Using Relative Interval Slope in Music Information Retrieval *

Kjell Lemström◇, Pauli Laine‡, Sami Perttu◇
◇ Department of Computer Science, P. O. Box 26
‡ Department of Musicology, P. O. Box 35
FIN-00014 University of Helsinki, Finland
Kjell.Lemstrom@cs.Helsinki.Fi, {Pauli.Laine,Sami.Perttu}@Helsinki.Fi

Abstract

The two descriptors most often used in music information retrieval (MIR) are pitch interval and note duration. We introduce a method combining the two in a single value called the interval slope (IS). It expresses the ratio of the sizes of pitch intervals to note durations. However, when representing music with an IS sequence, one only obtains invariance under transpositions. In order to also obtain invariance under different tempi, the proportions of consecutive IS values have to be considered. The resulting representation is called the relative interval slope (RIS) sequence. We experiment with the RIS method in exact matching and discuss how it could be used in approximate matching.

1 Introduction

Until the last couple of years practically all the research effort in multimedia information retrieval and pattern matching has been directed to applications of large image and text collections, although the music (and audio) is an equally important constituent of the multimedia domain. In the last few years the activity in this domain has noticeable increased, both in terms of implementations (such as McNab et al.'s *MELDEX* [1], Beeferman et al.'s *QPD* [2] and Prechelt and Typke's *TuneServer* [3]) and scientific articles. Furthermore, the increase of this activity will continue since music information retrieval issues are related both to the forthcoming MPEG-7 standard[1] and to the 5th Framework of European Community research[2].

Partitioning music into separate elements such as rhythm, pitch and harmony, makes it easier to observe each of them separately. This approach has been adopted in music theoretical literature. The approach has also been adopted in major part of MIR literature; very often the matching of a *query pattern* against the database melodies is based either on note durations or (absolute or relative) pitch information.

However, it should be advantageous to integrate some of these parameters in order to get a better understanding of the perceptual features of musical lines. Meyer [9, pp.18–19] considers pitch and time to be the most important descriptors of music. When pitch and rhythm descriptors are integrated we can speak of the melodic contour of music, which has been considered to be an important factor in music perception [5, p.169].

We introduce a new approach for MIR. In this approach the two descriptors are combined in a single value which we call interval slope (slope meaning the interval angles between two consecutive notes), or IS for short. The value describes how quickly and how much the pitches are changing. Furthermore, in order to make the system invariant under different tempi we introduce the concept of *relative interval slope* (RIS). RIS expresses changes between adjacent interval slopes, thus resulting in a "gesture signature" of the melody. This approach may be also used in music analysis: it might be advantageous in finding "gestural strategies" among certain style, homogenous set of compositions or in some individual composition. Furthermore, when needed, the local maximas and minimas (i.e. the highest and lowest notes) of a music score can be found by using the RIS representation.

Ghias et. al. [6] presented an idea for enabling inexactness to the query pattern. Since then this idea has been widely used and it is as follows. The alphabet of possible intervals is reduced to a set of only three distinct characters $\{+, -, =\}$ representing intervals upwards, downwards and same as

[1] http://www.cselt.it/mpeg/
[2] http://www.cordis.lu/fp5/home.html
* A work supported by the Academy of Finland under grant 44449

previous, respectively. However, in real applications with large databases more precise information is needed in order to avoid huge amount of irrelevant, spurious occurrences of the search pattern (an experiment of the undesirable behaviour of this approach can be found e.g. in [12]).

To make the representation more precise and avoid the multitude of "false positives" a combination of it and rhythmic information has been proposed (see e.g. [8, 11]). We claim that if both pitch and rhythm are to be considered in the matching, it is more natural to give the search pattern as a combined slope than as two separate values. Enabling approximate matching with the RIS method still requires some rethinking, but once it has been done the task of finding variations of a motive should become straightforward.

Since the two values are combined into one, a single pass through the database is required (although there might be a need for another pass due to the ambiguity of the signature). Furthermore, RIS enables more ways to give the query pattern. For instance, an MIR system with an approximate matching feature may provide the non-musical user with a canvas on which the query pattern may be painted as a polyline drawing. The user only has to define the vertices of the melodic contour; the horizontal distance between two consecutive vertices represents the duration of a note and the vertical distance the size and direction of the interval.

Problem Setting. We suppose that we are dealing with (semitonic) MIDI sources containing monophonic information in each track. The ideal input to us would be a notated score that is converted into MIDI format. Thus, any normalization related to the rhythmic features of the music, such as quantization or time shifting, is not needed (the quantization problem of music is considered e.g. in [4, 10]).

Henceforth, we will use two distinct notations for real numbers: x/y means that the corresponding decimal number is computed only by request and the fractional number is stored as a pair (x, y). $\frac{x}{y}$, on the other hand, is stored as a single decimal number.

2 Relative Interval Slope

Let us start by formally defining what we mean by an IS sequence. We denote the monophonic music sequence by $S = s_1 \dots s_n$, where each element s_i consists of a pair $\langle x_i, y_i \rangle$. The first component of the pair, x_i, is the chromatic pitch of the note s_i, where the middle C is given a value of 60 (as per the MIDI standard).

The second component y_i is the duration of the note s_i. The values of y_i are tied by the following assignment: `duration`(quarter note) = 1. Thus, the duration of an eighth note equals $\frac{1}{2}$, half note = 2, etc.

Now we are ready to define the sequence IS:

$$IS = \alpha_1 \dots \alpha_n;\ \alpha_i = \begin{cases} x_1/y_n, & \text{if } i = 1, \\ (x_i - x_{i-1})/y_{i-1}, & \text{else.} \end{cases}$$

Clearly, the elements α_i are well defined since their denominators are never equal to 0 (currently we are ignoring pauses, but negative durations may be used for them). Moreover, consider the first element α_1 as a "don't care" value that just stores the first value of the MIDI pitch value sequence and the last value of the duration sequence.

However, there is a problem with this representation; namely it is not invariant under different tempi. Clearly, the same melody played first with eighth notes and then with half notes results in two completely different IS sequences. Thus, we introduce the concept of an RIS sequence:

$$RIS = \beta_1 \dots \beta_n;\ \beta_i = \begin{cases} \alpha_i, & \text{if } i = 1, 2, \\ \frac{\alpha_i}{\alpha_{i-1}}, & \text{else.} \end{cases}$$

We use a slightly modified version of S to represent the monophonic music sequence. Let $S' = s'_1 \dots s'_n$ be a sequence of triples: $s'_i = \langle \gamma_i, \delta_i, \beta_i \rangle$, where γ_i and δ_i are the nominator and denominator of α_i, respectively. Please note that only the component β_i is missing from the representation that we have defined earlier in [7]. Thus, it has to be calculated on-line.

2.1 Properties of RIS

Actually, an IS sequence is a sequence of local first order derivatives. The elements of RIS sequences share a property of second order derivatives: The value of β_i is positive when the sign of α_{i-1} is preserved in α_i. If the subsequence $\alpha_{i-1}\alpha_i$ contains two different signs the value of β_i is negative.

The drawback of RIS is that it shortens the query pattern. If m is the length of the query pattern, $m-1$ is the true length of the corresponding IS sequence and $m-2$ is the true length of the corresponding RIS sequence.

However, the RIS method obtains desirable invariances, as stated in the following theorem.

Theorem 1 *RIS representation is invariant both under transposition and different tempi.*

Proof Let us consider an arbitrary subsequence $s_i s_{i-1}$ of consecutive notes of a musical excerpt. Now the corresponding elements in the IS representation are: $\alpha_i = (x_i - x_{i-1})/y_{i-1}$, $\alpha_{i-1} =$

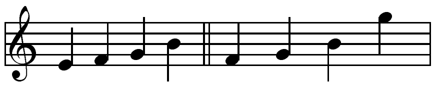

Figure 1: *The proportions of the consecutive intervals of the two excerpts are equal (the corresponding interval sequences are 1,2,4 and 2,4,8).*

$(x_{i-1} - x_{i-2})/y_{i-2}$. The corresponding RIS value can be expressed as follows:

$$\beta_i = \frac{x_i - x_{i-1}}{x_{i-1} - x_{i-2}} \times \frac{y_{i-2}}{y_{i-1}};\ i = 3, \ldots, n. \qquad (1)$$

The second factor of the multiplication sentence (1) expresses the proportion of the consecutive durations. Since the value of that term is the same whenever the proportion of consecutive durations is the same, the invariance under different tempi is obtained. The transposition invariance can be verified similarly with respect to the first factor of (1). □

Actually, the first factor of (1) contains a stronger invariance than just transposition invariance. All subsequences $< x_{i-2}, x_{i-1}, x_i >$ whose proportion $\frac{x_i - x_{i-1}}{x_{i-1} - x_{i-2}}$ are the same, fall in a same equivalence class. Thus, for instance, the two excerpts of Fig. 1 are equivalent.

Moreover, it is important to keep track of the sign of the first “real” *IS* sequence element, since otherwise two sequences containing slopes of equal size but opposite signs cannot be discriminated. For instance, consider Fig. 2 where all intervals are exactly equal with respect to *RIS*. Thus, all the subsequences of the same length are equal (with respect to *RIS*) unless the sign of the first *IS* value is considered. Also note that with respect to *IS* two equivalence classes are established: one for downward and one for upward intervals.

2.2 Avoiding Discontinuity

As was already mentioned, the values of an *IS* sequence are always well defined. However, with *RIS* there might be discontinuity problems: a prime interval occurring at some location i ($x_i - x_{i-1} = 0$) implies that the denominator of one element of the (RIS) sequence is 0 ($\beta_{i+1} = \frac{\alpha_i}{0}$). We overcome the discontinuity problem of *RIS* by preprocessing the music. In the preprocessing phase all consecutive notes with prime intervals are aggregated to a single note. Its duration is set equal to the sum of the durations of the notes that are aggregated.

48/0.5 12/4 -12/4 6/2 -6/2 3/1 -3/1 (IS)
48/0.5 12/4 -1 -1 -1 -1 -1 (RIS)

Figure 2: *RIS equivalence within a music sequence containing both downwards and upwards intervals. The corresponding IS and RIS sequences are given.*

$\|p\|$	avg hits	avg false positives
3	10 757.4	27 858.27
4	1 921.3	1 701.62
5	324.8	193.48
6	68.4	34.16
7	21.3	6.76
8	10.5	1.79
9	4.7	0.22
10	3.8	0.07
11	3.5	0.05
12	2.8	0.01
13	2.7	0
14	2.5	0
15	2.6	0
16	2.4	0
17	2.3	0
18	2.2	0
19	2.1	0
20	2.1	0

Table 1: The average number of true and false positive hits as a function of query pattern length ($|p|$).

3 Experiments

Since each equivalence class of *RIS* sequences contain more than just one particular sequence under different transpositions and tempi, it is important to ascertain the practical performance of the method. Here we consider exact matching only.

In order to get an idea of the magnitude of false positive hits versus true matches, we conducted some experiments on a database of simple melodies consisting of 6070 monophonic MIDI tracks. A true match is any matching *RIS* sequence whose intervals are identical to the intervals in the search pattern.

For each query pattern length between 3 (corresponding to one RIS value) and 20, one thousand test patterns were randomly selected from the database. An exhaustive search was then performed to find all matches. The whole run of 18.000 searches took approximately half an hour on a 350 MHz Pentium II. The averaged results are shown in Table 1. Please note that since the test patterns were picked from the database itself,

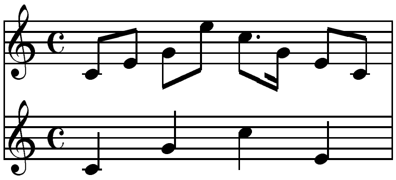

Figure 3: *Two melodic excerpts that are musically closely related.*

at least one true match was certain for each pattern.

The percentage of false positives from all matches decreases rapidly as the length of the query pattern increases. Still 33% at pattern length 6, it drops to 4.5% at pattern length 9. From pattern length 13 onwards, only true matches occur.

4 Discussion

We have presented a new approach to MIR applications. In this approach interval and duration information of a monophonic music sequence are combined into a single value. The resulting representation, which we call relative interval slope sequence, is invariant both under transposition and different tempi.

There is still much to do. For instance, consider the example given in Fig. 3 where the two excerpts are musically very similar. The traditional way to computationally measure the similarity between the two excerpts is to use the edit distance concept (see e.g. [13]). However, in this case at least 4 deletion (or insertion) operations would be required, and still the remaining RIS values would not be the same (unless they are calculated after the deletions, and the remaining durations are normalised).

We propose two different heuristic to be considered. The first one is somewhat similar to the *time warping* method. If consecutive RIS values are almost the same, they can be expressed with a single value. Thus, even rather long subsequences may be aggregated and compared as a single value to another RIS value. However, the case of the Fig. 3 would not be captured by this method since the big interval (x_3) and the dotted rhythm (y_5) of the upper excerpt have a strong impact on the corresponding RIS values. Another heuristic to be considered is straightforward metric position analysis. Only the notes in respective metric positions are considered, and the RIS values between the selected notes are calculated.

Acknowledgements

The authors are indebted to professor Esko Ukkonen for valuable remarks concerning on this research.

References

[1] http://www.nzdl.org/cgi-bin/gw?c=meldex&a=page&p=coltitle.

[2] http://www.link.cs.cmu.edu/qpd/.

[3] http://wwwipd.ira.uka.de/tuneserver/.

[4] A. T. Cemgil, P. Desain, and B. Kappen. Rhythm quantization for transcription. In *Proceedings of the AISB'99 Symposium on Musical Creativity*, pages 64–69, 1999.

[5] J. Edworthy. Melodic contour and musical structure. In P. Howell, I. Cross, and R. West, editors, *Musical Structure and Cognition*. Academic Press Inc., London, 1985.

[6] A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith. Query by humming - musical information retrieval in an audio database. In *ACM Multimedia 95*, 1995.

[7] K. Lemström and P. Laine. Musical information retrieval using musical parameters. In *Proceedings of the 1998 International Computer Music Conference*, pages 341–348, 1998.

[8] R.J. McNab, L.A. Smith, D. Bainbridge, and I.H. Witten. The New Zealand digital library MELody inDEX. *D-Lib Magazine*, 1997.

[9] L.B Meyer. *Style and Music. Theory, History and Ideology.* University of Pennsylvania Press, Philadelphia, 1989.

[10] E. D. Scheirer. Tempo and beat analysis of acoustic musical signals. *J. Acoust. Soc. Am*, 103(1):588–601, January 1998.

[11] T. Sonoda. A www-based melody retrieval system. In *Proceedings of the 1998 International Computer Music Conference*, pages 349–352, 1998.

[12] A. L. Uitdenbogerd and J. Zobel. Manipulation of music for melody matching. In *ACM Multimedia 98 Proceedings*, pages 235–240, 1998.

[13] E. Ukkonen. Algorithms for approximate string matching. *Information and Control*, 64:100–118, 1985.