SEQUENCE REPRESENTATION OF MUSIC STRUCTURE USING HIGHER-ORDER SIMILARITY MATRIX AND MAXIMUM-LIKELIHOOD APPROACH

Geoffroy Peeters
Ircam Sound Analysis/Synthesis Team - CNRS STMS
1, pl. Igor Stranvinsky - 75004 Paris - France

ABSTRACT

In this paper, we present a novel method for the automatic estimation of the structure of music tracks using a sequence representation. A set of timbre-related (MFCC and Spectral Contrast) and pitch-related (Pitch Class Profile) features are first extracted from the signal leading to three similarity matrices which are then combined. We then introduce the use of higher-order (2nd and 3rd order) similarity matrices in order to reinforce the diagonals corresponding to common repetitions and reduce the background noise. Segments are then detected and a maximum-likelihood approach is proposed in order to derive simultaneously the underlying sequence representation of the music track and the most representative segment of each sequence. The proposed method is evaluated positively on the MPEG-7 "melody repetition" test set.

1 INTRODUCTION

Music structure discovery (MSD) aims at estimating automatically the structure of a music track by analyzing its audio signal. It has become a major topic of interest in the recent years because it allows the development of new paradigms: active music listening (intra-document browsing [4]), acoustic browsing of music catalogues (fast browsing using automatically generated audio summaries [20] or using automatically located chorus, key-phrase, audio-thumbnail [15] [5] [3]), music creation (automatic segmentation into cognitively similar parts [12], music mosaicing), media compression [12] and automatic music analysis (understanding music structure through acoustic analysis).

MSD algorithms always start by extracting a set of features from the audio signal. The features are then used to detect repetitions of the signal content over time. This notion of "repetition" and "detection of repetition" is the basis of all MSD algorithms developed so far. It is also their main limitation since it does not allow detecting variations or evolutions of a part [1] . The choice of the features therefore plays a central role since it guides the kind of repetitions that can be observed: repetitions can be based on instrument-background repetitions (timbre-related features are for example used by [8]), repetitions of melody or chord-succession (pitch-related used by [3]) or repetitions of rhythm patterns (rhythm-related used by [20] [13]). In the current work both timbre-related and pitch-related features are used.

The temporal structure of a music track can then be visualized using a recurrence plot more often called a similarity matrix in the case of music [9] which represents the similarity between each pair of features over time. In order to extract from this visual representation, a numerical representation of the structure of a track, two kinds of representation can be used leading to two different approaches [19]: the state and the sequence representation.

The **"state" representation** (see left part of Fig. 1) considers that a music track is a succession of parts called states and that each time of a music track has emitted a specific state. A state is defined as a set of contiguous times, which contains similar acoustical information. A state does not need to be repeated later in the track. The notion of states is closely related to the notion of parts in popular music (introduction, verse, chorus and bridge) because for popular music the musical background is often constant during the duration of each part. In this case, the goal of MSD algorithms is to find the states that have been emitted at each time. The algorithms rely mainly on segmentation (novelty measure of [8]), partitional, agglomerative or spectral clustering algorithms [15] [6] [1] or hidden Markov models ([15], [20]).

The **"sequence" representation** (see right part of Fig. 1) considers that there exist sequences of time in the music track that are repeated over the track. A sequence is defined as a set of successive times, which is similar to another set of successive times. However the times inside a given sequence do not need to be necessarily identical to each other. All the times of a music track do not belong necessarily to a sequence. The notion of sequence is closely related to the notion of melody (sequence of notes) or chord succession in popular music. These sequences are visible in a similarity matrix through the diagonals, which represent succession of pairs of times with high similarity. The sequence approach allows a more precise description than the state approach, since it allows to detect only the parts which are repeated melodies and are therefore cognitively more memorable.

When considering the sequence representation, most approaches only attempts to detect the most representative audio extract from the similarity matrix in order to create a thumbnail [3], [5]. Few papers address the problem of estimating the actual sequence representation from the similarity matrix. When dealing with this problem most authors use Dynamic Time Warping or pattern matching techniques [7] [2] [16] [11]. Recent approaches combine

[1] Note however that [11] takes tonality modulation into account.

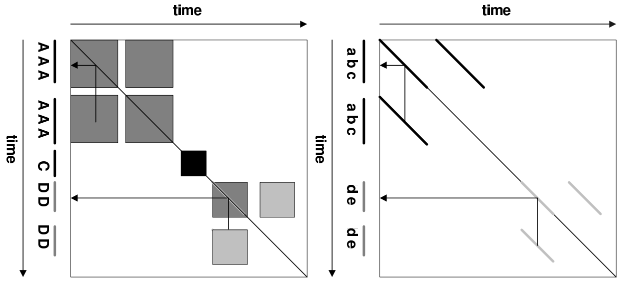

Figure 1. Structure representation in a similarity matrix as [left part] states: we observe three states noted A, C and D. The times noted A belong to the state A. Note that the state C is not repeated later in the track. [right part] sequences: we observe two sequences noted abc and de. Note that a sequence cannot exist if it is not repeated later in the track.

DTW with a hierarchical approach of the structure detection [18] [17]. Despite its efficiency, the DTW approach remains very heavy in computation time. In this paper, we propose a fast method for the estimation of the sequence of a music track based on a maximum-likelihood approach (parts 2.4 and 2.5). Other contributions of this paper are the simultaneous use of timbre and harmonic-related features combined into a unique similarity matrix (part 2.1) and the use of higher-order similarity matrices (part 2.2). Finally part 3 presents the evaluation of our system on the MPEG-7 “melody repetition” test set.

2 PROPOSED METHOD

2.1 Feature extraction and similarity matrix

The first stage of our system extracts features from the audio signal. For the reasons mentioned above (the fact that repetitions can be related either to timbre or pitch observations), three set of audio features are extracted:

- 13 Mel Frequency Cepstral Coefficients (excluding the 0th/ DC-component coefficient),
- 12 Spectral Contrast coefficients [14] (spectral contrasts and spectral valley coefficients into 6 frequency bands $[0, \frac{sr}{2^6}]$, $[\frac{sr}{2^6}, \frac{sr}{2^5}]$, ... $[\frac{sr}{2^2}, \frac{sr}{2}]$[2]),
- 12 Pitch Class Profile coefficients [10].

The frame analysis is performed with a window length of 80ms and a hop size of 40ms. Each dimension of the features is then modeled over time by its mean values over a sliding window of 4s with hop size of 500ms.

Principal Component Analysis is then applied to the three feature sets separately. For each set, only the principal components explaining more than 10% of the total variance are kept. The data are then projected on the retained principal components leading to the final features.

From the three modified feature sets, we compute separately three similarity matrices using an Euclidean distance. The matrices are then normalized to the range $[0, 1]$ and added. We note $S(t_x, t_y)$ the resulting matrix.

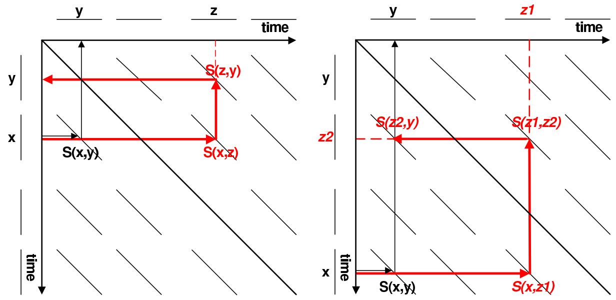

Figure 2. Computation of a [left] 2nd order similarity matrix [right] 3rd order similarity matrix

2.2 Higher order similarity matrix

We note $o(t_x)$ the feature vector extracted at time $t = t_x$. The similarity matrix $S(t_x, t_y)$ represents the similarity between two times t_x and t_y through the computation of the distance between the feature vectors extracted at time t_x and t_y. If t_z is a repetition of t_x, we observe a high value at $S(t_z, t_x)$. In the same way, if t_z is a repetition of t_y, we observe a high value at $S(t_z, t_y)$. By transitivity, since $o(t_z) \simeq o(t_x)$ and $o(t_z) \simeq o(t_y)$, we should have $o(t_y) \simeq o(t_x)$ and observe a high value at $S(t_y, t_x)$. However, in practice, because repetitions are not exact repetitions and because of the noise in the features, this repetition can be masked. The higher order similarity matrix uses the transitivity property to recover those missing values and emphasize the repetitions.

We define a second order similarity matrix $S_2(t_x, t_y)$ as the similarity between times t_x and t_y through all the times t_z (see left part of Fig. 2):

$$S_2(t_x, t_y) = \int_{t_z} S(t_x, t_z) S(t_z, t_y) dt_z \quad (1)$$

$S_2(t_x, t_y)$ measures the similarity between t_x and t_y using the fact that if t_x is similar to t_z, and t_z to t_y then t_x and t_y should be similar.

In the same way, we can define a third order similarity matrix as the similarity between time t_x and t_y through all the times t_{z1},t_{z2} (see right part of Fig. 2):

$$S_3(t_x, t_y) = \int_{t_{z1}} \int_{t_{z2}} S(t_x, t_{z1}) S(t_{z1}, t_{z2}) S(t_{z2}, t_y) dt_{z1} dt_{z2} \quad (2)$$

$S_3(t_x, t_y)$ measures the similarity between t_x and t_y using the fact that if t_x is similar to t_{z1}, t_{z1} to t_{z2} and t_{z2} to t_y then t_x and t_y should be similar.

In Fig. 3, we represent the 1st, 2nd and 3rd order similarity matrix for the track “She Loves You” from The Beatles. Using the 2nd and 3rd order matrices, the repetitions, especially at the beginning, become more visible.

2.3 Sequence representation

From the higher-order similarity matrix we derive the sequence representation. This is done in two steps. We first detect in the matrix sets of diagonals from which we derive a set of segments (part 2.4). We then estimate the sequence representation that best explains the detected segments (part 2.5). In the rest of this part, we will use the following terms:

[2] sr stands for sampling rate.

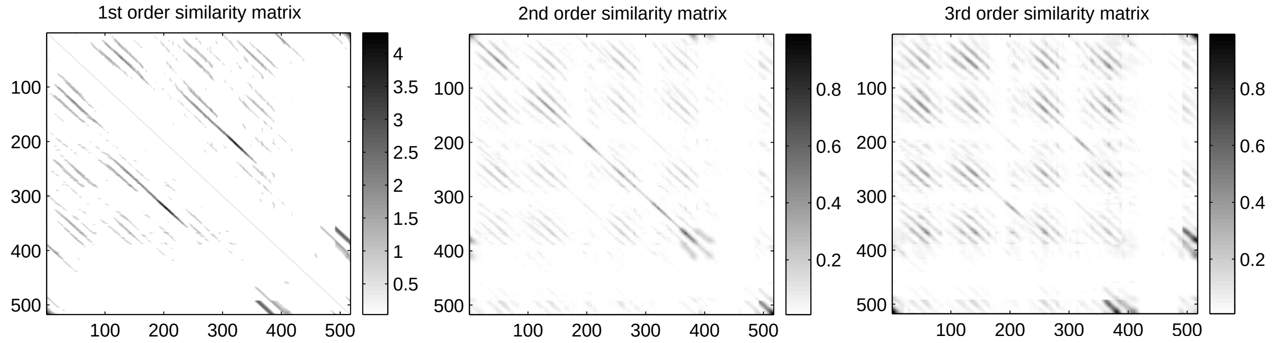

Figure 3. From left to right: 1st, 2nd, 3rd order similarity matrix on "She Loves You" from The Beatles

diagonal (line): a diagonal (line) is defined as a possibly discontinuous set of points in the similarity (lag) matrix,

segment: a segment is a set of successive (continuous) times defined by a starting and ending time. A diagonal in the matrix defines two segments: the original (projection on the x-axis) and the repetition one (projection on the y-axis).

sequence: a sequence is a set of segments representing similar information occurring at various times. A sequence is defined by a "mother" segment (the most typical segment) and a set of times which indicate at which times the "mother" segment is instantiated,

sequence representation: a sequence representation is defined by a set of sequences.

2.4 Diagonals (lines) and segments detection

2.4.1 Matrix filtering

In order to reinforce the diagonal elements in the matrix while removing the non-diagonal elements, a filter is applied to the matrix. The matrix $S(t_i, t_j)$ is first converted to a lag-matrix [3] $L(l_{ij}, t_j)$ with $l_{ij} = t_i - t_j$. The lag matrix transforms a diagonal repetition into a vertical constant-lag–line. The filter we use is the combination of a horizontal high-pass filter and a vertical low-pass filter. The high-pass filter is a gaussian kernel filter which is the combination of two opposed sign gaussian function: $g(t) = g_{\sigma_+}(t) - g_{\sigma_-}(t)$ [3]. In the experiment of part 3, we will use the following parameters: $\sigma_+ = 0.3s.$, $\sigma_- = 2s$. The low-pass filter is a simple averaging filter with length 8s.

2.4.2 Segment detection

The segments are detected from the resulting filtered high-order lag matrix using a method close to the one proposed by Goto [11]. Despite the fact that this method does not allow to detect repetitions of segments with time variations (accelerando, ritardando…), it was chosen because it is fast and most of the time reliable. We refer the reader to [11] for details about the method.

2.5 Sequence estimation using a maximum likelihood approach

The goal of the sequence representation is to represent all the segments detected in the matrix using the smallest possible set of sequences (mother segments and repetition times). In [19], we have proposed a method for solving this problem. The segments were first connected, then for each set of connected segments a mother segment was chosen. In this paper, we present a new approach, faster and more reliable, which allows solving the problems using a maximum likelihood approach. For each candidate mother segment, we measure how well it would "explain" the observed segments. We define $S_{seg}(t_i, t_j)$ a matrix with values set to 1 when a segment exist at (t_i, t_j) and to 0 otherwise The algorithm presented below is applied to $S_{seg}(t_i, t_j)$. The term "explain" is expressed by a score inspired by the "summary score" proposed by [5].

Proposed algorithm. We define m_{ij} as a candidate mother segment starting at time τ_i and ending at time $\tau_j > \tau_i$ (note that m_{ij} does not need to correspond to an existing segment). The times τ_i and τ_j define a row corridor in the matrix (see Fig. 4[A]). The summation over the length of the corridor (over all the columns of the matrix) is noted

$$\sigma(\tau) = \sum_{t=1}^{T} S_{seg}(\tau, t) \tag{3}$$

The summation over the width of the corridor (over the rows of the matrix defined by the corridor)

$$s_{ij}(t) = \sum_{\tau=\tau_i}^{\tau_j} S_{seg}(\tau, t) \;\; \forall t \in [1, T] \tag{4}$$

Using this notation, the "summary score" [5] for a segment m_{ij} would be $\sum_{\tau=\tau_i}^{\tau_j} \sigma(\tau)$ but would be computed using the feature-similarity matrix and not the segment-similarity matrix. Because of the use of S_{seg}, $\sigma(\tau)$ indicates the number of segments which are repetitions of a sequence existing at time τ.

First condition: If one segment crosses the corridor $[\tau_i, \tau_j]$ during an interval $t = [t_x, t_y]$ then $s_{ij}(t) = 1 \;\; \forall t \in [t_x, t_y]$ (see Fig. 4[A]). If two segments cross simultaneously the corridor during an interval $t = [t_x, t_y]$ then $s_{ij}(t) = 2 \;\; \forall t \in [t_x, t_y]$ (see Fig. 4[B]). Therefore $s_{ij}(t)$ provides an information about the number of simultaneous segments occurring during the interval $t = [t_x, t_y]$. Since two sequences cannot occurred simultaneously (only one mother

[3] $g_\sigma(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(t-\mu)^2}{2\sigma^2}}$

segment can be instantiated at a given time), values of $s_{ij}(t)$ larger than 1 should be avoided and τ_i and τ_j adapted in order to achieve that (by reducing the width of the corridor). The first condition is then: $s_{ij}(t) \leq 1 \;\; \forall t$ and τ_i and τ_j should be modified in order to fullfill that.

Segmentation: When $s_{ij}(t) \leq 1 \;\; \forall t$, applying a simple threshold ($s_{ij}(t) > 0$) allows to detect automatically the various segment occurrences of the mother segment m_{ij}. We note $[t_x^k, t_y^k] \;\; k \in [1, K]$ the starting and ending time of the k^{th} segment occurrence of the mother segment m_{ij}.

Second condition: However the condition $s_{ij}(t) = 1$ $\forall t \in [t_x^k, t_y^k]$) does not guarentee that $[t_x^k, t_y^k]$ is an instantiation of the mother segment m_{ij}. $[t_x^k, t_y^k]$ could also be - a part (the beginning or ending) of another sequence (see Fig. 4[C]) - the succession of two non-overlapping segments (see Fig. 4[D] and Fig. 4[E]). For this reason, we need to add a second condition. For this, we define a second score which is specific to each interval $k \in K$:

$$\sigma_{ijk}(\tau) = \sum_{t=t_x^k}^{t_y^k} S_{seg}(\tau, t) \;\; \forall \tau \in [\tau_i, \tau_j] \tag{5}$$

In the ideal case $\sigma_{ijk}(\tau)$ should be equal to 1 $\;\forall \tau \in [\tau_i, \tau_j]$ (such as $s_{ij}(t)$ should be equal to 1 $\;\forall t \in [t_x^k, t_y^k]$). If there exists a value $\tau \in [\tau_i, \tau_j]$ such that $\sigma_{ijk}(\tau) = 0$, it indicates that the segment only partially covers the duration of m_{ij} (see Fig. 4[C]). If there exists a value $\tau \in [\tau_i, \tau_j]$ such that $\sigma_{ijk}(\tau) > 1$, it indicates that the interval contains several segments (see Fig. 4[D][E]). The second condition is then: $\sigma_{ijk}(\tau) = 1 \;\; \forall \tau \in [\tau_i, \tau_j] \;\; \forall k \in K$. τ_i and τ_j should be modified in order to fullfill that condition too.

Best fit approach: Theoretically the width of the corridor should be reduced until $s_{ij}(t) \leq 1 \; \forall t$ and $\sigma_{ijk}(\tau) = 1 \; \forall \tau \in [\tau_i, \tau_j]$ for all the $k \in K$ intervals. In practice, since the detected segments are not perfect, this could lead to unnecessary reduction of the corridor width hence of the mother segment length. A best fit approach between errors and corridor-width-reduction is therefore used. For this, we define the following scores:

- $\epsilon_s(k)$: the number of frames fow which $s_{ij}(t \in [t_x^k, t_y^k])$ is > 1, relative to the length of the interval $(t_y^k - t_x^k)$
- ε_s: the number of intervals $k \in K$ for which $\epsilon_s(k)$ exceeds a given threshold T_s
- $\epsilon_\sigma(k)$: the number of frames for which $\sigma_{ijk}(\tau \in [\tau_i, \tau_j])$ differs from 1, relative to the length of the interval $(\tau_j - \tau_i)$
- ε_σ: the number of intervals $k \in K$ for which $\epsilon_\sigma(k)$ exceeds a given threshold T_σ

The corridor is reduced until both ε_s and ε_σ fall below a third threshold T. $T = 0$ indicates that we do not allow any overlap of sequences. $T = 1$ indicates that we allow any overlap or partial sequences (this is the summary score proposed by [5]).

How is the corridor reduced ? The corridor can be reduced by increasing τ_i or decreasing τ_j. For a specific interval $k \in K$, a value of $\sigma_{ijk}(\tau) > 1$ for τ close to τ_i indicates that an extra segment appears at the beginning of the interval $[t_x^k, t_y^k]$. In this case the width of the corridor ij should be reduced by increasing the value of τ_i. A value of $\sigma_{ijk}(t) > 1$ for τ close to τ_j indicates that an extra segment appears at the end of the interval $[t_x^k, t_y^k]$ and the width of the corridor should be reduced by decreasing τ_j. Each interval $[t_x^k, t_y^k]$ may require a different solution. Therefore the global action is made taking into account the best global action. This is made according to a vote: each interval k votes either to increase τ_i or decrease τ_j.

Score computation: After adaptation of τ_i and τ_j, a score is assigned to the current mother segment m_{ij}. It is defined as the sum of the lengths of all explained segments $[t_x^k, t_y^k]$, it represents the *likelihood* that this mother segment explains the observed segments. This process is repeated for each candidate mother segment. The candidate mother segment with the highest score (maximum likelihood) is chosen as the first (most important) mother segment.

Segment cancellation: The segments belonging to (that can be mostly explained by) this mother segment are then canceled. In order to do that, the values of the segment similarity matrix inside the corridor defined by the selected mother segment m_{ij} are canceled (set to 0). A new set of segments is then derived from the analysis of the new segment similarity matrix and the process is repeated for the detection of the next mother segment.

In theory any values of τ_i and τ_j can be chosen as the starting and ending time of a candidate mother segment. However, in order to save computation time, τ_i and τ_j are chosen from the set of detected segments.

3 EVALUATION

In this part we evaluate the performances of our algorithm. In particular, we compare the influence of the choice of the feature sets and the use of higher-order similarity matrix. This is, as far as we now, the first time an evaluation of sequence detection algorithm is performed.

3.1 Test set

For the evaluation of our system we have used the “melody repetition” part of the MPEG-7 test set. The MPEG-7 test set has been developed by the author for the task of music structure discovery. It is composed of two parts: state annotations (this part is currently merged with the QMUL test set[4] and sequence annotations [5]. The sequence test set is composed of 11 songs annotated into all their repeated melodies over time (up to 7 melodies).

3.2 Performance measure

Evaluating the performances of MSD algorithms is not an easy task. [1] and [18] already raised this issue and did proposals in the case of state representation (measuring segment boundaries and segment labels). In the case of sequence representation, measuring segment boundaries makes few sense: an annotated sequence ABCD can be detected as two sequences AB and CD or as A and BCD or as three sequences AB-, -BC- and -CD. Measuring sequence labels

[4] available at http:// www.elec.qmul.ac.uk/ digitalmusic/ downloads/

[5] available at http:// recherche.ircam.fr/ equipes/ analyse-synthese/ peeters/ mpeg7audio_msd_testset/

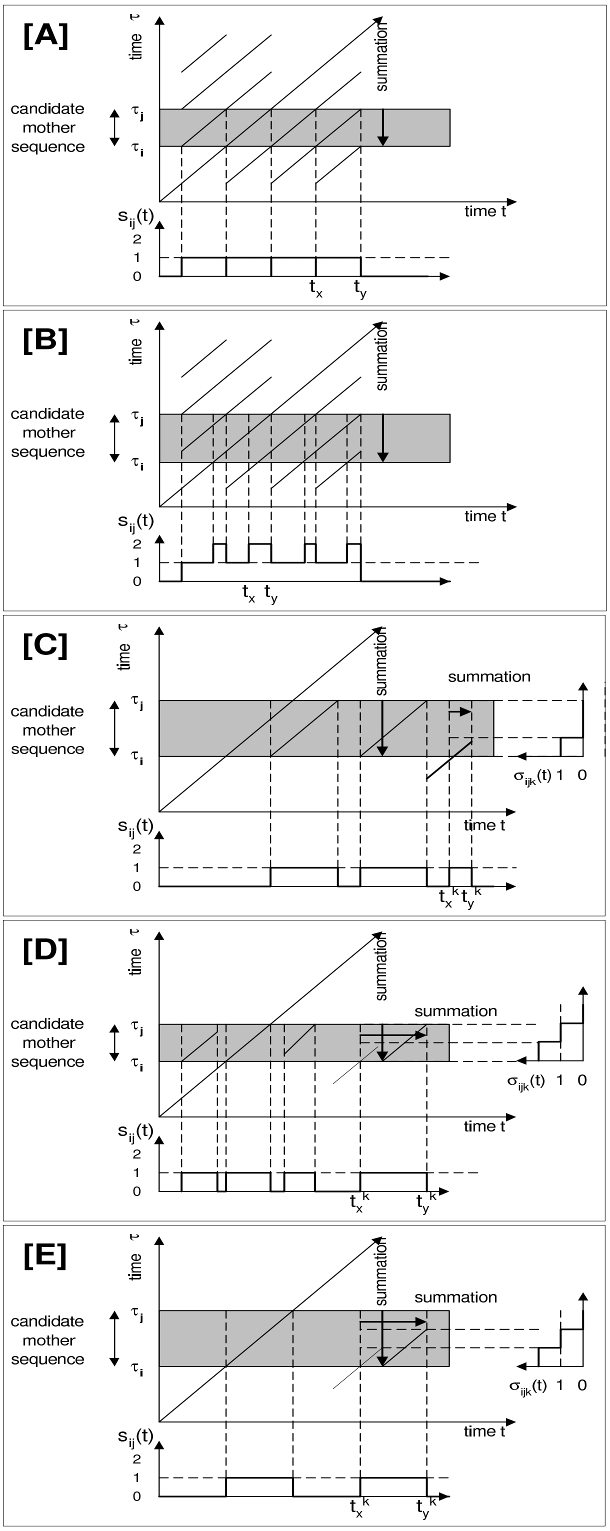

Figure 4. Sequence detection by maximum likelihood algorithm: [A] corridor $[\tau_i, \tau_j]$ of the candidate mother segment m_{ij} and corresponding function $s_{ij}(t)$; $s_{ij}(t) \leq 1$ indicates no simultaneous segments and allows segmentation; [B] $s_{ij}(t) > 1$ indicates simultaneous segments and requires further corridor width reduction; [C] $\sigma_{ijk}(\tau) = 0$ for the majority of $[t_x^k, t_y^k]$ indicates that the interval contains a partial sequence; [D] $\sigma_{ijk}(\tau) > 1$ indicates that the interval contains successive non-overlapping segments; [E] $\sigma_{ijk}(\tau) \neq 1$ indicates that the interval contains either successive non-overlapping segments or no segments

.

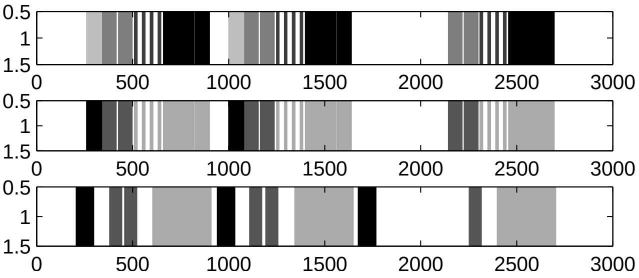

Figure 5. Mapping between annotated and detected sequence labels [top] annotated sequences $a_i(t)$ [middle] mapped annotated sequences $a_{k(j)}(t)$ [bottom] detected sequences $e_j(t)$ on "Smells like teen spirit" by Nirvana. The colors represent the various labels.

requires a previous mapping between annotated sequence labels and estimated sequence labels. The number of labels may differ between both: - 1) the annotation may gives finer details than what can be detected - 2) the annotation may group segments with different acoustical properties. The first case is more usual since - annotations tend to split melodies into sub-melody (according to lyric changes) - estimation tends to merge successive repeated melodies into a single one (if the verse is always repeated by the chorus then only the grouped verse-chorus segment will be detected). Therefore we allow mapping several annotated sequences to a unique estimated sequence (but not the opposite). We note $a_i(t)$ ($e_j(t)$) the time vector having a value of 1 when the annotated (estimated) sequence i (j) exists at time t and 0 otherwise. We assign each annotated sequence i to the estimated sequence j with the largest dot product: $k(j) = \text{argmax}_i \langle a_i(t), e_j(t) \rangle$ (j is the estimated sequence that best explains i). The mapping process is illustrated on Fig. 5. Finally, we assign a score to the estimated representation. This score is the sum of all "mapped" sequence dot products normalized by the total duration of the annotation:

$$s = \frac{\sum_j \langle a_{k(j)}(t), e_j(t) \rangle}{\sum_j \sum_t a_{k(j)}(t)} \tag{6}$$

This score indicates how much are the annotated sequences explained by the estimated sequences.

3.3 Results

We present the results of the sequence estimation for various configuration of our system into Tab. 1. The rows indicate the individual track scores. The last row indicates the average score over the 11 tracks. We first compare the results obtained using the three individual feature sets (MFCC, Spectral Contrast and Pitch Class Profile) with the ones obtained when combining their normalized individual similarity matrices ("Combined features" column). In almost all cases, the results obtained with the combined features (54.8%) are better than the ones obtained with the individual feature sets. We then compare the estimation of the sequences using a 1st order similarity matrix (54.8%) with the estimation using higher-order similarity matrix ("HOS" column). A 2nd order matrix has been used here. For the two cases, we indicate the individual track score and the number of detected segments. On average the use of

Track name	**Number of segments to be detected**	MFCC	Spectral Contrast	PCP	Combined features		HOS	
Alanis Morisette "Head over feet"	*3*	23,2	0,0	**34,4**	**61,9**	*3*	35,7	*2*
Dave Brubeck "Take Five"	*2*	**38,7**	3,9	29,6	**44,1**	*3*	**68,8**	*3*
Moby "Natural Blues"	*4*	14,3	16,1	29,4	24,5	*3*	**33,4**	*3*
Moby "Why does my heart"	*3*	22,5	18,9	**40,6**	**43,8**	*3*	26,6	*3*
Nirvana "Smells like teen spirit"	*4*	28,9	**51,3**	46,0	**73,1**	*3*	28,4	*3*
Oasis "Wonderwall"	*7*	27,2	**41,1**	36,7	36,0	*2*	**52,6**	*3*
Pink Floyd "The Wall"	*3*	40,3	**58,5**	46,3	37,6	*3*	**40,9**	*3*
Pink Martini "Je ne veux pas travailler"	*6*	17,6	32,2	**49,4**	45,6	*3*	35,7	*2*
Beatles "Hard days night"	*4*	27,3	18,4	**52,0**	**84,3**	*3*	76,0	*3*
Beatles "Love Me do"	*4*	62,2	56,5	**86,0**	84,9	*2*	72,1	*2*
Beatles "She loves you"	*4*	26,2	21,3	**30,7**	**66,8**	*3*	44,4	*3*
Average score		**29,8**	**28,9**	**43,7**	**54,8**		**46,8**	

Table 1. Comparison between various configurations of the sequence estimation algorithm on the MPEG-7 “melody repetition” test set.

the HOS makes the score decreases from 54.8% to 46.8%. However, for the Brubeck, Moby “Natural Blues”, Oasis and Pink Floyd tracks, the use of the HOS allows improving the estimation. For the Morisette and Nirvana tracks, the scores decrease. These two tracks have in common a chord progression repeated many times over the track duration. In this case, the use of HOS produces many diagonals in the matrix and masks the real melody repetitions.

4 CONCLUSION AND FUTURE WORKS

In this paper we proposed a system for the automatic estimation of the structure of music tracks using the sequence representation. Three sets of features were used (related to timbre and pitch) and combined into a unique similarity matrix. During an experiment, we showed that the combination of these three sets allows improving the estimation of the structure. We introduced the notion of higher-order similarity matrix, which allows taking into account higher order time repetitions in the computation of the matrix. However, the use of it only brings improvement in few cases. We finally presented a maximum likelihood approach to estimate the structure of the track from the segment detected in the similarity matrix. This approach allows to solve the estimation problem in a global way by looking at the sequences that best explain all observed segments and is much faster than the usual DTW algorithms. Finally, we introduced the MPEG-7 “melody repetition” test set and evaluated our algorithm positively on it.

In our current system, most estimation errors originate from the segment detection part (not from the sequence estimation part). Further works will therefore concentrate on adapting our sequence estimation algorithm to work directly on the similarity matrix, i.e. without requiring a previous detection of the segments. Also, it appeared that the starting time of most detected sequences did not match the annotated one. The annotation tends to start at the beginning of the lyrics. Information about voice presence and beat/ measure positions should certainly allows improving the location of this starting time. Finally, other measures should be studied for the evaluation of the quality of the sequence estimation.

5 ACKNOWLEDGEMENTS

Part of this work was conducted in the context of the French RIAM project Ecoute http://projet-ecoute.ircam.fr/.

6 REFERENCES

[1] S. Abdallah, K. Nolan, M. Sandler, M. Casey, and C. Rhodes. Theory and evaluation of a bayesian music structure extractor. In *ISMIR*, London, UK, 2005.

[2] J.-J. Aucouturier and M. Sandler. Finding repeating patterns in acoustic musical signals: applications for audio thumbnailing. In *AES 22nd Int. Conf. on Virt., Synth. and Ent. Audio*, 2002.

[3] M. Bartsch and G. Wakefield. To catch a chorus: Using chroma-based representations for audio thumbnailing. In *WASPAA*, New Paltz, NY, USA, 2001.

[4] G. Boutard, S. Goldszmidt, and G. Peeters. Browsing inside a music track, the experimentation case study. In *Workshop on LSAS*, Athens, Greece, 2006.

[5] M. Cooper and J. Foote. Automatic music summarization via similarity analysis. In *ISMIR*, Paris, France, 2002.

[6] M. Cooper and J. Foote. Summarizing popular music via structural similarity analysis. In *WASPAA*, New Paltz, NY, USA, 2003.

[7] R. Dannenberg. Pattern discovery techniques for music audio. In *ISMIR*, Paris, France, 2002.

[8] J. Foote. Automatic audio segmentation using a measure of audio novelty. In *ICME*, New York City, NY, USA, 1999.

[9] J. Foote. Visualizing music and audio using self-similarity. In *ACM Int. Conf. on Multimedia*, Orlando, Florida, USA, 1999.

[10] T. Fujishima. Realtime chord recognition of musical sound: a system using common lisp music. In *ICMC*, Bejing, China, 1999.

[11] M. Goto. A chorus-section detecting method for musical audio signals. In *ICASSP*, 2003.

[12] T. Jehan. Perceptual segment clustering for music description and time-axis redundancy cancellation. In *ISMIR*, Barcelona, Spain, 2004.

[13] K. Jensen. Rhythm-based segmentation of popular chinese music. In *ISMIR*, London, UK, 2005.

[14] D. Jiang, L. Lu, H.-J. Zhang, J.-H. Tao, and L.-H. Cai. Music type classification by spectral contrast. In *ICME*, Lausanne Switzerland, 2002.

[15] B. Logan and S. Chu. Music summarization using key phrases. In *ICASSP*, Istanbul, Turkey, 2000.

[16] N. Maddage, C. Xu, M. Kankanhalli, and X. Shao. Content-based music structure analysis with applications to music semantic understanding. In *ACM Int. Conf. on Multimedia*, New York, NY, USA, 2004.

[17] M. Mueller and F. Kurth. Towards structural analysis of audio recordings in the presence of musical variations. *Eurasip Journal on Advances in Signal Processing*, ID 89686, 2007.

[18] J. Paulus and A. Klapuri. Music structure analysis by finding repeated parts. In *ACM Multimedia*, Santa Barbara, CA, 2006.

[19] G. Peeters. Deriving musical structures from signal analysis for music audio summary generation: Sequence and state approach. In *LNCS 2771*. Springer-Verlag, 2004.

[20] G. Peeters, A. Laburthe, and X. Rodet. Toward automatic music audio summary generation from signal analysis. In *ISMIR*, Paris, France, 2002.