Evaluation of Distance Measures for MPEG-7 Melody Contours

Jan-Mark Batke, Gunnar Eisenberg, Philipp Weishaupt*, and Thomas Sikora
Communication Systems Group, Technical University of Berlin
email: {batke, eisenberg, sikora}@nue.tu-berlin.de, *philippweishaupt@web.de

Abstract— **In Query by Humming (QBH) systems the melody contour is often used as a symbolic description of music. The MelodyContour Description Scheme (DS) defined by MPEG-7 is a standardized representation of melody contours. For melody comparison in a QBH system a distance measure is required. This paper evaluates different distance measures for the MPEG-7 MelodyContour DS. The usability of each measure is discussed.**

I. Introduction

A music information retrieval (MIR) system can provide several means for music retrieval [1]. To search for a piece of music a short exerpt of an audio file can be used. An audio fingerprint of the excerpt is created and matched against a database, see FOOTE's work for an example [2]. Other descriptions like the score of the melody, genre classification or the artist's name are also possible to search for a piece of music.

A query by humming (QBH) system is a specialized MIR system. It enables the user to hum a melody into a microphone. The recorded signal is processed and the extracted melody is compared with melodies residing in a database. A result list of for example ten best matching titles is presented to the user.

The success of such a database query highly depends on the internal representation of the music. For melodies, a symbolic representation like music notation is commonly used. Score formats like Guido [3] can be used for MIR-systems. However, score notation is often too detailed for QBH systems. Therefore the melody contour representation is often used. The simplest form is to use three contour values describing the intervals from note to note, up (U), down (D) and repeat (R). Coding a melody using U, D, and R is also known as PARSON-Code [4]. A more detailed contour representation is to represent the melody as a sequence of changes in pitch. [5] In this relative pitch or *interval method*, each note is represented as a change in pitch form the prior note, e.g. providing the number of semitones up (positive value) or down (negative value). A variant on this technique is *modulo interval*, in which changes of more than an octave are reduced by twelve.

So far, no rhythmical features are taken into account. However, rhythm can be an important feature of a melody. The international standard MPEG-7 provides a five step melody contour representation, which also includes rhythmical information [6]. Thus, rhythmical information can be taken into account, as well. A distance measure is required for comparing melody contours. In this paper, several distance measures for use with the MPEG-7 Melody Contour DS are evaluated. These distance measures can use melody contour information, rhythmic information or both.

For our evaluation the QBH system Queryhammer is used [7]. In the next section all processing stages of Queryhammer which are used for query construction are described. Subsequently, the theoretical background for all distance measures used in this evaluation is given (section III). Advantages and disadvantages of each distance measure are given in section IV.

II. A query by humming system example

The query construction highly depends on the QBH system used. Therefore we briefly describe our QBH system Queryhammer used for our evaluation. A detailed description can be found in [7].

The architecture of the system is depicted in figure 1. A microphone takes the hummed input and sends it as PCM signal to the extraction part. The extracted information is given to the transcription part which in turn forms an MPEG-7 compliant representation to be compared with all contours residing in the database. Finally an ordered result list is presented to the user.

The extraction block generates all necessary information from the PCM input. The pitch contour requires information of the fundamental frequency of the hummed input. This information is extracted using a pitch detection algorithm as described in [8]. A more challenging task is the extraction of tempo information. Queryhammer uses the algorithm described in [9] for tempo extraction.

The extracted information is used by the transcription block to form an MPEG-7 MelodyContour representation. The output format of the transcription block, the MPEG-7 `MelodyContourType`, is shown in figure 2. It contains two vectors, `Contour` and `Beat`. The vector `Contour` contains a 5-level pitch contour representation of the melody [6] using values as shown in table I. The vector `Beat` contains the beat numbers where the contour changes take place, truncated to whole beats. The beat information is stored as a series of integers.

Figure 3 shows an example how a melody results in the described MPEG-7 values.

0-7803-8578-0/04/$20.00 ©2004 IEEE

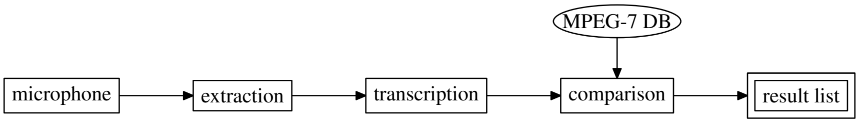


Fig. 1. The stages of the QBH system Queryhammer. A MPEG-7 compliant melody contour is extracted from the user input and compared with the MPEG-7 database content.

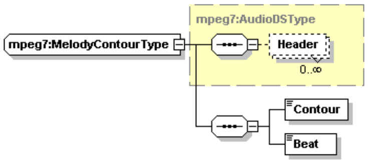


Fig. 2. The MPEG-7 MelodyContourType [10]. `Contour` holds the melody contour, `Beat` contains the beat numbers where the contour changes.

TABLE I

Melodic contour intervals defined for 5 step representation. The deviation of pitch is given in cent (1 cent is one-hundredth of a semitone).

Contour value	Change of c(f) in cents
-2	$c \leq -250$
-1	$-50 \leq c < -250$
0	$-50 < c < 50$
1	$50 \leq c < 250$
2	$c \geq 250$

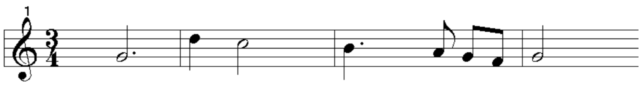
III. Distance measures

Several distance measures are evaluated in this paper. Our attention is turned to their usability for the comparison of MPEG-7 compliant melody contours.

To compare two melodies, different aspects of the melody representation can be used. Often, algorithms only take into account the contour of the melody, disregarding any rhythmical aspects. Another approach is to compare two melodies solely on the basis of their rhythmic similarity. Furthermore, melodies can be compared using contour and rhythm. McNab et al. also discuss other combinations like interval and rhythm [11].

Uitgenbogerd and Zobel presented different matching strategies [5], [12]. In their papers, matching techniques are divided into distances of the *longest common substring*, the *longest common subsequence* and *local alignment*. They experiment with a variety of these distance measures using contour representations only. However, they do not use a 5-step melody contour.

Two melodies represented in an MPEG-7 compliant manner can also be compared by only taking the melody's rhythmic properties into account. The limitations of the MPEG-7 beat vectors allow a computationally efficient comparison algorithm named *direct measure* [13]. Kim presents a distance measure that takes contour and rhythm into account [14].



```
<MelodyContour>
  <Contour>2 -1 -1 -1 -1 -1 1</Contour>
  <Beat>1 4 5 7 8 9 9 10</Beat>
</MelodyContour>
```

Fig. 3. Example: Moon River by Henry Mancini (top), MPEG-7 encoded melody contour (bottom) [6]

The distance measures evaluated in this paper are:

1) A flexible method to match the longest common substring is to use *n-grams*. An n-gram is a sequence of *n* consecutive elements of a string. Similarity is based on whether n-grams of the query also occur in the melody [12]. *Coordinate matching* simply counts these distinct n-grams.
2) *Sum of frequencies* is comparable to *coordinate matching*, but bases similarity on the *frequency of occurrence* of each of the n-grams [12].
3) Ghias [15] uses an algorithm presented by Baeza-Yates for *string matching with mismatches* [16].
4) *Local alignment* matches both melodies in a more complex way. Similarity of any two parts of the two strings is calculated by assigning costs and values to different operations like insert, delete, match, and mismatch [5]. Dynamic programming is used to determine the best match of the two strings on a local basis.
5) *Direct measure* is an efficiently computable distance measure based on dynamic programming [13]. It compares only the melodies' rhythmic properties.
6) Kim presents the algorithm TPBMI that compares the two melody's contour values *of each beat* and sums these *beat similarity scores* up to receive an overall similarity [14].

We implemented these algorithms, as the experiments done in the publications above manifested them as being superior and highly useful for this task.

A. N-gram techniques

N-gram techniques involve counting the common (or different) n-grams of the query and melody to arrive at a score representing their similarity [12]. A melody contour described by M interval values is given by

$$C = [m(1), m(2), \ldots, m(M)] \tag{1}$$

To create a n-gram of length N we build vectors

$$G(i) = [m(i), m(i+1), \ldots, m(i+N-1)], \tag{2}$$

containing N consecutive interval values, where $i = 1 \ldots M - N + 1$. The total amount of all n-grams is $M - N + 1$.

Q represents the vector with contour values of the query, and P is the piece to match against. Let Q_N and P_N be the sets of n-grams contained in Q and P, respectively.

Coordinate matching (CM) counts the n-grams $G(i)$ that occur in both Q and P:

$$R_{\text{CM}} = \sum_{G(i) \in Q_N \cap P_N} 1 \tag{3}$$

Sum of frequencies (SF) on the other hand counts how often the n-grams $G(i)$ common in Q and P occur in P:

$$R_{\mathrm{SF}} = \sum_{G(i) \in Q_N \cap P_N} U(G(i), P) \tag{4}$$

where $U(G(i), P)$ is the amount of occurrences of n-gram $G(i)$ in P.

B. String matching with mismatches

Since the vectors Q and P can be understood as strings, also string matching techniques can be used for distance measurement. Baeza-Yates and Perleberg [16] describe an efficient algorithm for string matching with mismatches which is suitable for QBH systems. String Q is sliding along string P, and each character $q(n)$ is compared with its corresponding character $p(m)$. R contains the highest similarity score after evaluating P.

C. Local Alignment

The dynamic programming approach *local alignment* determines the best match of the two strings Q and P [5], [12]. This techniques can be varied by choosing different penalties for inerstions, deletions, and replacements.

D. Direct measure

MPEG-7 Beat vectors have two crucial limitations which enable the efficient computation of a distance measure called *direct measure* [13]. All of the vector's elements are positive integers and every element is equal or bigger than its predecessor. The *direct measure* which is robust against single note failures can be computed by the following iterative process for two beat vectors U and V:

1) Compare the two vector elements $u(i)$ and $v(j)$, (starting with $i = j = 1$ for the first comparison)
2) If $u(i) = v(j)$ the comparison is considered a match. Increment the indices i and j and proceed with step 1.
3) If $u(i) \neq v(j)$ the comparison is considered a miss.
    a) If $u(i) < v(j)$ increment only the index i and proceed with step 1.
    b) If $u(i) > v(j)$ increment only the index j and proceed with step 1.

The comparison-process should be continued until the last element of one of the vectors has been detected as a match or the last element in both vectors is reached. The Distance R is then computed as the following ratio with M being the number of misses and V being the number of comparisons.

$$R = \frac{M}{V} \tag{5}$$

E. TPBM I

The algorithm *TPBM I* (time pitch beat matching) is described in [17], [14]. It uses melody and beat information plus time signature information as a triplet TPB. The time signature information is ignored in our evaluation.

IV. Evaluation of distance measures

Our test setup consists of three query databases and one song database. The song database comprises 406 MIDI files retrieved from the Internet. Included are the German Top Ten Hits of March 2003 (pop music).

The first query database is made up of the melodies of the refrains of the ten Top Ten MIDI songs. The other two databases contain queries hummed by four singers. For the second database singers were asked to hum the exact refrain of each of the Top Ten songs, for the third database singers could hum an arbitrary part of each of the ten songs. They were not given any restrictions. Note that this experimental setup is most challenging for the query system. Not two users used the same melodies, nor a metronome click was provided while humming.

MPEG-7 compliant melody contour sets of all songs and queries are automatically extracted with Queryhammer. Distance measures were evaluated matching each query with each song of the database.

The similarity scores depend on the parameter normalization and also on the parameter N for n-gram distance measures. We tested the following normalizations: none, length, log, ninth, fifth, and third root of the length of the contour vector. n-gram lengths of 3, 4, 5, 6, and 7 are evaluated.

The first simulation matches all MIDI queries of the first database with each of the 406 songs of the song database. All six distance measures are used. As expected for these "perfect" queries, the results are very good. For the distance measure *coordinate matching* with an n-gram length of four to seven the query song is always at position one in the result list. With n-gram length three some queries do not appear at position one. This makes an assessment of the optimal normalization method possible. It can be seen that the results get better the less the normalization affects the distance score. Division by the ninth root yields best results. This confirms the results in [12]. *Sum of frequencies* is more susceptible to n-gram length. With increasing n-gram length results improve with $N = 7$ yielding best results. Still, five out of ten are not at position one of the result list as was the case for *coordinate matching*.

The results for string matching with mismatches are very good for all four normalizations normalizations log, 3rd root, fifth root, and ninth root. The query song is always at position one of the results list. This is also the case for the distance measures *local alignment* and TPBM I.

Figure 4 depicts the average distances for queries of the second database, also for all distance measures. Ninth root normalization is used for all distance measures except TPBM I using log. Best results are obtained with *local alignment*. TPBM I results worst because of weak time information in the singing input. *Direct measure*, on the other hand, yields results as good as *sum of frequencies*.

Finally the third query database is used. The results for these arbitrary queries are worse than for the queries of the second query database. This is due to several factors. On the one hand the queries are very short, encompassing only one

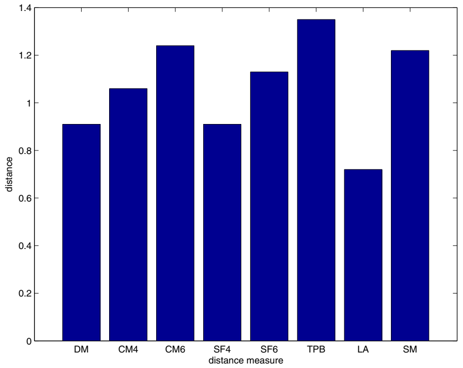


Fig. 4. Averaged distances for all compared distance measures. The singers were asked to hum the melody as contained in the MIDI queries.

to ten contour values. Then, the queries are rather imprecise. The singers hummed mainly from memory and did not listen closely to the actual song before recording their query. This was intended to simulate a real time environment for the QBH-system. Furthermore, with the same intend, singers were not restricted in the way they hummed. Several notes of the queries turned out too short for the extraction algorithm to work properly. Also some queries were hummed too fast to allow adequate extraction by Queryhammer.

The results for the distance measures *local alignment*, string matching with mismatches, and TPBM I are useless with almost all queries having a distance larger than ten. For the n-gram methods *coordinate matching* and *sum of frequencies* the results improve with increasing n-gram length. However, for n-gram length five and higher in many cases the hummed song has zero similarity to the query. This is mainly due to the short queries. Again, normalization with the ninth root proves to be a good choice as already shown in the other simulations.

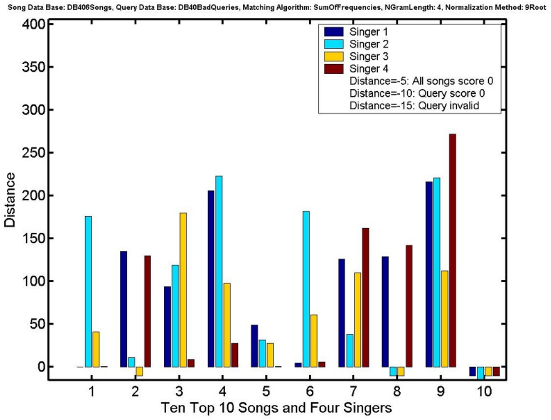


Fig. 5. Distances for arbitrary queries. Best results are obtained using distance measure *sum of frequencies*, n-gram length 4, and ninth root normalization.

V. Conclusions

Distance measures from different publications were tested. TPBM I and *direct measure* require temporal information and are only useful for QBH systems providing a metronome click to the user. For free singing, good results are obtained using the n-gram methods *sum of frequencies* and *coordinate matching*.

String matching with mismatches and *local alignment* require relatively long queries while users tend to make short queries. For sufficiently long queries, *local alignment* yields good results.

The MPEG-7 melody contour representation contains both, contour and beat information. The only algorithm related to this representation, TPBM I, is practical only for queries extracted from MIDI information. On the other hand, *direct measure* provides useful results from beat information only, extracted by Queryhammer. Therefore, a combination of a contour only and beat only measurement is likely to benefit from the MPEG-7 melody contour representation.

References

[1] G. Haus and E. Pollastri, "A multimodal framework for music inputs," in *Proceedings of the 8th ACM International Conference on Multimedia*. Los Angeles: ACM, 2000.

[2] J. T. Foote, "Content-based retrieval of music and audio," in *Proc. SPIE*, vol. 3229, 1997.

[3] H. H. Hoos, K. Renz, and M. Görg, "Guido/mir — an experimental musical information retrieval system based on guido music notation," in *Proceedings of the Second Annual International Symposium on Music Information Retrieval*, 2001.

[4] L. Prechelt and R. Typke, "An interface for melody input," *ACM Transactions on Computer-Human Interaction*, vol. 8, no. 2, pp. 133–149, 2001.

[5] A. L. Uitdenbogerd and J. Zobel, "Matching techniques for large music databases," in *Proceedings of the ACM Multimedia Conference*, D. Bulterman, K. Jeffay, and H. J. Zhang, Eds., Orlando, Florida, Nov. 1999, pp. 57–66.

[6] *Information Technology – Multimedia Content Description Interface – Part 4: Audio*, ISO/IEC, June 2001, 15938-4:2001(E).

[7] J.-M. Batke, G. Eisenberg, P. Weishaupt, and T. Sikora, "A query by humming system using mpeg-7 descriptors," in *Proc. of the 116th AES Convention*, Berlin, May 2004.

[8] P. Boersma, "Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound," in *IFA Proceedings 17*, 1993.

[9] E. D. Scheirer, "Tempo and beat analysis of acoustic musical signals," *J. Acoust. Soc. Am.*, vol. 103, no. 1, pp. 588–601, January 1998.

[10] B. S. Manjunath, P. Salembier, and T. Sikora, Eds., *Introduction to MPEG-7*, 1st ed. West Sussex, England: Wiley, 2002.

[11] R. J. McNab, L. A. Smith, I. H. Witten, C. L. Henderson, and S. J. Cunningham, "Towards the digital music library: Tune retrieval from acoustic input," in *Proceedings of the first ACM international conference on Digital libraries*. Bethesda, Maryland, United States: ACM, 1996, pp. 11–18.

[12] A. Uitdenbogerd and J. Zobel, "Music ranking techniques evaluated," in *Proceedings of the Australasian Computer Science Conference*, M. Oudshoorn, Ed., Melbourne, Australia, Jan. 2002, pp. 275–283.

[13] G. Eisenberg, J.-M. Batke, and T. Sikora, "Beatbank – an mpeg-7 compliant query by tapping system," in *Proc. of the 116th AES Convention*, Berlin, May 2004.

[14] Y. E. Kim, W. Chai, R. Garcia, and B. Vercoe, "Analysis of a contour-based representation for melody," in *Proc. International Symposium on Music Information Retrieval*, October 2000.

[15] A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith, "Query by humming: Musical information retrieval in an audio database," in *ACM Multimedia*, 1995, pp. 231–236.

[16] C. P. R. Baeza-Yates, "Fast and practical approximate string matching," *Combinatorial Pattern Matching, Third Annual Symposium*, pp. 185–192, 1992.

[17] W. Chai and B. Vercoe, "Melody retrieval on the web," in *Proceedings of ACM/SPIE Conference on Multimedia Computing and Networking*, January 2002.