information can be used to segment a music work into computational segments, the segmentation result are quite different from those obtained by human.

References：

[1] Yuen-Hsien Tseng, "*Content-Based Retrieval for Music Collections*". In Proceedings of the ACM International Conference on Research and Development in Information Retrieval(SIGIR), Berkeley, CA, 1999. 176-182.

[2] Jia-Lien Hsu, Arbee L. P. Chen, C. C. Liu, "*Efficient Repeating Pattern Finding in Music Database*", Proceedings of the seventh international conference on Information and knowledge management, Bethesda, Maryland, United States, November 02-07, 1998, 281-288.

[3] C. C. Liu, J. Hsu, and Arbee L. P. Chen, "*Efficient Theme and Non-Trivial Repeating Pattern Discovering in Music Database*". In Proceedings IEEE International Conference on Data Engineering, 1999, 14-21.

[4] Colin Meek and William P. Birmingham, "*Thematic Extractor*", International Symposium on Music Information Retrieval, 2001, 119-128.

[5] Richard A. Medina, Lloyd A. Smith and Deborah R. Wagner, *"Content-based Indexing of Musical Scores"*, In Proceedings Joint Conference on Digital Libraries, 2003, 18-26.

[6] Xiaona Ren, Lloyd A. Smith, and Richard A. Medina, *"Discovery of Retrograde and Inverted Themes for Indexing Musical Scores"*, In Proceedings Joint Conference on Digital Libraries, 2004, 252-253.

[7] Edward Aldwell and Carl Schachter, "*Harmony and Voice Leading",* (second edition), Harcourt Brace Jovanovich College Publishers, 1989.

[8] Joseph Kerman, "*Listen*", Worth Publishers, 1976.

[9] Barlow H. and Morgenstern S. "*A Dictionary of Musical Themes"*. Crown Publishers, NY, 1948.