A CHROMA-BASED TEMPO-INSENSITIVE DISTANCE MEASURE FOR COVER SONG IDENTIFICATION USING THE 2D AUTOCORRELATION

Jesper Højvang Jensen
Aalborg University
Dept. Electron. Syst.
jhj@es.aau.dk

Mads G. Christensen
Aalborg University
Dept. Electron. Syst.
mgc@es.aau.dk

Søren Holdt Jensen
Aalborg University
Dept. Electron. Syst.
shj@es.aau.dk

ABSTRACT

In the following, we describe the refined version of our 2007 cover song identification algorithm [3] submitted for the 2008 MIREX cover song identification task. The refined algorithm is faster and has higher recognition performance than the original version.

1 INTRODUCTION

The MIREX cover song identification task was introduced in 2006 and has since been quite popular. In 2007, we introduced a cover song detection algorithm using features that were insensitive to changes in instrumentation, tempo and time shifts. If it had attended the 2006 cover song detection contest, it would have performed on par with the winner, despite being much faster. However, in 2007 standards had increased tremendously, leaving our algorithm on a fourth place of eight contenders. Our 2008 algorithm uses the same principles as our 2007 algorithm but is even faster and performs slightly better on the covers80 data set [1]. The source code of the algorithm is available as part of the Intelligent Sound Processing toolbox [1] .

2 FEATURE EXTRACTION

It is assumed that a song and its cover versions share the same melody, but might differ with respect to instrumentation, time shifts, tempo and transpositions. The extracted featuere is thus insensitive to changes in instrumentation, tempo and time shifts. Feature extraction comprises the following steps:

1. Compute the chromagram functions $c_s^{(1)}(n)$, $s \in 0, \cdots, 11$ and $n \in 0, \cdots, N$, from the sampled song using the code from [2]. The value $c_s^{(1)}(n)$ is the power of semitone s at time n. As the chromagram primarily captures melodic information, it is somewhat insensitive to differences in instrumentation.

2. To reduce the influence of peaks, compute $c_s^{(2)}(n) = |c_s^{(1)}(n)|^{0.7}$.

3. To remove any DC offset, filter all $c_s^{(2)}(n)$ by the high-pass filter $h(n)$:

$$c_s^{(3)}(n) = c_s^{(2)}(n) * h(n) \tag{1}$$

4. In order to obtain a representation insensitive to temporal alignment and key, we compute the 2D autocorrelation function $R^{(4)}(m,k)$:

$$R^{(4)}(m,k) = \sum_{s=0}^{11} \sum_{n=0}^{N} c_s^{(3)}(n) c_{s+m \mod 12}^{(3)}(n+k). \tag{2}$$

This step is different from last year's submission. Back then we computed the power spectrum of each individual $c_s^{(3)}(n)$ to avoid alignment problems.

5. By summing the values of the autocorrelation in 16 exponentially distributed bands, B_l, we obtain a representation that is insensitive to different tempi (see [3]):

$$R^{(5)}(m,l) = \sum_{k=0}^{N} R^{(4)}(m,k) B_l(k) \tag{3}$$

6. Finally, we normalize $R^{(5)}(m,l)$:

$$R^{(6)}(m,l) = \frac{R^{(5)}(m,k)}{\sqrt{\sum_{m=0}^{11} \sum_{l=0}^{16} R^{(5)}(m,l)^2}} \tag{4}$$

We end up with a 2-D function $R^{(6)}(m,l)$ where $m \in 0, \cdots, 11$ and $l \in 0, 16$, i.e., with $12 \cdot 17 = 204$ different values.

This research was supported by the Intelligent Sound project, Danish Technical Research Council grant no. 26–04–0092, and the Parametric Audio Processing project, Danish Research Council for Technology and Production Sciences grant no. 274–06–0521.

[1] http://isound.es.aau.dk/

3 DISTANCE COMPUTATION

The similarity between two songs, s_{ij}, where the songs are represented by the features $R_i^{(6)}(m,l)$ and $R_j^{(6)}(m,l)$, respectively, is given by

$$s_{ij} = \max_{d \in -1,0,1} \sum_{m=0}^{11} \sum_{l} (R_i^{(6)}(m,l+d) - R_i^{(6)}(m,l))^2. \quad (5)$$

The corresponding distance measure, d_{ij}, is given by $d_{ij} = 2 - s_{ij}$. It obeys the triangle inequality.

4 EVALUATIONS

The covers80 dataset [1] consists of 80 titles each in two different versions, i.e., a total of 160 songs. With this set, a song's nearest neighbor was the cover version in 36% of the cases for our 2007 submission. For the 2008 edition, accuracy has increased to 48%. However, as parameters have been tweaked using this dataset, some degree of overtraining is inevitable. On a 1.86 GHz Intel Core 2 CPU using only a single core, feature extraction took less than 2 seconds on average for the covers80 data set.

5 REFERENCES

[1] D. Ellis and G. Poliner, "Identifying cover songs with chroma features and dynamic programming beat tracking," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 2007, pp. 1429–1432.

[2] D. P. Ellis, "Identifying 'cover songs' with beat-synchronous chroma features," in *Music Information Retrieval Evaluation eXchange*, 2006.

[3] J. H. Jensen, M. G. Christensen, D. P. Ellis, and S. H. Jensen, "A tempo-insensitive distance measure for cover song identification based on chroma features," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 2008.