MIREX 2008: QUERY BY TAPPING

Rainer Typke[1] ,
Austrian Research Institute for Artificial Intelligence (OFAI), Vienna

ABSTRACT

In order to compare rhythmic patterns with a dissimilarity measure that is continuous and supports partial matching, one can adapt the Earth Mover's Distance (EMD). Partial matching is a useful feature in situations where, as in this year's "QBT" MIREX task, a query is given in the form of an audio file from which onsets are extracted. The onset detection might lead to both false positives and false negatives; in both cases, it should be possible to ignore the onsets for which no corresponding onsets exist in an otherwise similar group of onsets.

1 COMPARING RHYTHMIC PATTERNS USING THE EARTH MOVER'S DISTANCE

1.1 Representation of rhythmic patterns as point sets, normalization, EMD

Rhythmic patterns are compared as follows:

1. Represent onset times as a sequence of one-dimensional weighted points. The coordinate of each point reflects the onset time, and the weight is constant (each onset gets weight 1).

2. Before applying the EMD, normalize two point sets by setting the first and last onset to fixed numbers and adjusting the points in between such that the proportions of onset times are preserved.

3. Calculate the Earth Mover's Distance.
$\mathrm{EMD}_d(A,B) = \frac{\min_{F\in\mathcal{F}} \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} d_{ij}}{min(W_A, W_B)}$.
(d: ground distance, A, B: weighted point sets; F: a flow; $\mathcal{F}$: all possible flows; f_{ij}: amount of mass carried from point i to point j; d_{ij} : ground distance between points i and j; W_A, W_B: total weight of set A/B). For details, see [1].

See Figure 1 as an illustration of how rhythmic patterns can be represented as sequences of onset times. Also, this example illustrates a situation where partial matching is desirable: in the third measure, the bottom variant has one more note than the top variant, but otherwise, the rhythms are still quite similar.

[1] R. Typke gratefully acknowledges support by the Austrian Research Fund (FWF), Project Number M1027-N15.

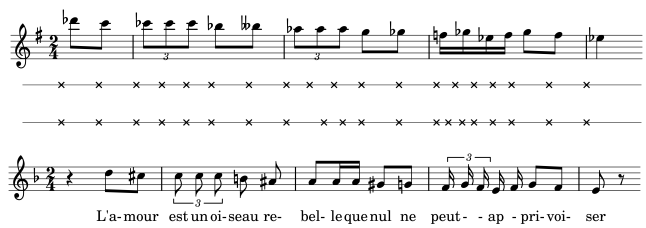

Figure 1. Comparing rhythms using sequences of onset times. *Top:* Ferruccio Busoni: Sonatina No.6: Chamber Fantasia on Bizet's Opera Carmen (3rd theme); *bottom:* Georges Bizet: Carmen: Habanera; *middle:* two sequences of onset times representing the incipits at the top and bottom. A possible numeric representation of the two series of onset times: **Busoni**: (0, 3, 6, 8, 10, 12, 15, 18, 20, 22, 24, 27, 30, 31.5, 33, 34.5, 36, 39, 42); **Bizet**: (0, 3, 6, 8, 10, 12, 15, 18, 21, 22.5, 24, 27, 30, 31, 32, 33, 34.5, 36, 39, 42).

1.2 Finding rhythms anywhere in a piece: Segmentation

Section 1.1 explains how one can compare rhythmic patterns of approximately the same length. Usually, for finding pieces of music containing a given rhythmic pattern, one will want to find relatively short patterns that are contained somewhere within a large piece of music. To achieve this, one can segment a long onset sequence representing a whole piece of music into short, overlapping segments, possibly of varying length. The query can then be compared either directly to those segments (if its length is similar to the segment length), or it can also be segmented in the same way.

1.3 Optimized vantage indexing and tunneling

The EMD is computationally expensive, especially for large point sets. The variant described above (EMD on sets of 1-dimensional points, where every point has weight 1), which we call "Manhattan EMD", is a prametric; the triangle inequality does not hold, and a distance of zero does not imply identity. This makes it tricky to build a good index.

In [1], we show how one can achieve logarithmic complexity for searching r-near neighbours with the Manhattan EMD without any false negatives and reasonably few false positives. The next few paragraphs will give a brief overview.

1.3.1 *Optimized vantage indexing for the l_1 norm*

The Manhattan EMD has metric subspaces: on sequences of equal length, the Manhattan EMD is equivalent to the l_1 norm. Therefore, one can use vantage indexing for searching those metric subspaces.

Vantage indexing guarantees that no false negatives occur if the triangle inequality holds; however, false positives are possible.

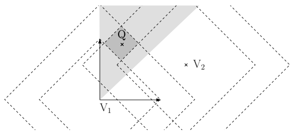

Figure 2. In this two-dimensional example, only the light gray area is inhabited by database objects since only there the second coordinate is greater than the first. The dark gray ball around Q with a given radius can be created by intersecting onion layers around V_1 and V_2 whose thickness equals the radius.

For the l_1 norm, one can eliminate all false positives by ensuring that the vantage objects are placed in a way that one can create exactly the interesting r-neighbourhood around any query (and with any radius r) by intersecting onion-like layers with thickness r around the vantage objects. For this to work, the vantage objects need to be placed in an optimum way. For example, one can put them into the corners of the area that can possibly contain data items (this is a finite area because all coordinates are restricted to a certain range due to the normalization).

1.3.2 *Searching across dimensions with a tunneling technique*

To search the whole Manhattan EMD space, instead of just metric subspaces, for r-near neighbours of a query, one can pre-calculate connections between nearest neighbours in different subspaces. One should probably limit the number of dimensions to cross, both for keeping the database small enough and for keeping search results meaningful.

If such connections between nearest neighbours in different subspaces exist, one can search as follows:

1. Search the query's own metric subspace using optimum vantage indexing.

2. For searching higher metric subspaces, follow the links from every item that was retrieved in the query's own subspace into higher-dimensional subspaces, and check whether the linked items lie within the search radius.

3. Lower-dimensional subspaces can be searched by using optimized vantage indexing for finding items that are close to projections of the query into the subspace (or, alternatively, close to points that are linked to the query by a tunnel).

Figure 3 illustrates how tunneling can introduce false positives and false negatives. In this example, we first search Q's own metric subspace for near neighbours and find P_4 and P_5. By following their respective links, we find the higher-dimensional points P_1 (a false positive) and P_2 (a true positive). We do not, however, find P_3 (which is therefore a false negative).

By adding the projections of higher-dimensional points (here shown as hollow circles) to the database if their distance to the nearest lower-dimensional neighbour lies above a threshold t and at the same time increasing the search radius by t, we can completely eliminate all false negatives. The distance error of false positives is limited to $2t$. Since false positives can now only occur near the surface of balls, the precision grows with the search radius.

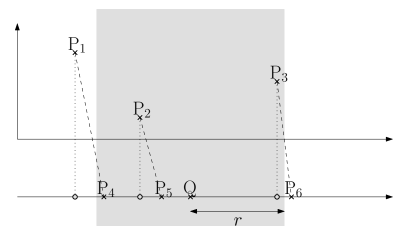

Figure 3. False negatives and false positives resulting from tunneling, and how to avoid them.

2 IMPLEMENTATION FOR MIREX 2008

For the "Query by Tapping" task at MIREX 2008, it is known that we are only looking for rhythmic patterns at the beginning of MIDI files. Therefore, it is not necessary to segment the data or the queries and find matching segments; instead, it is sufficient to take a certain constant number of onsets from each query (we use 17 since in most queries, there are enough onsets for making this a good number). We match these sets of 17 points (or fewer, if the query was shorter) with the first 16 to 20 notes in each MIDI file. That is, we allow up to one extra note in the query or up to three extra notes in the data to be matched.

Since the database is small (little over 100 pieces), we do not use the indexing technique described in Section 1.3. This has the consequence that actually, if the query happens to be shorter than 17 notes, we allow even more extra notes in the data to be matched – up to 18 extra notes if the query has only 2 notes. This high number of allowed extra notes might somewhat decrease the result quality for short queries. Introducing a small penalty for crossing dimensions could possibly lead to still better results.

Our algorithm matches the given ".onset" text files to MIDI files; we do not attempt to extract onsets from audio files ourselves.

3 MIREX 2008 RESULTS

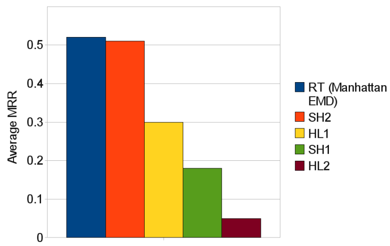

Figure 4. Average mean reciprocal ranks; the Manhattan EMD performs best among 5 submitted algorithms.

The MIREX 2008 results (see Figure 4) indicate that the Manhattan EMD is indeed suitable for comparing rhythmic patterns in cases where partial matching is needed (because of misdetected onsets) and where continuity is useful (because of rhythmic inaccuracies that are introduced by manual tapping).

4 REFERENCES

[1] R. Typke and A. Walczak-Typke. A tunneling-vantage indexing method for non-metrics. In *Proceedings of the Eighth International Conference on Music Information Retrieval (ISMIR)*, 2008.