A TEMPO-INSENSITIVE DISTANCE MEASURE FOR COVER SONG IDENTIFICATION BASED ON CHROMA FEATURES

Jesper Højvang Jensen[1]*, Mads G. Christensen*[1]*, Daniel P.W. Ellis*[2]*, and Søren Holdt Jensen*[1]

[1] Dept. of Electronic Systems
Aalborg University, Denmark

[2] LabROSA, Dept. Elec. Eng.
Columbia University, USA

ABSTRACT

We present a distance measure between audio files designed to identify cover songs, which are new renditions of previously recorded songs. For each song we compute the chromagram, remove phase information and apply exponentially distributed bands in order to obtain a feature matrix that compactly describes a song and is insensitive to changes in instrumentation, tempo and time shifts. As distance between two songs, we use the Frobenius norm of the difference between their feature matrices normalized to unit norm. When computing the distance, we take possible transpositions into account. In a test collection of 80 songs with two versions of each, 38% of the covers were identified. The system was also evaluated on an independent, international evaluation where it despite having much lower complexity performed on par with the winner of last year.

***Index Terms*—** Feature extraction, Music.

1. INTRODUCTION

As the size of digital music collections increases, navigating such collections become increasingly difficult. One purpose of music information retrieval is to develop algorithms to facilitate such navigation, for instance by finding songs with similar instrumentation, rhythm or melody. Based on the initial success using MFCCs for genre classification, much research has until now directly or indirectly focused on finding songs with similar instrumentation [1–4]. With the introduction of a cover song identification contest in 2006, the Music Information Retrieval Evaluation eXchange (MIREX) community has put focus on musical structure rather than spectral statistics. In the MIREX 2006 cover song identification contest, the system in [5] had the best retrieval performance. This system had relatively high storage and computational requirements. It combines the chromagram, which is an octave-independent magnitude spectrum, with a beat tracker in order to obtain a beat-synchronous chromagram that is insensitive to differences in tempo.

Most cover song identification systems depend on estimates of musical properties and are therefore sensitive to the accuracy of the estimates. The system in [5] uses a beat estimate, [6] extracts the melody, and both [7] and [8] rely on chord recognition. Like [5, 7, 8], the proposed system is based on the chromagram, but unlike the aforementioned systems, it does not directly attempt to extract musical properties. Instead, it applies a number of transformations in order to obtain a feature that compactly describes a song and is not sensitive to instrumentation, time alignment or tempo. The feature is somewhat similar to the rhythm patterns in [9] that describe the amount of modulation in certain frequency bands, and the result is a system with performance similar to [5], but with a complexity that is heavily reduced.

In Section 2 and 3, we describe the extracted features and the distance measure between them, respectively. We evaluate the performance of the proposed system in Section 4 before giving the conclusion in Section 5.

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.

2. FEATURE EXTRACTION

The assumptions behind the proposed system are that a song and its cover versions share the same melody, but might differ with respect to instrumentation, time shifts, tempo and transpositions. We extract a feature matrix which is insensitive to the former three properties, while the distance computation ensures invariance to transpositions. In Fig. 1 and 2, examples of a signal at different stages during the feature extraction are given, and in Fig. 3 a block diagram of the process is shown. Note that except for a horizontal shift of one band, Fig. 1(c) and 2(c) are very similar.

The first stage of extracting the feature matrix is to compute the chromagram from a song. It is conceptually a short time spectrum which has been folded into a single octave [10]. This single octave is divided into 12 logarithmically spaced frequency bins that each correspond to one semitone on the western musical scale. Ideally, the chromagram would be independent of instrumentation and only reflect the notes of the music being played. We use the implementation described in [5] to compute the chromagram. We found that elementwise taking the logarithm of the chromagram increased performance, possibly because it better reflects human loudness perception. Let the chromagram matrix $\boldsymbol{Y}$ be given by

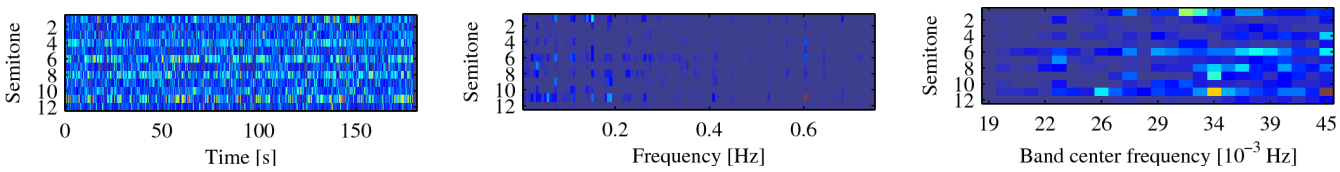
$$\boldsymbol{Y} = \begin{bmatrix} \mathbf{y}_1^{\mathrm{T}} \\ \vdots \\ \mathbf{y}_{12}^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} y_1(1) & y_1(2) & \cdots & y_1(N) \\ & \vdots & & \\ y_{12}(1) & y_{12}(2) & \cdots & y_{12}(N) \end{bmatrix} \tag{1}$$

where $y_n(m)$ represents the magnitude of semitone n at frame m. The chromagram after the logarithm operation, $\boldsymbol{Y}_{\log} = [\mathbf{y}'_1, \cdots, \mathbf{y}'_{12}]^{\mathrm{T}}$, is given by $(\boldsymbol{Y}_{\log})_{i,j} = \log(1 + (\boldsymbol{Y})_{i,j}/\delta)$, where $(\cdot)_{i,j}$ is the element of row i and column j, and δ is a small constant.

To avoid time alignment problems, we remove all phase information from $\boldsymbol{Y}_{\log}$ by computing the power spectrum for each row, i.e.,

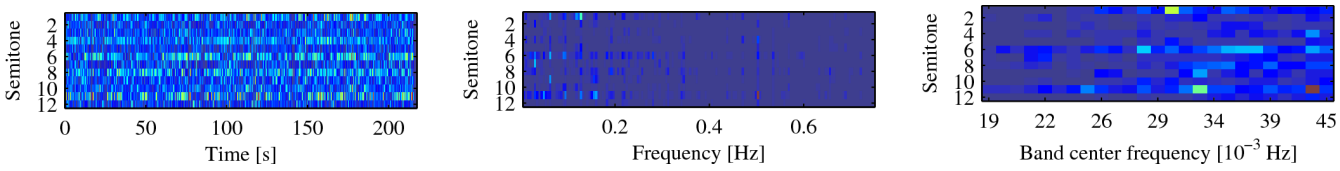
$$\boldsymbol{Y}_{\mathrm{pwr}} = \begin{bmatrix} |\mathcal{F}\{\mathbf{y}'_1{}^{\mathrm{T}}\}|^2 \\ \vdots \\ |\mathcal{F}\{\mathbf{y}'_{12}{}^{\mathrm{T}}\}|^2 \end{bmatrix}, \tag{2}$$

where $\mathcal{F}$ is the Fourier operator. This also removes all semitone co-occurence information, which may contain useful information.



(a) Chromagram after the logarithm. (b) Power spectrum of the chromagram rows. (c) Energy in the 25 exponentially spaced bands.

Fig. 1. Different stages of feature extraction from a MIDI song with duration 3:02.



(a) Chromagram after the logarithm. (b) Power spectrum of the chromagram rows. (c) Energy in the 25 exponentially spaced bands.

Fig. 2. Feature extraction from the same MIDI song as in Fig. 1, except it is stretched to have duration 3:38.

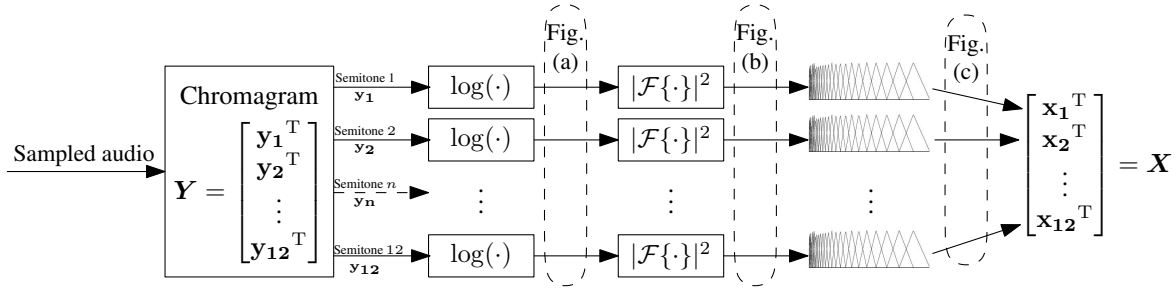


Fig. 3. Block diagram of the feature extraction process.

Moving on to temporal differences, let $x(t)$ be a continuous signal and let $X(f) = \mathcal{F}\{x(t)\}$ be its Fourier transform. A temporal scaling of $x(t)$ will also cause a scaling in the frequency domain: $\mathcal{F}\{x(kt)\} = X(f/k)$. This approximately holds for discrete signals as well and thus for the rows of $\boldsymbol{Y}_{\text{pwr}}$. For cover songs it is reasonable to assume that the ratio between the tempo of a song and its cover is bounded, i.e., that two songs do not differ in tempo more than, e.g., a factor c, in which case $\frac{1}{c} \leq k \leq c$. Now, if either the time or frequency axis is viewed on a logarithmic scale, a scaling (i.e., $k \neq 1$) will show up as an offset. This is used in e.g. [11] to obtain a representation where the distances between the fundamental frequency and its harmonics are independent of the fundamental frequency itself. If the scaling k is bounded, then the offset will be bounded as well. Thus, by sampling the rows of $\boldsymbol{Y}_{\text{pwr}}$ on a logarithmic scale, we convert differences in tempo to differences in offsets. We implement this by representing each row of $\boldsymbol{Y}_{\text{pwr}}$ by the output of a number of exponentially spaced bands. In Fig. 4, the 25 bands with 50% overlap that we used are shown. The lowest band start at 0.017 Hz, and the highest band end at 0.667 Hz, thus capturing variations on a time scale between 1.5 s and 60 s. The amount of temporal scaling allowed is further increased when computing the distance. The resulting feature is a 12×25 matrix where component i, j reflects the amount of modulation of semitone i in frequency band j. In comparison, if a song is 4 minutes long and has a tempo of 120 beats per minute, the beat-synchronous feature in [5] will have a dimension of 12×480.

3. DISTANCE MEASURE

We compute the distance between two feature matrices $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ by normalizing them to unit norm and compute the minimum Frobenius distance when allowing transpositions and frequency shifts. First, we normalize to unit Frobenius norm:

$$\boldsymbol{X}'_1 = \boldsymbol{X}_1 / \|\boldsymbol{X}_1\|_{\text{F}}, \tag{3}$$

$$\boldsymbol{X}'_2 = \boldsymbol{X}_2 / \|\boldsymbol{X}_2\|_{\text{F}}. \tag{4}$$

Let $\mathbf{T}_{12}$ be the 12×12 permutation matrix that transposes $\boldsymbol{X}'_1$ or $\boldsymbol{X}'_2$ by one semitone:

$$(\mathbf{T}_{12})_{i,j} = \begin{cases} (\boldsymbol{I})_{i+1,j} & \text{for } i < 12, \\ (\boldsymbol{I})_{1,j} & \text{for } i = 12, \end{cases} \tag{5}$$

where $\boldsymbol{I}$ is the identity matrix. To compensate for transpositions, we minimize the Frobenius distance over all possible transpositions:

$$d'(\boldsymbol{X}'_1, \boldsymbol{X}'_2) = \min_{p \in \{1,2,\cdots,12\}} \|\mathbf{T}_{12}^p \boldsymbol{X}'_1 - \boldsymbol{X}'_2\|_{\text{F}}. \tag{6}$$

To allow even further time scaling than permitted by the effective bandwidths, we also allow shifting the matrices by up to two columns:

$$d(\boldsymbol{X}'_1, \boldsymbol{X}'_2) = \min_{s \in \{-2,-1,0,1,2\}} d'(\boldsymbol{X}'^{(s)}_1, \boldsymbol{X}'^{(-s)}_2), \tag{7}$$

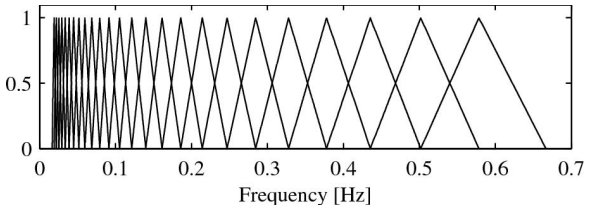


Fig. 4. Bandwidths of the 25 logarithmically spaced filters.

where

$$\boldsymbol{X}_l'^{(s)} = \begin{cases} [\, \mathbf{0}_s \quad \boldsymbol{X}_l' \,] & \text{if } s \geq 0, \\ [\, \boldsymbol{X}_l' \quad \mathbf{0}_{-s} \,] & \text{if } s < 0, \end{cases} \tag{8}$$

and where $\mathbf{0}_s$ is a $12 \times s$ matrix of zeros. Since the distance measure is based on the Frobenius norm, it obeys the triangle inequality.

4. EVALUATION

We have evaluated the distance measure by using a nearest neighbor classifier on two different datasets, namely a set of synthesized MIDI files [12] and the covers80 set [13]. Furthermore, the algorithm was evaluated as part of the MIREX 2007 cover song identification task [14].

The basic set of MIDI files consists of 900 MIDI songs that are 30 different melodies of length 180 seconds played with 30 different instruments. To measure the sensitivity to transpositions and variations in tempo, queries that are transposed and lengthened/shortened are used. For each query, the nearest neighbor is found, and the fraction of nearest neighbor songs that share the same melody is counted. In Fig. 5 the effect of transpositions is shown, and in Fig. 6 the effect of changing the tempo is shown. It is seen that transposing songs hardly affects performance, and that changing the tempo between a factor 0.7 and 1.4 also does not affect performance too seriously.

The covers80 dataset consists of 80 titles each in two different versions, i.e., a total of 160 songs. The vast majority of the titles have been recorded by two different artists, although a few consist of a live version and a studio version by the same artist. The 160 songs are split into two sets with one version of each song in each set. When evaluating the cover song detection system, the nearest neighbor in the second set to a query from the first set is assumed to be the cover. With this setup, the cover version was found in 38% of the cases. However, as parameters have been tweaked using this dataset, some degree of overtraining is inevitable. In the following, by rank of a cover song we mean rank of the cover when all songs are sorted by their distance to the query. A rank of one means the nearest neighbor to the query song is its cover version, while a rank of e.g. 13 means there are 12 other songs that are considered closer than the real cover by the system. In Fig. 7, a histogram of the ranks of all the covers is shown. A closer inspection of the data reveals that 66% of the cover songs are within the 10 nearest neighbors. In Table 1, the songs with the highest ranks are listed. For most of these, the two versions are very different, although a few, such as "Summertime Blues", are actually quite similar. Nevertheless, improving on the heavy tail is probably not possible without taking lyrics into account.

Comparing different music information retrieval algorithms has long been impractical, as copyright issues have prevented the development of standard music collections. The annual MIREX evaluations overcome this problem by having participants submit their algorithms which are then centrally evaluated. This way, distribution of song data is avoided. We submitted the proposed system to the MIREX 2007 audio cover song identification task. The test set is closed and consists of 30 songs each in 11 versions and 670 unrelated songs used as noise. Each of the 330 cover songs are in turn used as query. Results of the evaluation are shown in Table 2, where it is seen that the proposed system came in fourth. Interestingly, it has almost the exact same performance as the 2006 winner.

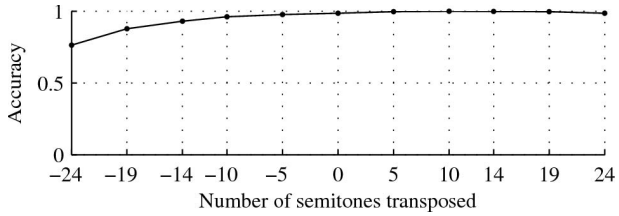


Fig. 5. Effect of transpositions on melody recognition accuracy.

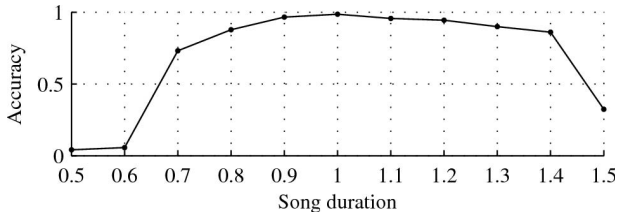


Fig. 6. Effect of lengthening or shortening a song on melody recognition accuracy. The duration is relative to the original song.

5. CONCLUSION

We have presented a low complexity cover song identification system with moderate storage requirements and with comparable performance to the cover song identification algorithm that performed best at the MIREX 2006 evaluation. Since the proposed distance measure obeys the triangle inequality, it might be useful in large-scale databases. However, further studies are needed to determine whether the intrinsic dimensionality of the feature space is too high to utilize this in practice.

6. ACKNOWLEDGEMENTS

The authors would like to thank the IMIRSEL team for organizing and running the MIREX evaluations.

7. REFERENCES

[1] B. Logan and A. Salomon, "A music similarity function based on signal analysis," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2001, pp. 745 – 748.

[2] E. Pampalk, "Computational models of music similarity and their application to music information retrieval," Ph.D. dissertation, Vienna University of Technology, Austria, 2006.

[3] J.-J. Aucouturier, "Ten experiments on the modelling of polyphonic timbre," Ph.D. dissertation, University of Paris 6, France, 2006.

[4] J. H. Jensen, M. G. Christensen, M. N. Murthi, and S. H. Jensen, "Evaluation of MFCC estimation techniques for music similarity," in *Proc. European Signal Processing Conf.*, 2006.

[5] 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.

[6] W.-H. Tsai, H.-M. Yu, and H.-M. Wang, "A query-by-example technique for retrieving cover versions of popular songs with similar melodies," in *Proc. Int. Symp. on Music Information Retrieval*, 2005, pp. 183–190.

[7] K. Lee, "Identifying cover songs from audio using harmonic representation," in *Music Information Retrieval Evaluation eXchange*, 2006.

[8] J. P. Bello, "Audio-based cover song retrieval using approximate chord sequences: Testing shifts, gaps, swaps and beats," in *Proc. Int. Symp. on Music Information Retrieval*, 2007, pp. 239–244.

[9] T. Lidy and A. Rauber, "Combined fluctuation features for music genre classification," in *Music Information Retrieval Evaluation eXchange*, 2005.

[10] M. A. Bartsch and G. H. Wakefield, "To catch a chorus: using chroma-based representations for audio thumbnailing," in *Proc. IEEE Workshop on Appl. of Signal Process. to Aud. and Acoust.*, 2001, pp. 15 – 18.

[11] S. Saito, H. Kameoka, T. Nishimoto, and S. Sagayama, "Specmurt analysis of multi-pitch music signals with adaptive estimation of common harmonic structure," in *Proc. Int. Symp. on Music Information Retrieval*, 2005, pp. 84–91.

[12] J. H. Jensen, M. G. Christensen, and S. H. Jensen, "A framework for analysis of music similarity measures," in *Proc. European Signal Processing Conf.*, 2007, pp. 926–930.

[13] D. P. Ellis. (2007) The "covers80" cover song data set. [Online]. Available: http://labrosa.ee.columbia.edu/projects/coversongs/covers80/

[14] J. S. Downie, K. West, D. Ellis, and J. Serrà. (2007) MIREX audio 2007 cover song identification.

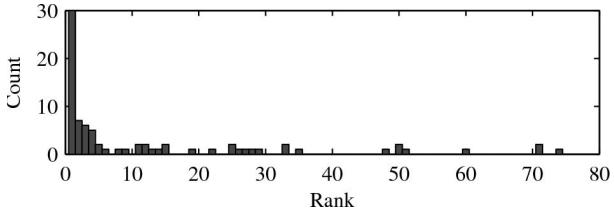


Fig. 7. Histogram of the cover song ranks.

Title	Artists	Rank
My Heart Will Go On	Dion/New Found Glory	74
Summertime Blues	A. Jackson/Beach Boys	71
Yesterday	Beatles/En Vogue	71
Enjoy The Silence	Dep. Mode/T. Amos	60
I Can't Get No Satisfact.	B. Spears/R. Stones	51
Take Me To The River	Al Green/Talking Heads	50
Wish You Were Here	Pink Floyd/Wyclef Jean	50
Street Fighting Man	RATM/R. Stones	48
Tomorrow Never Knows	Beatles/Phil Collins	35
I'm Not In Love	10cc/Tori Amos	33
Red Red Wine	Neil Diamond/UB40	33

Table 1. Titles of songs with rank > 30.

Rank	Participant	Avg. prec.	Covers in top 10
1	Serrà & Gómez	0.521	1653
2	Ellis & Cotton	0.330	1207
3	Bello, J.	0.267	869
4	*Jensen, Ellis, Christensen & Jensen*	*0.238*	*762*
5	Lee, K. (1)	0.130	425
6	Lee, K. (2)	0.086	291
7	Kim & Perelstein	0.061	190
8	IMIRSEL	0.017	34

Table 2. MIREX 2007 Audio Cover Song Identification results. In comparison, the 2006 winner [5] identified 761 cover songs in top 10.