Finding Cover Songs by Melodic Similarity

Christian Sailer and Karin Dressler
Fraunhofer IDMT
Langewiesener Str. 22
98693 Ilmenau/Germany
{sar,dresslkn}@idmt.fraunhofer.de

Abstract

This paper describes the submission to the MIREX 06 (Music Information Retrieval EXchange) Audio Cover Song task delivered by Fraunhofer IDMT. A method to detect cover songs by comparing the most salient melodies of musical pieces is proposed, based on the assumption that cover versions of one song contain the same melody. To this end, the predominant melody of the musical pieces to be considered is extracted and characteristic parts are sought for. The melodic similarity between the pieces is calculated and derived from these values, a distance matrix is constructed.

Keywords: MIREX, Audio Cover Song, melodic similarity, melody extraction, melody alignment

1. Introduction

The basic idea behind this approach is that a cover song is by definition a new rendition of a previously recorded song. As it is a version of the same song, it should carry the same melody as one of the conceptual features inherent to the musical entitiy of song. Considering this, it is even more probable that the covered song has the same melody (or some variation of it) as the original version, than having a similar sound or instrumentation. This may easily be seen by comparing e.g. Tori Amos' version of Nirvana's "Smells Like Teen Spirit" to the original.

Thus, a method to find cover versions by comparing the salient melodies of musical pieces is proposed.

2. Implementation Overview

The algorithm is implemented in C++ and is available for Linux and Windows platforms. It consists of two parts, an indexing program (a) and the actual retrieval tool (b).

(a) extracts the melody (predominant voices) from audio files, processes them to relevant pieces and stores them in a data base. The algorithm works sample rate agnostic and reads wav, aiff and mp3 files. The run time is linear with the duration of audio input and is about 10-12 times faster than real time [1].

(b) reads the data base generated by (a) and calculates the distance matrix as is described in the requirements for the Audio Musical Similarity, respectively the Audio Cover Song task on the MIREX 2006 site [2]. As this is a quadratic matrix where each piece is compared to each other, the runtime for N pieces is $O(N^2)$.

3. Indexing

3.1. Pitch Detection

The first step of the melody extraction is using a multi resolution FFT and the algorithm proposed by Dressler in [1, 2]. It yields a voiced/non voiced detection in conjunction with a pitch line (pitch in cent over time) for the voiced parts.

3.2. Melody Quantization

In a process derived from the melody segmentation works of Heinz [3], note boundaries are estimated from the pitch line and further spectral information. The resulting note candidates are then subjected to a plausibility test where objects of too short duration or insufficient loudness are discarded. Finally, a discrete pitch is estimated for each note candidate, providing melody information on a note basis.

3.3. Relevance Weighting

There are several reasons why storing the melody over the entire length of the song is not desirable. To begin with, a lot of processing time is used for the data base search if both the query and the reference are very large. Furthermore, it is more probable to receive relevant results by matching several short excerpts of one piece against another melody instead of matching entire melodies, as structural differences within a musical piece or between versions may lead to confusing results when comparing whole melodies.

For these reasons, the extracted melodies are split into relevant pieces. In order to find such pieces, parts of the extracted melody are sought that fulfil the requirements to be a musical theme: They must be between 3 and 8 seconds long, and consist of neither too few nor too many notes [4]. Following the idea that at least in western popular music,

[1] All runtimes are measured on a 3GHz Intel Pentium IV
[2] See http://www.music_ir.org/mirex2006

important themes (like verse, chorus, etc.) are usually repeated, these theme candidates are looked up within the same piece of music and weighted according to their number of appearances within the piece.

4. Evaluation

To evaluate the indexed data, up to three excerpts of the melody are matched against all entries in the data base of indexed data, using a melody alignment algorithm that has been developed for a Query by Humming system [5].

4.1. Melody Alignment

The look-up is carried out as a string alignment process (see [6] for a general explanation), which has been adapted for melody search [7]. As basic search alphabet, the relative change of a melody over time is used. Thus, not notes but descriptions of note transitions represented by note intervals and the ratio of inter-onset intervals are considered. This frees the search algorithm from a dependency on absolute tempo and pitch – attributes that may vary between different versions of a piece of music.

The alignment is carried out as a semi-local alignment, meaning that the whole query string must match any part of the reference string. It returns a value which is the higher, the better the query matches to the reference.

In a post processing step, the contours of the matching part of the reference are matched to the contour of the query, and a correction of the alignment value is carried out.

As the resulting alignment values depend on the size of the query string (the longer the string, the greater the maximum possible value), the values are finally normalized to the range of $[0..1]$.

4.2. Distance Matrix

The melody alignment yields a melodic similarity measure s_{ab} between two melodies a and b ranging from 0 (no similarity at all) and 1 (equality). To get the distance d_{ab} as it is required by the Audio Cover Song task, all similarities where $s_{ab} = 0$ are set to $s_{ab} = 10^{-7}$, and then the distance is calculated as $d_{ab} = \frac{1}{s_{ab}} - 1$, thus being a non-negative number between 0 and 10^7.

5. Results and Discussion

An overview of the results [3] of the Audio Cover Song evaluation can be found in table 5. The test bed consists of a set of about 5000 songs, embedded into which are 30 songs along with 10 cover versions of each of these song.

Each of these 330 versions is used as query and the ten most similar songs returned are evaluated

As can easily be seen, the submitted algorithm yields a higher success rate on dectecting cover versions than the musical similarity algorithms evaluated for this task. Considering the fact that cover versions do not necessarily have the same sound, and often are intentionally rearranged, this may not be a big surprise. On the other hand, it yielded lower success rates than algorithms that have been carefully crafted to detect cover versions based on an extensive structural and musical analysis of the audio tracks.

[3] Please see http://www.music-ir.org/mirex2006/index.php/MIREX2006_Results for full results and further information on contestants

Table 1. Overview of the results of the Audio Cover Song task, this submission in bold. Entries marked with * are musical similarity algorithms. The first result line shows total cover versions found out of 3300 possible, the second line shows inverse average rank of best cover version

CS	DE	KL1	KL2	KWL*	KWT*	LR*	TP*
211	761	365	314	117	102	149	116
0.21	0.49	0.22	0.22	0.10	0.10	0.12	0.09

Several problems could be identified for this algorithm: Considering just one feature, in this case melody, makes the algorithm vulnerable to extraction errors as well as to cases where melody is not the key feature of a song or has been altered in the cover version. Additionally, the melody extraction, though state of the art, is far from being perfect. The note segmentation required for the qbh algorithms to work is highly dependent on the quality of pitch extraction and furthermore adds its own errors. Finally, the segmentation of the melody in theme like pieces is based solely on the melody information previously extracted in this approach, which leads to spurious results in some cases. Recapitulating, it becomes obvious that error propagation from the first steps, promoted by a strong dependency on exact inputs by later stages, diminishes the overall performance of the system.

6. Conclusive Remarks

With this submission, it could be shown that melody is an important feature for detecting cover versions, and that automatically extracted melodies can be a handy tool for this task. However, it also became clear that automatically extracted melodies alone are not sufficient to conduct this task satisfactorily. Besides optimising the melody extraction and segmentation, an interesting prospect will be the integration of such a melody similarity assessment into a larger, multi feature system, using melody as one feature as well as using a song segmentation for detecting the relevant musical pieces.

7. Acknowledgements

We would like to thank the ISMIRSEL team for the organization and running of this evaluation.

Work on the Query by Humming algorithms used in this approach was partially funded by the EU-Project Semantic

Hifi (FP6-507913) [4]. Karin Dressler's work on melody extraction was partially funded by a scholarship of the state of Thuringia.

References

[1] K. Dressler, "Sinusoidal extraction using an efficient implementation of a multi-resolution FFT," in *Proc. of the Int. Conf. on Digital Audio Effects (DAFx-06)*, Montreal, Quebec, Canada, Sept. 18–20, 2006, pp. 247–252.

[2] ——, "Audio melody submission," in *MIREX 06*, 2006.

[3] T. Heinz, "Ein physiologisch gehörgerechtes verfahren zur automatisierten melodietranskription," Ph.D. dissertation, Technische Universität Ilmenau, 2006.

[4] D. Temperley, *The Cognition of Basic Musical Structures*. The MIT Press, 2001.

[5] C. Sailer, "Two note based approaches to query by singing/humming," in *MIREX 06*, 2006.

[6] D. Gusfield, *Algorithms on Strings, Trees and Sequences*. Cambrigde University Press, 1997.

[7] C. Sailer, "Using string alignment in a query-by-humming system for real world applications," Talk at 150th ASA Fall Meeting, October 2005.

[4] See http://shf.ircam.fr