Model-based Cover Song Detection via Threshold Autoregressive Forecasts

Joan Serrà[1,2] *, Holger Kantz[2], Ralph G. Andrzejak[1]

[1] Dept. of Inf. and Comm. Technology, Universitat Pompeu Fabra, Roc Boronat 138, 08018 Barcelona, Spain

[2] Max Planck Institute for the Physics of Complex Systems, Nöthnitzer Strasse 38, 01187 Dresden, Germany

Oct 25, 2010

Abstract

Current systems for cover song detection are based on a model-free approach: they basically search for similarities in descriptor time series reflecting the evolution of tonal information in a musical piece. In this contribution we propose the use of a model-based approach. In particular, we explore threshold autoregressive models and the concept of cross-prediction error, i.e. a measure of to which extent a model trained on one song's descriptor time series is able to predict the covers'. Results indicate that the considered approach can provide competitive accuracies while being considerably fast and with potentially less storage requirements. Furthermore, the approach is parameter-free from the user's perspective, what provides a robust and straightforward application of it.

1 Introduction

Cover songs are alternative renditions of the same underlying musical piece. The automatic detection of cover songs based on the audio content has been a very active area of study within the music information retrieval (MIR) community over the last years [11]. This is clearly due to the revolution which has lashed this field, intrinsically related to the introduction of digital ways to share and distribute information, which challenges the search and organization of musical contents [2, 3].

Cover song detection is a very simple task from a user's perspective: a query song is provided and the system is asked to retrieve all versions of it that are found in a given music collection. However, from an MIR perspective it becomes a very challenging task, since cover songs might differ from their originals in several musical aspects such as timbre, tempo, song structure, key, arrangement, lyrics, or language of the vocals. In spite of these differences, cover songs might retain a considerable part of their tonal evolution (or harmonic progression). Indeed, the big majority of current approaches are based on the detection of common patterns in time series of tonal descriptors [11].

Another major characteristic that is shared among state-of-the-art approaches for cover song detection is the lack of specific modeling strategies for tonal descriptor time series. In the present contribution we proceed in this direction by introducing a model-based system for cover song detection. In particular, we study a model-based forecasting approach, where we employ the concept of cross-prediction error. Intuitively, once a model has learned the patterns found in the time series of a given query song, one should expect the average prediction error to be relatively small when the time series of a candidate cover song is used as input. Otherwise, i.e. when an unrelated (non-cover) candidate song is considered, the prediction error should be higher.

Our approach consists of training a threshold autoregressive (TAR) model [13] to learn the characteristics of a query song's descriptor time series, and then assessing the predictions of the model when a target time series of a candidate song is considered. We show that the approach is very promising in the sense that it achieves competitive accuracies and furthermore provides additional advantages when compared to state-of-the-art approaches, such as lower computational complexities and potentially less storage requirements. Perhaps the most interesting aspect of the proposed approach is that no parameters need to be adjusted. More concretely, models' parameters and coefficients are automatically learned for each song and descriptor time series individually (no intervention of the user is needed). Accordingly, the system can be readily applied to different music collections or descriptor time series.

The paper is structured as follows. First, an overview of the employed time series of tonal descriptors is done (Sec. 2). Then our approach for cover song detection is presented (Sec. 3). Some details about our evaluation follow (Sec. 4). We subsequently present our results (Sec. 5) and discuss our approach (Sec. 6). We end with some short conclusions and provide an outlook for future work (Sec. 7).

*This is a preliminary author's version of the article. For the final version please check the digital library of the ACM.

2 Descriptor time series

We experiment with three descriptor time series reflecting the evolution of the tonal information of a musical piece. These are extracted from the raw audio signal using a frame length of 116.1 ms and a hop size of 104.5 ms. The extraction process results in a multidimensional descriptor time series, which we denote as a matrix $\mathcal{S} = [\mathbf{s}_1 \dots \mathbf{s}_N]$, where N is the total number of samples (frames) and $\mathbf{s}_n$ is a column vector with D components representing a D-dimensional descriptor at sample n. Therefore, element $s_{d,n}$ of $\mathcal{S}$ represents the magnitude of the d-th descriptor component of the n-th frame. The descriptors we use are:

1) Pitch class profiles (PCP): PCP features [4] are derived from the frequency dependent energy in a given range of the frame's spectrum (typically from 50 to 5000 Hz). This energy is usually mapped into an octave-independent histogram representing the relative intensity of each of the 12 semitones of the equal-tempered chromatic scale. Important PCP characteristics include [4]: robustness against non-tonal components, independence of timbre and the specific instruments used, and independence of a musical piece's loudness and volume fluctuations. We here use *harmonic* PCPs [4] which, apart from above properties, reduce the influence of noisy spectral components, take into account the presence of harmonic frequencies, and are tuning independent.

2) Tonal centroid (TC): PCP features are mapped to the interior space of a 6-dimensional polytope, where perceptually close harmonic relations appear as small Euclidean distances [5]. This mapping is obtained by multiplying each PCP vector by a suitable transformation matrix and then normalizing by the L_1 norm of the former.

3) Harmonic change (HC): The harmonic change detection function [5] is simply defined as the Euclidean distance between pairs of consecutive TC samples.

3 Model-based cover song detection

3.1 State space reconstruction

Since an isolated sample $\mathbf{s}_n$ might not contain the necessary information for a reliable prediction at some future time step h, one might consider information from past samples. As a notational representation of the present and recent past of a time series we use the concept of delay coordinate state space embedding, a tool that is routinely employed in nonlinear time series analysis [6]. Noticeably, there is evidence that nonlinear time series analysis tools can be beneficial for music retrieval systems (see our previous work [12] and references therein).

In our case, for multidimensional samples $\mathbf{s}_n$, we construct delay coordinate state space vectors $\mathbf{s}^*_n$ through vector concatenation, i.e.

$$\mathbf{s}^*_n = \left(\mathbf{s}_n^T \;\; \mathbf{s}_{n-\tau}^T \;\; \cdots \;\; \mathbf{s}_{n-(m-1)\tau}^T\right)^T, \qquad (1)$$

where superscript ^{T} denotes vector transposition, m is the embedding dimension, and τ is the time delay [6]. The sequence of these reconstructed samples yields again a multidimensional time series $\mathcal{S}^* = \left[\mathbf{s}^*_{w+1} \dots \mathbf{s}^*_N\right]$, where $w = (m-1)\tau$ corresponds to the so-called embedding window. Notice that Eq. (1) still allows for the use of the raw time series samples (i.e. if $m = 1$ then $\mathcal{S}^* = \mathcal{S}$).

3.2 Autoregressive (AR) models

A widespread way to model linear time series data is through an AR process, where predictions are based on a linear combination of m previous measurements [1]. We here employ a multivariate AR model [7] and the previous state space representation [Eq. (1)]. In particular, we first construct delay coordinate state space vectors $\mathbf{s}^*_n$ and then express the forecast $\hat{\mathbf{s}}_{n+h}$ at h steps ahead from the n-th sample $\mathbf{s}_n$ as

$$\hat{\mathbf{s}}_{n+h} = \mathcal{A}\, \mathbf{s}^*_n, \qquad (2)$$

where $\mathcal{A}$ is the $D \times mD$ coefficient matrix of the multivariate AR model. By considering samples $n = w + 1, \dots N - h$, one obtains an overdetermined system

$$\hat{\mathcal{S}} = \mathcal{A}\, \mathcal{S}^* \qquad (3)$$

which, by ordinary least squares fitting [10], allows to estimate $\mathcal{A}$.

3.3 Threshold autoregressive (TAR) models

TAR models generalize AR models by introducing nonlinearity [13]. A single TAR model consists of a collection of AR models where each single one is valid only for certain time series samples, which are grouped according to their similarities (piecewise linearization [6]). For determining all TAR coefficients we group the samples of $\mathcal{S}^*$ into K clusters with a K-medoids algorithm[1] [9] and determine, independently for each partition, AR coefficients as above [Eqs. (2,3)]. Importantly, each AR model is associated to the corresponding cluster medoid. When forecasting, we again construct delay coordinate state space vectors $\mathbf{s}^*_n$ from each input sample $\mathbf{s}_n$ and calculate their Euclidean distance to all $k = 1, \dots K$ medoids. The forecast at horizon h is then

$$\hat{\mathbf{s}}_{n+h} = \mathcal{A}^{(k')}\, \mathbf{s}^*_n, \qquad (4)$$

where $\mathcal{A}^{(k')}$ is the $D \times mD$ coefficient matrix of the multivariate AR model associated to the cluster medoid closest to $\mathbf{s}^*_n$, being k' the index of this medoid.

[1] We re-implement it from the cited reference without further modifications.

3.4 Training and testing

TAR models are completely described by a series of parameters m (embedding dimension), τ (time delay), and K (number of clusters), and a series of coefficients $\mathcal{A}^{(k)}$, $k = 1, \dots K$. In our experiments these values are learned independently for each song and descriptor using the entire time series as training set. This learning is done in an unsupervised way, with no prior information about parameters and coefficients. More concretely, for each song and descriptor time series we calculate the corresponding model coefficients for different parameter configurations and then select the solution that leads to the best in-sample approximation of the data. We perform a grid search over $m \in$ [1, 2, 3, 5, 7, 9, 12, 15], $\tau \in$ [1, 2, 6, 9, 15], and $K \in$ [1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 15, 20, 30, 40, 50]. Intuitively, with such a search for the best parameter combination for a concrete song's time series, part of the time series modeling is also done through the appropriate parameter setting, since m, τ, and K are parameters that also define time series' characteristics [6]. Notice that the prediction horizon h cannot be optimized in-sample since best approximations would always correspond to $h = 1$ due to inherent sample correlations. The impact of h can only be assessed on the out-of-sample prediction, when the model is applied to the candidate song.

Since we aim at obtaining compact descriptions of our time series and we want to avoid overfitting, we limit the total number of model parameters and coefficients to be less than 10% of the number of values of the time series data. This implies that parameter combinations leading to models with more than $(N \times D)/10$ values are automatically discarded at the training phase. We also limit the embedding window to $w < N/20$.

Once a TAR model is trained on a descriptor time series for a given query song i, we transpose the time series of a candidate song j by the optimal transposition index method [11] in order to match the key of the query song. Once this preliminary step is done, we perform an out-of-sample prediction with the i-th song model using the j-th song time series both as input and target.

To evaluate prediction accuracy we use a normalized mean squared error measure, both when training our models (to select the best parameter combination) and when forecasting. We define

$$e = \frac{1}{N-h-w} \sum_{n=w+1}^{N-h} \frac{1}{D} \sum_{d=1}^{D} \frac{\left(\hat{s}_{d,n+h} - s_{d,n+h}\right)^2}{\sigma_d}, \quad (5)$$

where σ_d is the variance of the d-th descriptor component over all samples $n = w + h + 1, \dots N$ of the target time series. We use the notation $e_{i,j}$ when a model trained on song i is used to forecast song j.

4 Evaluation

We use an in-house music collection consisting of 2125 cover songs. This music collection spans a variety of genres and styles and is an extension of the one used in our previous work [12]. It includes 523 groups of covers and the average number of songs per group is 4.06, ranging from 2 to 18.

To evaluate the accuracy in detecting cover songs we proceed as follows. Given a music collection with Q songs, we calculate $e_{i,j}$ for all $Q \times Q$ possible pairwise combinations and then create a symmetric dissimilarity matrix $\mathcal{D}$, whose elements are $d_{i,j} = e_{i,j} + e_{j,i}$. Once $\mathcal{D}$ is computed, we can resort to standard information retrieval (IR) measures to evaluate the discriminative power of this information. In particular we use the mean of average precisions (MAP) measure [8]. This measure, which ranges between 0 and 1, is routinely employed in the IR [8] and MIR [3] communities, and specially in evaluating cover song detection systems [11]. A baseline MAP across 99 iterations of a random matrix $\widetilde{\mathcal{D}}$ is computed for additional assessment of our results.

5 Results

In preliminary trials we saw that the prediction horizon h had an important impact in system's performance, so we decided to study the accuracy for different h values with a reduced set of 102 arbitrarily selected cover songs (17 groups of 6 versions). The results are shown in Table 1. We see that accuracies increase with h until they reach a more or less stable plateau for $h \geq 7$ (more than 731 ms). We hypothesize that this intriguing behavior is due to strong correlations between subsequent time series samples which, at short time intervals h, do not allow for the learning of relevant patterns that can characterize a song (and thus that could be useful for detecting its covers). Recall that we are using a hop size of 104.5 ms and that PCP, TC, and HC values might not change dramatically in such a short time interval [4]. However, a more thorough understanding of this phenomenon requires further research.

The final MAP achieved with the full collection ($h = 19$) is 0.386 for the PCP descriptor, 0.441 for the TC descriptor and 0.064 for the HC descriptor. We see that the HC descriptor is much less powerful than the other two. This is to be expected, since HC compresses tonal information to a univariate value. Furthermore, tonal change might be less informative than tonal values themselves, which already contain the change information in their temporal evolution. However, the HC MAP is still higher than the random baseline, which is 0.006. Apart from this, we see that TC performs better than PCP. This does not necessarily imply that TC descriptors provide a better representation of a song's tonal information (actually they are directly derived from PCPs), but that TAR models might better capture the essence of

Table 1: System's MAP in dependence of the prediction horizon h for the three descriptors tested. Results computed for a reduced set of 102 songs (see text). The random baseline MAP for 102 songs is 0.085.

Descriptor	Prediction horizon h									
	1	4	7	10	13	16	19	22	26	30
PCP	0.292	0.554	0.588	0.592	0.604	0.616	0.623	0.623	0.610	0.607
TC	0.279	0.633	0.694	0.688	0.690	0.692	0.703	0.666	0.658	0.648
HC	0.194	0.314	0.293	0.318	0.293	0.267	0.280	0.280	0.254	0.279

their temporal evolution. Noticeably, the combination of $K = 1$ and $m = 1$, what would correspond to a simple AR model with no state space reconstruction, was never selected in our experiments. Further trials with $K = 1$ and/or $m = 1$ yielded worse results.

6 Discussion

It might seem that a MAP around 0.4 is not a big success for a cover song detection approach. To properly asses this success one has to compare with the performance of current systems. According to MIREX [3], the best accuracy achieved to date within the cover song detection task corresponds to our previous work [12]. This system reached a MAP of 0.66 with the MIREX dataset and a MAP of 0.698 with the music collection used here. Thus the current approach does not outperform [12]. However, one should notice that MAP values around 0.4 are in line with other state-of-the-art accuracies, or even better [11].

Beyond accuracy comparisons, some other aspects can be discussed. Indeed, another reason for appreciating the solution obtained here comes from the consideration of storage capabilities and computational complexities at the retrieval stage. Since we limit our models to a size of 10% of the total number of training data, they require 10% of the storage that would be needed for saving the entire time series (state-of-the-art systems usually store the full time series for each song). This fact could be exploited in a retrieval stage, although for doing so one might have to get rid of the symmetrization of cross-prediction errors $\mathcal{D}$ and use the elements $e_{i,j}$ directly. Regarding computational complexity, many approaches for cover song detection (including our previous method [12]) are quadratic in the length of the time series, requiring at least an Euclidean distance calculation for every pair of sample points. In contrast, the approach presented here is linear in the length of the time series: we just need to do a pairwise distance calculation between samples and the K medoids, plus a matrix multiplication and subtraction. More concretely, if we compare our previous approach [12] with the TAR-based strategy by considering an average time series length $\bar{N}$, we have that the former is roughly $O\left(\bar{N}^2 mD\right)$, while the latter is $O\left(\bar{N} mD(K + D)\right)$, being $K + D \ll \bar{N}$. To put some numbers: with $\bar{N} = 2304$ (approximately 4 min of music), descriptor dimensionality $D = 12$ (the largest one among PCP, TC, and HC), and $K = 50$ (the maximum allowed), we obtain a minimal relative speed improvement of $2304/(50 + 12) \approx 37$ times.

A further and very interesting advantage of the approach considered here is that it does not need any parameter optimization by the user, therefore making its application robust and straightforward. Usually, cover song detection systems have multiple parameters that can be dependent, for instance, on the music collection, the music descriptor time series, or the types of cover songs under consideration [11]. Our previous method [12] was not an exception: as we did not have a way to a priori set its specific parameters, these were set by trial and error with an independent out-of-sample music collection. With the TAR-based approach, the best parameter configuration is automatically found for each song and descriptor time series by the minimization of the in-sample training error $e_{i,i}$.

7 Conclusions and future work

We see that considering cross-predictions of TAR models leads to a parameter-free approach for cover song detection. Furthermore, the approach is fast, allows for reduced storage, and still maintains a highly competitive accuracy when compared to state-of-the-art systems. Thus, time series modeling strategies stand as a really promising approach for cover song detection and, by extension, for music and multimedia retrieval in general [2, 3].

Future research will be devoted to the application of other common time series models to the cover song detection task. Moreover, we will focus on how the forecasts of these models behave as a function of the prediction horizon. This will allow us to study the behavior of music descriptor time series from a wider perspective. Finally, we will try to improve model-based approaches by introducing specific modifications for the cover song detection task [11] (e.g. taking into account tempo or structural changes between cover songs).

8 Acknowledgments

We thank Emilia Gómez for her review on a previous version of this article. J.S. has been partially funded by the A/09/96235 grant from the *Deutscher Akademischer*

Austausch Dienst and by the Music 3.0 (TSI-070100-2008-318) and Buscamedia (CEN-20091026) projects. R.G.A. has been funded by the BFU2007-61710 grant of the Spanish Ministry of Education and Science.

References

[1] G. Box and G. Jenkins. *Time series analysis: forecasting and control*. Holden-Day, rev. edition, 1976.

[2] M. Casey, R. C. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney. Content-based music information retrieval: current directions and future challenges. *Proceedings of the IEEE*, 96(4):668–696, April 2008.

[3] J. S. Downie. The music information retrieval evaluation exchange (2005–2007): a window into music information retrieval research. *Acoustical Science and Technology*, 29(4):247–255, 2008.

[4] E. Gómez. *Tonal description of music audio signals*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2006. Available online: `http://mtg.upf.edu/node/472`.

[5] C. Harte, M. B. Sandler, and M. Gasser. Detecting harmonic change in musical audio. *ACM Workshop on Audio and Music Computing Multimedia*, pages 21–26, 2006.

[6] H. Kantz and T. Schreiber. *Nonlinear time series analysis*. Cambridge University Press, 2nd edition, 2004.

[7] H. Lütkepohl. *Introduction to multiple time series analysis*. Springer, 2nd edition, 1993.

[8] C. D. Manning, R. Prabhakar, and H. Schutze. *An introduction to information retrieval*. Cambridge University Press, 2008.

[9] H. S. Parka and C. S. Jun. A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications*, 36(2):3336–3341, March 2009.

[10] W. H. Press, B. P. Flannery, S. A. Tenkolsky, and W. T. Vetterling. *Numerical recipes*. Cambridge University Press, 2nd edition, 1992.

[11] J. Serrà, E. Gómez, and P. Herrera. *Audio cover song identification and similarity: background, approaches, evaluation, and beyond*, volume 16 of *Studies in Computational Intelligence*, chapter 14, pages 307–332. Springer-Verlag, March 2010.

[12] J. Serrà, X. Serra, and R. G. Andrzejak. Cross recurrence quantification for cover song identification. *New Journal of Physics*, 11:093017, September 2009.

[13] H. Tong and K. S. Lim. Threshold autoregression, limit cycles and cyclical data. *Journal of the Royal Statistical Society*, 42(3):245–292, 1980.