

Unsupervised Detection of Music Boundaries by Time Series Structure Features

Joan Serra¹, Meinard Müller², Peter Grosche², and Josep Ll. Arcos¹

¹ Artificial Intelligence Research Institute (IIIA-CSIC), Bellaterra, Barcelona, Spain

² Max Planck Institute for Computer Science and Saarland University, Saarbrücken, Germany
 {jserra,arcos}@iiia.csic.es, {meinard,pgrosche}@mpi-inf.mpg.de

Abstract

Locating boundaries between coherent and/or repetitive segments of a time series is a challenging problem pervading many scientific domains. In this paper we propose an unsupervised method for boundary detection, combining three basic principles: novelty, homogeneity, and repetition. In particular, the method uses what we call structure features, a representation encapsulating both local and global properties of a time series. We demonstrate the usefulness of our approach in detecting music structure boundaries, a task that has received much attention in recent years and for which exist several benchmark datasets and publicly available annotations. We find our method to significantly outperform the best accuracies published so far. Importantly, our boundary approach is generic, thus being applicable to a wide range of time series beyond the music and audio domains.

Introduction

Time series data are ubiquitous and pose challenges to many scientific domains, including artificial intelligence (Keogh 2011). Research on time series has a long tradition, but its application to real-world datasets requires to cope with new relevant issues, such as the multiple dimensionality of data or limited computational resources. Specifically, dealing with large-scale data, (1) algorithms must be efficient, i.e. they have to scale, (2) supervised approaches may become unfeasible, and (3) solutions must use general techniques, i.e. they should be as independent of the domain as possible (see Mueen and Keogh 2010 for a more detailed discussion).

A challenge crossing diverse scientific domains is the segmentation of time series into meaningful, coherent units by automatically detecting their boundaries or transition points. Following Paulus et al. (2010), one can group existing approaches into three main categories: repetition-based, novelty-based, and homogeneity-based. Repetition-based approaches focus on identifying recurrent patterns, called motifs, like in Vahdatpour et al. 2009, where an unsupervised method based on graph clustering is proposed to cope with multi-dimensional time series. Novelty-based

approaches usually go along with some change of the signal's property, trying to detect local transitions between contrasting parts. In these approaches, the challenge is the design of domain-independent mechanisms without a predefined knowledge of the segments, such as in Armstrong and Oates 2007. Finally, homogeneity-based approaches exploit stationarities in the time series, trying to detect consistencies with respect to some property. This strategy was partially followed in Firoiu and Cohen 2002, where homogeneous regions (or "patterns") were detected by means of artificial neural networks. Transitions between regions were then modeled by hidden Markov models. Notice that novelty-based and homogeneity-based approaches are two sides of a coin: novelty detection is based on observing some surprising event or change after a more homogeneous period.

One of the domains with an intensive research on detecting time series boundaries is music information retrieval (Casey et al. 2008; Müller et al. 2011), where lots of effort have been devoted to the automatic identification of musical structure and their boundaries (see Paulus et al. 2010 for a survey). In music, boundaries may occur because of multiple changes, such as a change in instrumentation, a change in harmony, or a change in tempo. The seminal approach by Foote (2000) estimated these changes by means of a so-called novelty curve, obtained by sliding a short-time checkerboard kernel over the diagonal of a self-similarity matrix of pairwise sample comparisons. Works inspired by Foote's approach explicitly make use of the concept of novelty curves (Paulus et al. 2010). Other music-targeted approaches exploit homogeneities in a time series by employing more refined techniques like hidden Markov models (Levy and Sandler 2008) or dynamic texture mixtures (Barrington et al. 2010). Moreover, since repetitions play a crucial role in music, boundaries may not solely be determined by a change with regards to some acoustic property, but also by some structural property that refers to the entire signal. Thus, there are many approaches to music segmentation based on repetitions (Paulus et al. 2010). However, few studies combine different segmentation principles like homogeneity and repetition as done, for instance, by Paulus and Klapuri (2009).

In this paper we propose a new method for unsupervised boundary detection in multi-dimensional time series. The main idea is to combine the homogeneity and repetition prin-

ciples in a single representation we refer to as *structure feature*. Structure features are computed for every time series sample, but encode the structural relation of this sample to all other samples in the time series. Since we deal with a local phenomenon (a boundary) which may depend on some global phenomenon (the repetitive structure of the time series), a novelty curve is then computed from the sequence of structure features. Thus, we obtain a novelty curve that reflects both local changes and global characteristics¹, specially including beginnings and endings of repeated and/or homogeneous segments. Notably, our approach and the construction of structure features is independent of the type of time series used and, therefore, it is not restricted to music-derived signals.

The remainder of the paper is organized as follows. First, we describe our method. Second, we summarize the evaluation methodology followed. Third, we report on the results obtained. Fourth, a brief conclusion closes the paper.

Method

Our method consists of four main steps: (1) Emulate the capacity of short-time human memory by encapsulating, in a new sample, the most recent past of a time series sample. (2) Assess homogeneities and repetitions found in a time series by pairwise comparison of the previously encapsulated samples. (3) Construct the structure features of a time series. Structure features are obtained by considering temporal lag information and by estimating a bivariate probability density with Gaussian kernels. (4) Compute differences between consecutive structure features to simulate boundary perception. This final step yields a novelty curve whose peak positions indicate boundary estimates. We now detail these four steps.

Accounting for the recent past

Let there be a time series $X = [\mathbf{x}_1, \dots, \mathbf{x}_{N'}]$, of length N' , with potentially multi-dimensional samples $\mathbf{x}_i$ (column vectors). To emulate short-time memory (Baddeley 2003), information of the most recent past is incorporated to every sample $\mathbf{x}_i$. This can be easy and elegantly done by using delay coordinates, a technique routinely employed in nonlinear time series analysis (Kantz and Schreiber 2004). New samples are constructed by vector concatenation as

$$\hat{\mathbf{x}}_i = [\mathbf{x}_i^T \quad \mathbf{x}_{i-\tau}^T \quad \dots \quad \mathbf{x}_{i-(m-1)\tau}^T]^T, \quad (1)$$

where T denotes vector transposition, m is the so-called embedding dimension, and τ is a time delay. Although there are recipes to estimate the optimal values of m and τ from the information contained in a time series X , we leave them as parameters (see below). Note that the value of m indicates the amount of past information being considered for the task which, in general, ranges a time span of $w = (m-1)\tau$. In preliminary analysis we found that considering past information ($m > 1$) provided more stability to the method and substantial increases in accuracy (see also the results section). By applying Eq. 1 for $i = w+1, \dots, N'$ we obtain

¹This point largely differs from Foote's approach.

a multi-dimensional time series $\hat{X} = [\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_N]$ of length $N = N' - w$.

Homogeneities and recurrences

The next step consists in assessing homogeneities and recurrences. For that we resort to a recurrence plot (Marwan et al. 2007), which consists of a square matrix R whose elements $R_{i,j}$ indicate pairwise resemblance between samples at times i and j (Fig. 1, third row). Formally, for $i, j = 1, \dots, N$,

$$R_{i,j} = \Theta(\varepsilon_{i,j} - \|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|), \quad (2)$$

where $\Theta(z)$ is the Heaviside step function, yielding 1 if $z > 0$ and 0 otherwise, $\|\cdot\|$ can be any norm (we use the Euclidean norm), and ε is a suitable threshold.

As done by Serrà et al. (2009), we set a different threshold $\varepsilon_{i,j}$ for each cell (i, j) . Specifically, our procedure for building R can be outlined as follows. First, for each sample $\hat{\mathbf{x}}_i$, $i = 1, \dots, N$, we search for its K nearest neighbors. Then, neighbor mutuality is forced by setting $R_{i,j} = 1$ only if $\hat{\mathbf{x}}_i$ is a neighbor of $\hat{\mathbf{x}}_j$ and, at the same time, $\hat{\mathbf{x}}_j$ is a neighbor of $\hat{\mathbf{x}}_i$. In our experience with recurrence plots we found this restrictive strategy to be more robust against noise than the variants outlined by Marwan et al. (2007). To account for time series of different lengths, we set $K = \kappa N$, i.e. we set the number of nearest neighbors to a fraction $\kappa \in [0, 1]$ of the length of the time series being considered.

Structure features

The subsequent steps involve the creation of structure features (SF). We first represent the homogeneities and recurrences of R in a circular time-lag matrix L (Fig. 1, third row, left). Such process is similar to the typical process of constructing a time-lag matrix (Goto 2006) but incorporates the information of future, as well as past time lags. We do it by circularly shifting the rows of R such that

$$L_{i,j} = R_{i,k+1} \quad (3)$$

for $i, j = 1, \dots, N$, where k equals to $i + j - 2$ modulo N .

The circular time matrix L can then be considered as a sample from a bivariate distribution $\bar{P}$ along the time and lag axes (i and j axes, respectively). This bivariate distribution would correspond to a probability mass function of time-lag recurrences². The estimate P such an underlying distribution $\bar{P}$ is obtained using bivariate kernel density estimation (Simonoff 1996), a fundamental data smoothing concept where inferences about a population are made based on a finite sample of it. In our case, P is estimated by convolving L with a bivariate rectangular Gaussian kernel G :

$$P = L * G. \quad (4)$$

The kernel G is obtained by multiplying two Gaussian windows $\mathbf{g}_t$ and $\mathbf{g}_l$ of variance σ^2 , with sizes s_t and s_l , corresponding to the time and lag dimensions of L , respectively. This way, G has s_t rows and s_l columns:

$$G = \mathbf{g}_t \mathbf{g}_l^T. \quad (5)$$

²Actually, it is not needed that the values of $\bar{P}$ sum to 1, since such normalization only introduces a scale parameter that is eliminated in a subsequent operation (see below).

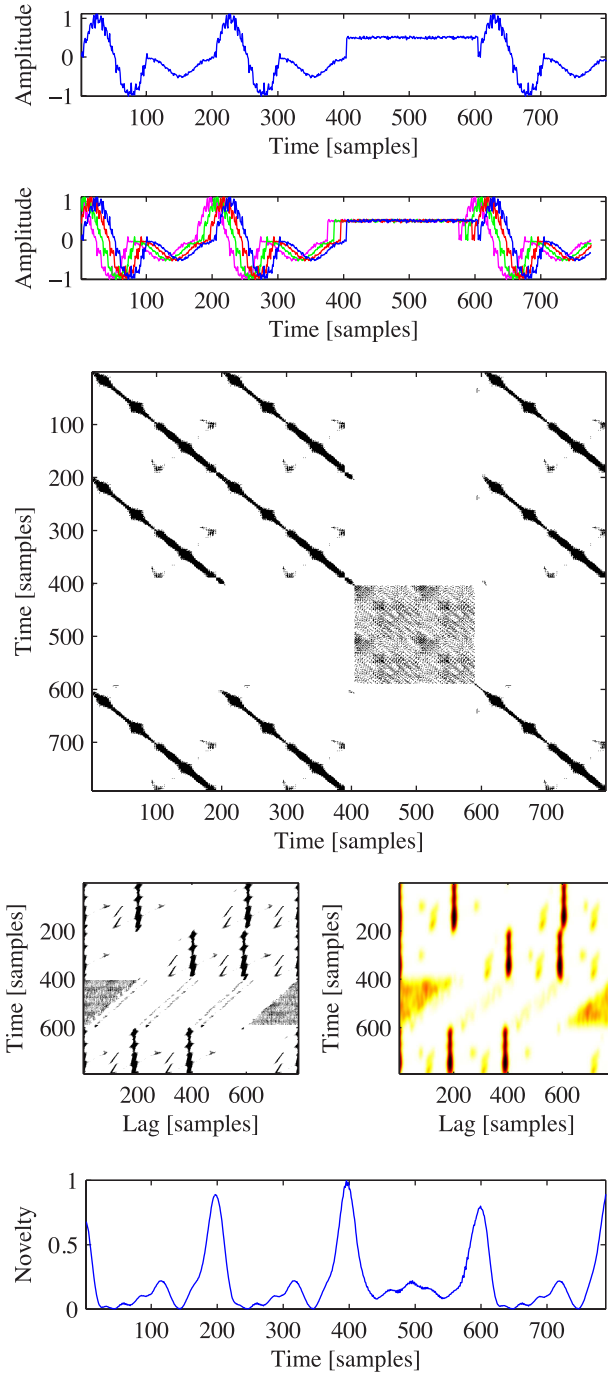


Figure 1: Illustration of the method using an artificially generated time series. From top to bottom, sub-figures show X , $\hat{X}$, R , L and P , and $\mathbf{c}$. Notice how R , and therefore also L and P , reflect the homogeneity region between 400 and 600 and the repetitions starting near 200 and 600. These boundaries are clearly reflected as prominent peaks in $\mathbf{c}$.

The estimated kernel density P can be seen as a time series along the time axis (Fig. 1, third row, right). Structure

	Num.	Length	N. Bound.	Interval
BEATLES-A	174	158.2 (51.5)	8.2 (2.3)	17.2 (12.3)
BEATLES-B	180	162.9 (56.6)	9.2 (2.3)	16.0 (13.9)
RWC-POP-A	100	242.2 (41.5)	16.1 (4.0)	14.1 (6.8)
RWC-POP-B	100	224.1 (41.4)	16.8 (3.4)	13.7 (7.2)

Table 1: Datasets’ statistics (standard deviations into parenthesis): number of recordings, average length, average number of boundaries, and average length of inter-boundary interval (the time between two consecutive boundaries). Time values are given in secs.

features $\mathbf{p}_i$ are then defined to be the rows of P , i.e. $P = [\mathbf{p}_1, \dots, \mathbf{p}_N]^T$. Because of the nature of the recurrence plot, they encapsulate both homogeneities and repetitions. Furthermore, by employing a Gaussian kernel, they gain robustness against time and lag deviations and transitions between them become smooth (see below).

Novelty curve

Our observation is that structural boundaries of the time series $\hat{X}$ correspond to relative changes in the sequence of structure features P . To measure these changes we compute the difference between successive structure features $\mathbf{p}_i$. This yields a unidimensional novelty curve $\mathbf{c} = [c_1, \dots, c_{N-1}]$, where $c_i = \|\mathbf{p}_{i+1} - \mathbf{p}_i\|^2$ (we again use the Euclidean norm, but now take the square of it). The novelty curve $\mathbf{c}$ can be linearly normalized between 0 and 1 by subtracting its minimum value and subsequently dividing it by the resultant maximum (Fig. 1, bottom).

Positions of prominent peaks of $\mathbf{c}$ are finally selected as segment boundaries. Here, we opt for a rather simple peak selection strategy: a sample c_i is considered to be a peak if it is above a certain threshold δ and, at the same time, corresponds to the global maximum of a window of length λ centered at c_i . Finally, the locations of the boundaries of the original time series $\hat{X}$ are set to the locations of the selected peaks plus $w/2$ to compensate for the offset introduced by delay coordinates in Eq. 1.

Experimental setup

Datasets

To evaluate our method in the context of music boundary detection we use two benchmark music datasets with boundary and structure annotations: Beatles and RWC-Pop. The Beatles dataset corresponds to all the recordings in the 12 original albums of the band. There are two versions for ground truth annotations of this dataset, which are denoted as BEATLES-A³ and BEATLES-B⁴ (Table 1). Since many works in the literature have used BEATLES-A for evaluation, it is easy to compare the results obtained here with theirs.

The second dataset consists of all recordings of the Real World Computing Popular Music Database (Goto et al. 2002). These recordings represent Japanese mainstream music and, to a less extent, American chart hits. We also

³http://www.cs.tut.fi/sgn/arg/paulus/beatles_sections_TUT.zip

⁴<http://isophonics.net/content/reference-annotations>

use two versions of annotations as ground truth, which are denoted by RWC-POP-A⁵ and RWC-POP-B⁶ (Table 1). RWC-POP-B has been used as part of the Music Information Retrieval Evaluation eXchange (MIREX), an international evaluation campaign for certain music processing tasks (Downie 2008). The RWC recordings and the two annotations are publicly available.

Music descriptor time series

The time series we use are sequences of harmonic pitch class profiles (HPCPs; Gómez 2004). Pitch class profiles (PCP; also called chroma) are widely used in the music information retrieval community (Casey et al. 2008; Müller et al. 2011) and have proven to work well as primary source of information for many music processing tasks, including boundary and structure retrieval (Paulus et al. 2010). PCP descriptors represent the harmonic (or tonal) properties of a music signal, and are specially suitable for highlighting structure repetitions.

PCP descriptors are derived from the frequency dependent energy in a given range in short-time spectral representations of audio signals computed in a moving window. This energy is usually mapped into an octave-independent histogram representing the relative intensity of each of the 12 semitones of the western chromatic scale (12 pitch classes: C, C#, D, D#, etc.). To normalize with respect to loudness, this histogram can be divided by its maximum value, thus leading to values between 0 and 1. In general, PCPs are robust against non-tonal components (e.g. ambient noise or percussive sounds) and independent of timbre and the specific instruments used.

HPCPs are an enhanced version of PCPs: they reduce the influence of noisy spectral components, are tuning independent, and take into account the presence of harmonic frequencies (Gómez 2004). The computation of HPCPs in a moving window results in a 12-dimensional time series X for each music recording (Fig. 2, top). We use the same implementation and parameters employed by Serrà et al. 2008, with 12 pitch classes, a window length of 0.186 sec, and a hop size of 0.14 sec⁷.

Method parameters

From the explanation of the method it may seem that quite a number of parameters need to be adjusted. However, by considering the nature of the time series and the characteristics of the task, many of them can be pre-specified. For instance, since the time series we will consider fluctuate rapidly, we set $\tau = 1$ so that no information from the recent past of $\mathbf{x}_i$ is lost in building $\hat{\mathbf{x}}_i$ (Eq. 1). Similarly, since no dramatic speed or tempo change may occur in our time series, there will be few fluctuations along the lag dimension of L , and therefore a value of $s_1 = 0.3$ sec for the kernel G will suffice to track them (Eq. 5). For the Gaussian windows $\mathbf{g}_t$ and $\mathbf{g}_l$ we can

use a variance $\sigma^2 = 0.16$, what ensures a value close to 0 at the borders of $\mathbf{g}$ (Simonoff 1996). Finally, to take the peaks of $\mathbf{c}$, we set $\delta = 0.05$ (5% of the maximum amplitude of $\mathbf{c}$; values of $\delta \in [0, 0.25]$ did not affect the results in a substantial way) and $\lambda = 12$ sec (note that we force boundaries to be separated at least 6 sec; therefore we prevent the trivial case of Baseline 3 mentioned in the next subsection).

The previous setting leaves us with just three important parameters: m (the amount of recent past we consider for $\mathbf{x}_i$; Eq. 1), κ (which controls the amount of a sample’s nearest neighbors in R ; Eq. 2), and s_t (the length of the time dimension of the kernel G ; Eq. 5). Thus, our method based on structure features is mostly parameterized by $\text{SF}(m, \kappa, s_t)$. We here do not perform an exhaustive search for the best parameters, but report results only for some representative parameter combinations. As it turns out, the results are already very good without any optimization and rather independent of the specific parameter settings.

The parameter values we try for m , κ , and s_t can be also justified by the nature of the data and the characteristics of the task. Suitable values for m may be comprised between 0 and 5 secs, according to the literature on short-term memory (cf. Baddeley 2003). Suitable values for κ may be roughly below 0.05 (i.e. below 5% of the length of our time series). If we suppose we have 10 repetitions in a time series of length 1000, this implies $\kappa = 0.01$ (or higher if we consider some noisy matches in R). Finally, suitable values for s_t may be found around 30 secs. If the length of our Gaussian kernel is 30 secs, then the Gaussian shape is maximal at 15 secs and decreases by 50% at 8 and 23 secs (Simonoff 1996). This yields an ‘effective’ kernel length of approximately 15 secs, close to the average time between boundaries shown in Table 1.

Evaluation measures

We use the same evaluation measures as used in the literature of music boundary detection: hit rates and median deviations (cf. Smith 2010). For hit rates, segment boundaries are accepted to be correct if they are within a certain threshold from a boundary in the ground truth annotation. Common thresholds are 0.5 and 3 sec (Ong and Herrera 2005; Turnbull et al. 2007). Based on the matched hits, standard precision, recall, and F-measure can be computed for each music recording and averaged across the whole dataset. Since we observed that some annotations were not accurate at a resolution of 0.5 sec, we only report precision, recall, and F-measure using a 3 sec threshold.

Additionally, two median deviation values can be computed, counting the median times (given in secs) from true-to-guess and from guess-to-true (Turnbull et al. 2007). The median true-to-guess (MT2G) is the median time from boundaries in the ground truth to the closest boundaries in the result. The median guess-to-true (MG2T) is, similarly, the median time from boundaries in the result to boundaries in the ground truth. Median values are also computed for each music recording and averaged across the whole dataset.

One should be cautious with the aforementioned evaluation measures, since they may not be the most suitable for the task. For instance, placing a boundary every second can

⁵<http://staff.aist.go.jp/m.goto/RWC-MDB/AIST-Annotation>

⁶<http://musicdata.gforge.inria.fr>

⁷We express all time-related values in seconds; they can be easily converted to samples by multiplying by the sampling rate $0.14^{-1} = 7.14$ Hz.

already yield a Recall of 1 and an F-measure close to 0.5 (Smith 2010). The two median times may also hide bad results for some fraction of the boundaries. However, for the sake of comparison with existing approaches, we also revert to these measures. Additionally, we provide three baselines for the evaluation measures above: placing a boundary at every average boundary length (Baseline 1), placing a certain number of boundaries randomly (Baseline 2), and placing a boundary every 3 secs (Baseline 3).

In addition, one should note that two different human annotators could disagree in the placement of a music structure boundary, or even whether a given musical event or sequence should be considered as such (see Peeters and Deruty 2009 or, for instance, the summaries for the two versions of the annotations we show in Table 1). To obtain a reference of human performance when evaluating against our ground truth we propose to cross-evaluate annotations. That is, given two annotations for the same music dataset, we use annotation B as a result when evaluating with the ground truth provided by annotation A and vice versa (e.g. using BEATLES-B for evaluating BEATLES-A). This way we can have a rough upper estimate of how a human would perform in the boundary detection task.

Results

Before presenting the quantitative results, we first give an illustrating example based on a real-world HPCP time series computed from “All you need is love” by The Beatles (Fig. 2). We see that homogeneities (gray areas) and repetitions (straight diagonal lines) are already visible in the recurrence matrix R . In this case, the majority of the detected boundaries correspond to real boundaries (Fig. 2, bottom). Interestingly, there is an undetected boundary at sample 683, which would have been detected if we had not set a too strict λ for preventing the trivial case of Baseline 3 (see above). This boundary corresponds to a change from a guitar solo to a ‘bridge’ part, both before the voice enters again. The two false positive boundaries towards the end (samples 1408 and 1518) correspond to a short musical quotation of the main chorus of the song “She loves you”, which The Beatles included in the final fade-out of this recording. These (actually musically meaningful) boundaries had not been annotated by humans, which again highlights the problems of collecting ground truth annotations for this task.

Let us turn to quantitative results (Tables 2 and 3). First, we observe that, independently of the dataset and ground truth we use, different parameter combinations for the structure feature (SF) approach yield similar results, no matter which evaluation measure we consider. Actually, many of these results turned out to be not statistically significantly different⁸ between them. This highlights the robustness of the approach against different parameter settings.

Second, we can see that SF reaches accuracies that are

⁸Statistical significance was assessed with a T-test ($p < 0.01$), assuming a Gaussian distribution of the evaluation measures. When standard deviations were not reported in the literature, equal variances were assumed for the test. In the rest of our manuscript, standard deviations are always reported into parenthesis.

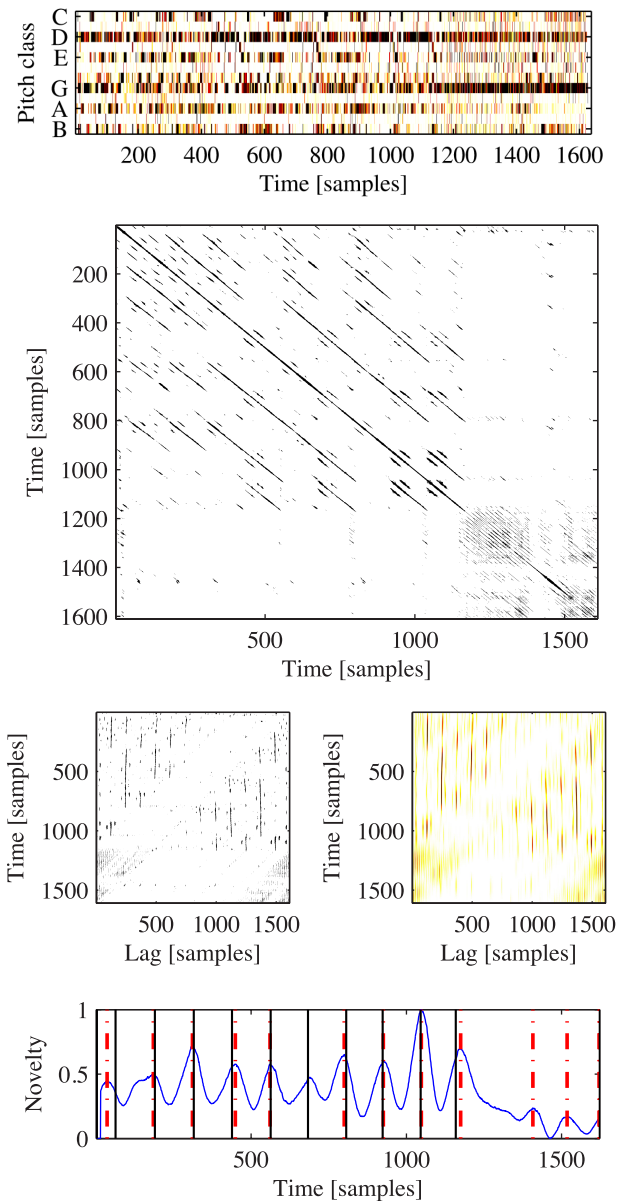


Figure 2: Example with “All you need is love” from The Beatles. From top to bottom the plots show X , R , L and P , and $\mathbf{c}$ (we omit $\hat{X}$ due to space constraints). The novelty curve $\mathbf{c}$ also shows the found boundaries (red dash-dotted lines) and the ground truth boundaries (black solid lines).

very far above the baselines (Table 2). The highest F-measure obtained was 0.576 (0.094) with Baseline 3 and RWC-POP-A, whereas the lowest F-measure obtained by the SF approach is 0.724 (0.152) with BEATLES-A. Human performance is still higher than our approach’s, but the difference gets tighter. For instance, with RWC-POP-B we get an F-measure of 0.800 (0.107) and human performance is at 0.899 (0.086). MT2G and MG2T are quite different though.

Third, the accuracies achieved by SF are statistically

Method	F-meas	Precision	Recall	MT2G	MG2T
BEATLES-A					
Baseline 1 (every 17 sec)	0.403	0.387	0.438	4.74	3.77
Baseline 2 (8 rand. bound.)	0.410	0.401	0.441	4.42	4.20
Baseline 3 (every 3 sec)	0.505	0.347	0.998	4.94	0.64
Human	0.911	0.889	0.937	0.28	0.27
BEATLES-B					
Baseline 1 (every 16 sec)	0.404	0.388	0.443	4.97	3.73
Baseline 2 (9 rand. bound.)	0.465	0.440	0.519	4.47	3.13
Baseline 3 (every 3 sec)	0.516	0.355	1.000	5.15	0.69
Human	0.911	0.937	0.889	0.27	0.28
RWC-POP-A					
Baseline 1 (every 14 sec)	0.423	0.421	0.437	3.85	3.52
Baseline 2 (16 rand. bound.)	0.449	0.478	0.438	3.41	4.00
Baseline 3 (every 3 sec)	0.576	0.411	1.000	3.88	0.68
Human	0.899	0.921	0.891	0.23	0.33
RWC-POP-B					
Baseline 1 (every 14 sec)	0.424	0.401	0.462	4.08	3.31
Baseline 2 (17 rand. bound.)	0.453	0.442	0.478	3.80	3.37
Baseline 3 (every 3 sec)	0.569	0.402	1.000	4.05	0.67
Human	0.899	0.891	0.921	0.33	0.23

Table 2: Baselines and estimated human accuracies (best baseline F-measures are highlighted in bold). Baselines 1 and 2 use the information specified in Table 1.

significantly higher than the ones reported in the literature so far (Table 3). For instance, the best F-measures we could find for BEATLES-A were all below 0.62, as reported by Smith (2010). Contrastingly, our approach reaches 0.752 (0.152). With RWC-POP-A, the same author reported higher F-measures, but all below 0.68. With the same ground truth we reach 0.791 (0.122). With RWC-POP-B, Sargent et al. (2011) obtained an F-measure of 0.627, whereas we reach 0.800 (0.107). MT2G and MG2T measures also corroborate a significant difference in accuracy between the SF approach and the numbers available in the literature.

Apart from the literature results shown in Table 3, one should mention the F-measure of 0.75 obtained by Ong and Herrera (2005) on a 54-song subset of BEATLES-A. Their method was quite elaborate, involving two phases and nine steps, combining the information from different music descriptor time series, and applying a post-processing step to refine the results. Using the same song subset with SF(4,0.04,30) we obtain an F-measure of 0.77 (0.134), which is higher than theirs but not statistically significantly different. Therefore, we obtain a comparable accuracy with a much simpler and generic approach. In addition, it should be noted that we do not perform an exhaustive parameter optimization which may further improve the results.

We should finally discuss the results obtained by Turnbull et al. (2007), who report relatively high F-measures for the RWC-Pop recordings using a supervised approach combining decision stumps and boosting. However, it is difficult to compare such results. First, they used an alternative annotation just considering 4 structural labels. This annotation discards many boundaries, and has been reported to be slightly unreliable (Smith 2010). Furthermore, they reported hit rates with a threshold of 0.5 instead of 3 sec. As mentioned, we empirically found some inaccuracies in the annotations within a resolution of 0.5 sec, which could only be correctly detected by a supervised approach. Finally, we should mention that they achieve an MT2G of 1.58 and an

Method	F-meas	Precision	Recall	MT2G	MG2T
BEATLES-A					
<i>Levy & Sandler (2008)^a</i>	<i>0.612</i>	<i>0.600</i>	<i>0.646</i>	-	-
<i>Peiszer (2007)^b</i>	<i>0.616</i>	<i>0.515</i>	<i>0.824</i>	-	-
SF(4,0.04,30)	0.752	0.726	0.797	1.87	1.17
SF(3.5,0.03,30)	0.749	0.726	0.790	1.89	1.27
SF(3,0.03,25)	0.735	0.689	0.806	2.04	1.19
SF(2.5,0.02,25)	0.724	0.679	0.792	2.01	1.22
BEATLES-B					
SF(4,0.04,30)	0.769	0.753	0.805	2.00	1.13
SF(3.5,0.03,30)	0.774	0.760	0.807	1.91	1.07
SF(3,0.03,25)	0.756	0.718	0.819	2.14	1.03
SF(2.5,0.02,25)	0.750	0.715	0.810	2.07	1.08
RWC-POP-A					
<i>Levy & Sandler (2008)^b</i>	<i>0.661</i>	<i>0.774</i>	<i>0.755</i>	-	-
<i>Peiszer (2007)^b</i>	<i>0.680</i>	<i>0.613</i>	<i>0.807</i>	-	-
SF(4,0.04,30)	0.764	0.823	0.729	1.42	1.70
SF(3.5,0.03,30)	0.771	0.834	0.733	1.42	1.67
SF(3,0.03,25)	0.786	0.813	0.776	1.35	1.45
SF(2.5,0.02,25)	0.791	0.817	0.783	1.32	1.41
RWC-POP-B					
<i>Peeters (2010)^c</i>	<i>0.571</i>	<i>0.575</i>	<i>0.591</i>	<i>1.57</i>	<i>1.78</i>
<i>Mauch et al. (2009)^c</i>	<i>0.605</i>	<i>0.736</i>	<i>0.544</i>	<i>2.66</i>	<i>1.27</i>
<i>Sargent et al. (2011)</i>	<i>0.627</i>	<i>0.634</i>	<i>0.641</i>	-	-
SF(4,0.04,30)	0.781	0.821	0.757	1.27	1.35
SF(3.5,0.03,30)	0.790	0.834	0.764	1.25	1.33
SF(3,0.03,25)	0.798	0.808	0.803	1.23	1.20
SF(2.5,0.02,25)	0.800	0.810	0.805	1.27	1.18

Table 3: Boundary detection results (best F-measures highlighted in bold). Lines in italics correspond to the best results reported in the literature so far. The ^{a,b,c} superscripts denote results reported by Paulus and Klapuri (2009), Smith (2010), and the MIREX campaign (Downie 2008), respectively. Standard deviations for SF were 0.16 ± 0.02 for precision, recall, and F-measure, 1.2 ± 0.9 for MT2G, and 0.7 ± 0.2 for MG2T.

MG2T of 4.29, which are worse than the values reported here for RWC-POP-A and RWC-POP-B.

Conclusion

In this paper we have introduced an unsupervised method for detecting time series boundaries based on structure features, incorporating three basic principles for boundary detection: novelty, homogeneity, and repetition. We tested the usefulness of the method in the task of segmenting music recordings based on their structure and found that it performed significantly better than any other existing approach. Since the proposed method is generic, it can be easily applied to time series from other scientific domains beyond music information retrieval. In the future we intend to use similar techniques for segmenting motion-capture data into coherent body movements or for automatically detecting non-trivial boundaries in stock market data.

Acknowledgments

JS and JLA acknowledge 2009-SGR-1434 from Generalitat de Catalunya, TIN2009-13692-C03-01 from the Spanish Government, and EU Feder Funds. JS also acknowledges JAEDOC069/2010 from Consejo Superior de Investigaciones Científicas. MM and PG acknowledge the Cluster of Excellence on Multimodal Computing and Interaction (MMCI).

References

- Armstrong, T., and Oates, T. 2007. UNDERTOW: multi-level segmentation of real-valued time series. In *Proc. of the AAAI Int. Conf. on Artificial Intelligence*, 1842–1843.
- Baddeley, A. 2003. Working memory: looking back and looking forward. *Nature Reviews Neuroscience* 4(10):829–839.
- Barrington, L.; Chan, A. B.; and Lanckriet, G. 2010. Modeling music as a dynamic texture. *IEEE Trans. on Audio, Speech, and Language Processing* 18(3):602–612.
- Casey, M. A.; Veltkamp, R.; Goto, M.; Leman, M.; Rhodes, C.; and Slaney, M. 2008. Content-based music information retrieval: current directions and future challenges. *Proceedings of the IEEE* 96(4):668–696.
- Downie, J. S. 2008. The music information retrieval evaluation exchange (2005–2007): a window into music information retrieval research. *Acoustical Science and Technology* 29(4):247–255.
- Firoiu, L., and Cohen, P. R. 2002. Segmenting time series with a hybrid neural networks - hidden Markov model. In *Proc. of the AAAI Int. Conf. on Artificial Intelligence*, 247–252.
- Foote, J. 2000. Automatic audio segmentation using a measure of audio novelty. In *Proc. of the IEEE Int. Conf. on Multimedia and Expo (ICME)*, 452–455.
- Gómez, E. 2004. Tonal description of polyphonic audio for music content processing. *INFORMS Journal on Computing* 18(3):294–304.
- Goto, M.; Hashiguchi, H.; Nishimura, T.; and Oka, R. 2002. RWC music database: popular, classical, and jazz music databases. In *Proc. of the Int. Conf. on Music Information Retrieval (ISMIR)*, 287–288.
- Goto, M. 2006. A chorus section detection method for musical audio signals and its application to a music listening station. *IEEE Trans. on Audio, Speech and Language Processing* 14(5):1783–1794.
- Kantz, H., and Schreiber, T. 2004. *Nonlinear time series analysis*. Cambridge, UK: Cambridge University Press.
- Keogh, E. 2011. Machine learning in time series databases (and everything is a time series!). Tutorial at the AAAI Int. Conf. on Artificial Intelligence.
- Levy, M., and Sandler, M. 2008. Structural segmentation of musical audio by constrained clustering. *IEEE Trans. on Audio, Speech and Language Processing* 16(2):318–326.
- Marwan, N.; Romano, M. C.; Thiel, M.; and Kurths, J. 2007. Recurrence plots for the analysis of complex systems. *Physics Reports* 438(5–6):237–329.
- Mauch, M.; Noland, K. C.; and Dixon, S. 2009. Using musical structure to enhance automatic chord transcription. In *Proc. of the Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, 231–236.
- Mueen, A., and Keogh, E. 2010. Online discovery and maintenance of time series motifs. In *Proc. of the Conf. on Knowledge Discovery and Data Mining (KDD)*, 1089–1098.
- Müller, M.; Ellis, D. P. W.; Klapuri, A.; and Richard, G. 2011. Signal processing for music analysis. *IEEE Journal of Selected Topics in Signal Processing* 5(6):1088–1110.
- Ong, B. S., and Herrera, P. 2005. Semantic segmentation of music audio contents. In *Proc. of the Int. Computer Music Conf. (ICMC)*.
- Paulus, J., and Klapuri, A. 2009. Music structure analysis using a probabilistic fitness measure and a greedy search algorithm. *IEEE Trans. on Audio, Speech, and Language Processing* 17(6):1159–1170.
- Paulus, J.; Müller, M.; and Klapuri, A. 2010. Audio-based music structure analysis. In *Proc. of the Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, 625–636.
- Peeters, G., and Deruty, E. 2009. Is music structure annotation multi-dimensional? A proposal for robust local music annotation. In *Proc. of the Workshop on Learning the Semantics of Audio Signals (LSAS)*, 75–90.
- Peeters, G. 2010. MIREX 2010 music structure segmentation task: IRCAMSUMMARY submission. Music Information Retrieval Evaluation eXchange (MIREX).
- Peiszer, E. 2007. *Automatic audio segmentation: segment boundary and structure detection in popular music*. MSc thesis, Vienna University of Technology, Vienna, Austria.
- Sargent, G.; Bimbot, F.; and Vincent, E. 2011. A regularity-constrained Viterbi algorithm and its application to the structural segmentation of songs. In *Proc. of the Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, 483–488.
- Serrà, J.; Gómez, E.; Herrera, P.; and Serra, X. 2008. Chroma binary similarity and local alignment applied to cover song identification. *IEEE Trans. on Audio, Speech, and Language Processing* 16(6):1138–1151.
- Serrà, J.; Serra, X.; and Andrzejak, R. G. 2009. Cross recurrence quantification for cover song identification. *New Journal of Physics* 11(9):093017.
- Simonoff, J. S. 1996. *Smoothing methods in statistics*. Springer Series in Statistics. Berlin, Germany: Springer.
- Smith, J. B. L. 2010. *A comparison and evaluation of approaches to the automatic formal analysis of musical audio*. MSc thesis, McGill University, Montreal, Canada.
- Turnbull, D.; Lanckriet, G.; Pampalk, E.; and Goto, M. 2007. A supervised approach for detecting boundaries in music using difference features and boosting. In *Proc. of the Int. Conf. on Music Information Retrieval (ISMIR)*, 51–54.
- Vahdatpour, A.; Amini, N.; and Sarrafzadeh, M. 2009. Toward unsupervised activity discovery using multi-dimensional motif detection in time series. In *Proc. of the Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 1261–1266.