Sequence representation of Music Structure using Higher-Order Similarity Matrix and Maximum likelihood approach

G. Peeters peeters@ircam.fr

IRCAM (Sound Analysis/Synthesis Team) - CNRS (STMS)

supported by the RIAM Ecoute Project

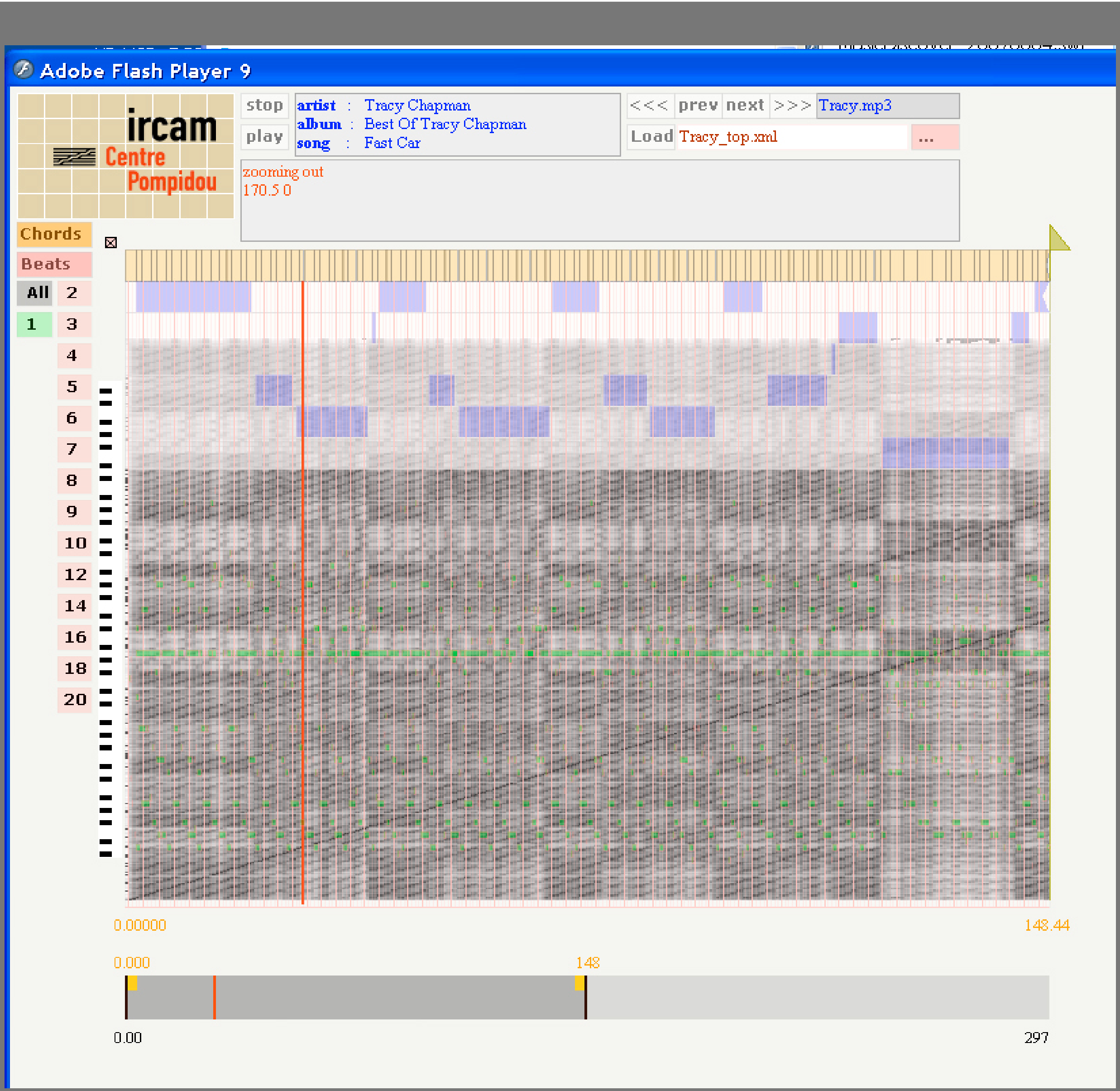
Introduction

- Music Structure Discovery (MSD)
  - estimate automatically the structure of a music track by analyzing its audio signal

- Applications
  - active music listening (intra-document browsing)
  - acoustic browsing of music catalogues (audio summary)
  - music creation (automatic segmentation, music mosaicing)
  - media compression
  - automatic music analysis

Introduction

- Music listening tool



Adobe Flash Player 9

artist : Tracy Chapman
album : Best Of Tracy Chapman
song : Fast Car

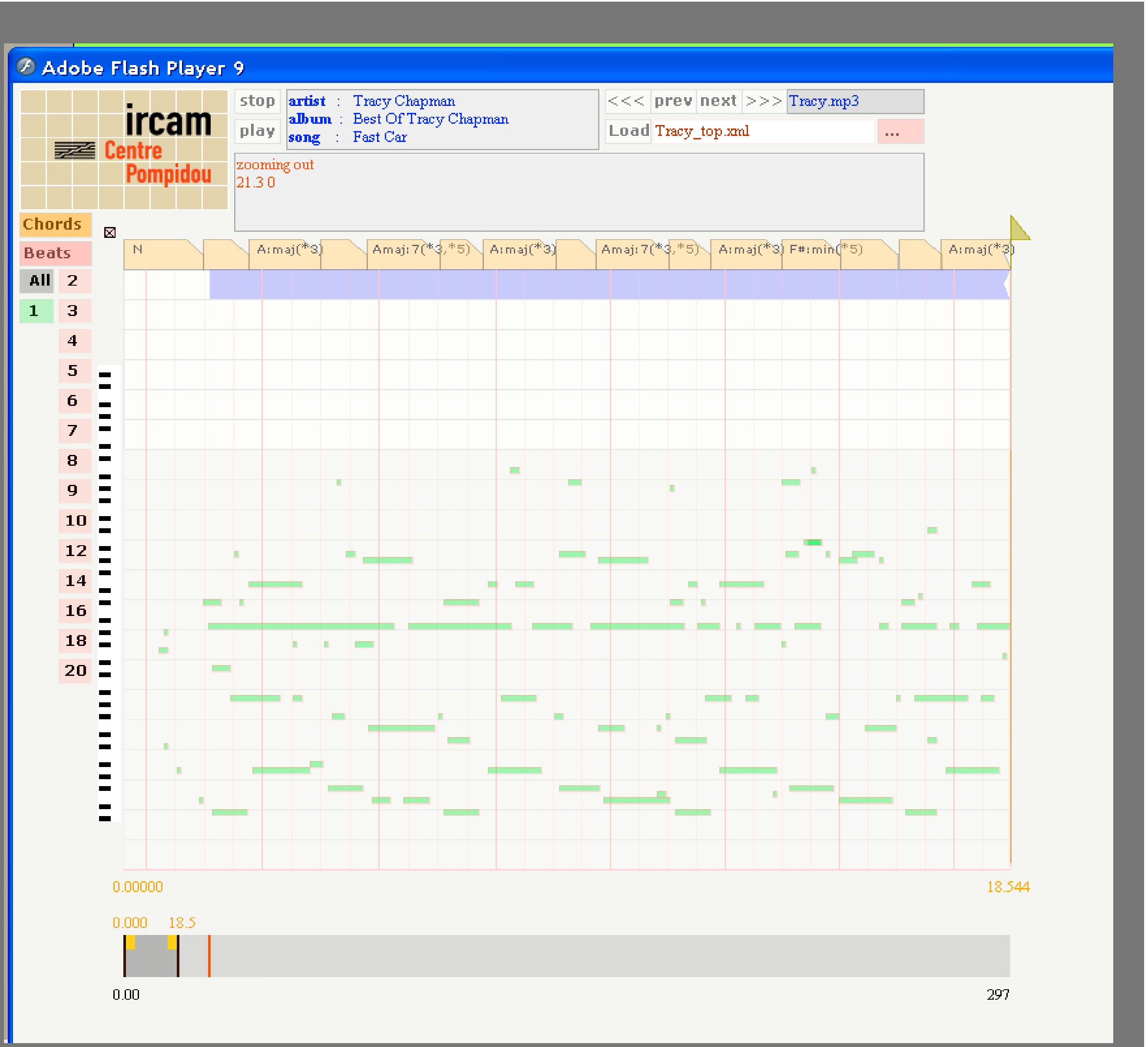
Tracy.mp3

Load Tracy_top.xml

zooming out
170.5 0

Introduction

- Music listening tool



artist : Tracy Chapman
album : Best Of Tracy Chapman
song : Fast Car

Tracy.mp3

Load Tracy_top.xml

zooming out
21.3 0

Introduction

- Music Structure Discovery (MSD)
  - estimate automatically the structure of a music track by analyzing its audio signal

- Applications
  - active music listening (intra-document browsing)
  - acoustic browsing of music catalogues (audio summary)
  - music creation (automatic segmentation, music mosaicing)
  - media compression
  - automatic music analysis

- MSD algorithms
  - extract a set of features
    - choice of the features determines what kind of repetitions can be observed (timbre, melody, rhythm repetitions ?)
  - search for repetitions in the set of extracted features
    - we can only detect repetitions, not evolution of patterns

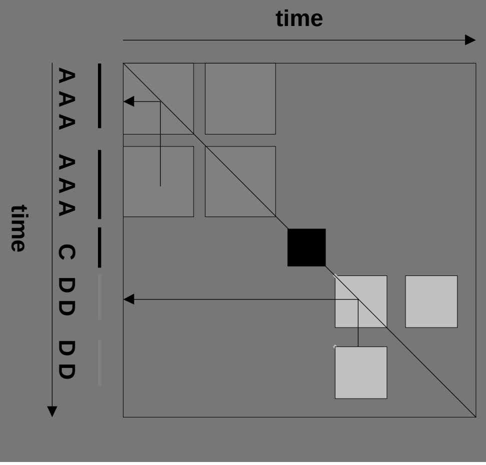
- Visualization
    - recurrence plot, similarity matrix

History

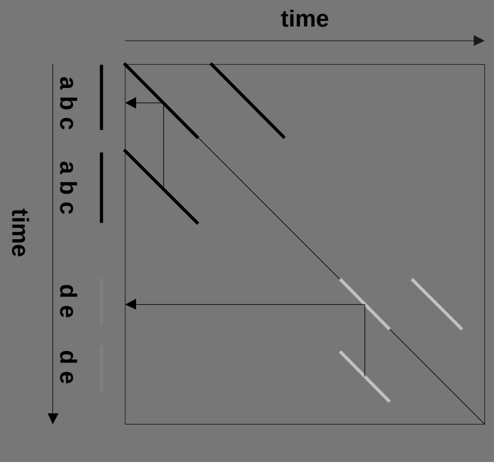
- 2002 ISMIR Peeters, Laburthe, Rodet "Toward audio summary generation …"
  - state representation by Segmentation + K-means + HMM
  - dynamic features -> modulation spectrum ?
- 2002 Report Laburthe, Peeters "Sequence detection by Factor Oracle"
- 2003 ICMC Peeters "Sequence and state representation"
  - state representation …
  - sequence representation by Similarity Matrix, 2D Structuring Filter + Matching
- 2003 MPEG-7 Test set for state and sequence representation
- 2005 Report Wronecki, Peeters "DTW for sequence detection"
- 2005 Report Mislin, Peeters "Audio summary for contemporary music"
- 2006 LSAS Boutard, Goldsmith, Peeters "Browsing inside a music track"
- => 2007 ISMIR Peeters "Sequence representation …"
  - new method for sequence representation

State / Sequence representation

- State representation
  - music track = succession of parts called states, each time of a music track belongs to a specific state
  - state= a set of contiguous times which contains similar acoustical information
  - all the times of a music track belongs to a state
  - a state does not need to be repeated later in the track
  - related to the notion of parts (intro, verse, chorus, bridge) in popular music
  - algorithms: segmentation (novelty measure), partitional, agglomerative or spectral clustering algorithms



- Sequence representation
  - there exists sequences of time in the music track that are repeated over the track
  - sequence= a set of successive times, which is similar to another set of of successive times
  - all the times of a music track do not belong necessarily to a sequence
  - sequence need at least to be repeated twice
  - related to the notion of melody, chord progression in popular music
  - algorithms: …



Sequence representation

- Related works
  - detect the most representative audio extract from the similarity matrix
    -> in order to create to create a thumbnail
  - estimating the actual sequences ?
    - Dynamic Time Warping
    - Recent approaches: DTW + hierarchical approach

    - -> very heavy in computation time !

Feature extraction and similarity matrix

- Similarity can come from either
  - timbre,
  - harmony or
  - rhythm
- Feature extraction: 3 sets
  - 13 MFC coefficients (excluding the 0th/DC-component coefficient)
  - 12 Spectral Contrast coefficients (spectral contrast + spectral valley)
  - 12 Pitch Class Profile coefficients

  - window= 80ms, hop size=40ms

Feature extraction and similarity matrix

- Similarity can come from either
  - timbre,
  - harmony or
  - rhythm
- Feature extraction: 3 sets
  - 13 MFC coefficients (excluding the 0th/DC-component coefficient)
  - 12 Spectral Contrast coefficients (spectral contrast + spectral valley)
  - 12 Pitch Class Profile coefficients
  - window 80ms, hop size=40ms
- Modeling over time: mean value of 4s/ 500ms
- Principal Component Analysis applied to the 3 sets separately
  - keep components >10% max variance
- Three similarity/distance matrices
  - using Euclidean distance
  - normalization between [0,1]
- Combining the three matrices:

$$S(t_x,t_y) = S_{mfcc}(t_x,t_y) + S_{scc}(t_x,t_y) + S_{pcp}(t_x,t_y)$$

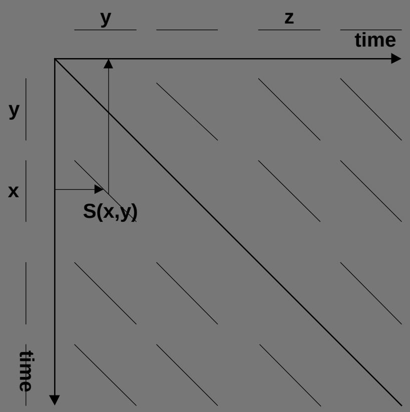
Higher order similarity matrix

- Principle:
  - Transitivity rule:
    - if $o(t_z)=o(t_x)$ and $o(t_z)=o(t_y)$ => then $o(t_y)=o(t_x)$
    - can be hidden in the matrix
  - Goal of HOS matrix
    - recover the missing values, emphasize the repetitions

Higher order similarity matrix

- Principle:
  - Transitivity rule:
    - if $o(t_z)=o(t_x)$ and $o(t_z)=o(t_y)$ then $o(t_y)=o(t_x)$
    - can be hidden in the matrix
  - Goal of HOS matrix
    - recover the missing values, emphasize the repetitions
  - Second order similarity matrix

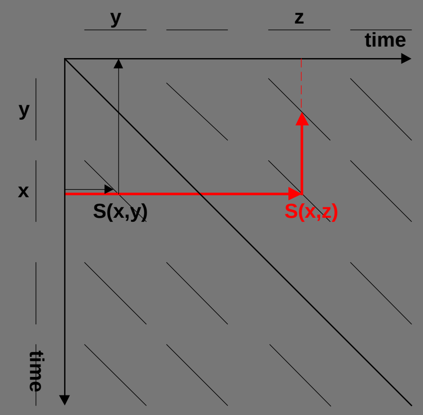
$$S_2(t_x, t_y) = \int_{t_z} S(t_x, t_z) S(t_z, t_y) dt_z$$



Higher order similarity matrix

- Principle:
  - Transitivity rule:
    - if $o(t_z)=o(t_x)$ and $o(t_z)=o(t_y)$ then $o(t_y)=o(t_x)$
    - can be hidden in the matrix
  - Goal of HOS matrix
    - recover the missing values, emphasize the repetitions
  - Second order similarity matrix

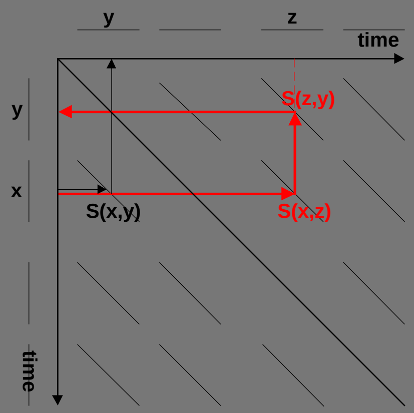
$$S_2(t_x,t_y)=\int_{t_z} S(t_x,t_z)S(t_z,t_y)dt_z$$



Higher order similarity matrix

- Principle:
  - Transitivity rule:
    - if $o(t_z)=o(t_x)$ and $o(t_z)=o(t_y)$ then $o(t_y)=o(t_x)$
    - can be hidden in the matrix
  - Goal of HOS matrix
    - recover the missing values, emphasize the repetitions
  - Second order similarity matrix

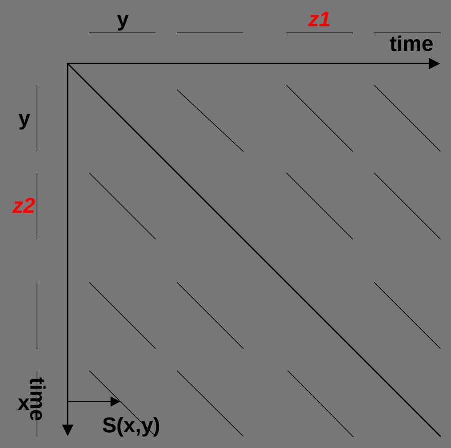
$$S_2(t_x, t_y) = \int_{t_z} S(t_x, t_z) S(t_z, t_y) dt_z$$



Higher order similarity matrix

- Principle:
  - Transitivity rule:
    - if $o(t_z)=o(t_x)$ and $o(t_z)=o(t_y)$ then $o(t_y)=o(t_x)$
    - can be hidden in the matrix
  - Goal of HOS matrix
    - recover the missing values, emphasize the repetitions
  - Second order similarity matrix
  - Third order similarity matrix

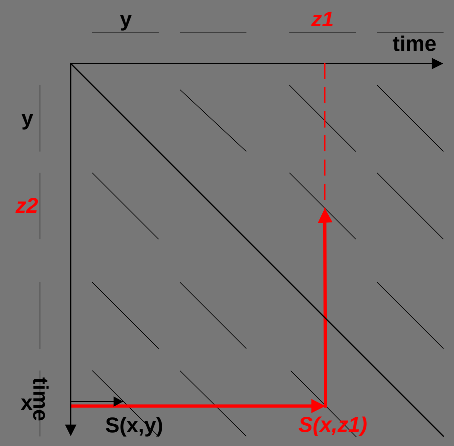
$$S_3(t_x, t_y) = \int_{t_{z1}} \int_{t_{z2}} S(t_x, t_{z1}) S(t_{z1}, t_{z2}) S(t_{z2}, t_y) dt_{z1} dt_{z2}$$



Higher order similarity matrix

- Principle:
  - Transitivity rule:
    - if $o(t_z)=o(t_x)$ and $o(t_z)=o(t_y)$ then $o(t_y)=o(t_x)$
    - can be hidden in the matrix
  - Goal of HOS matrix
    - recover the missing values, emphasize the repetitions
  - Second order similarity matrix
  - Third order similarity matrix

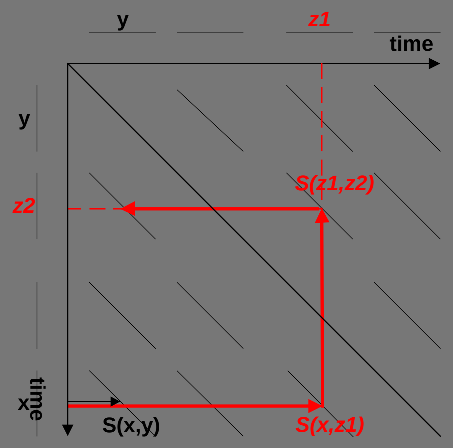
$$S_3(t_x, t_y) = \int_{t_{z1}} \int_{t_{z2}} S(t_x, t_{z1}) S(t_{z1}, t_{z2}) S(t_{z2}, t_y) dt_{z1} dt_{z2}$$



Higher order similarity matrix

- Principle:
  - Transitivity rule:
    - if o(t_z)=o(t_x) and o(t_z)=o(t_y) then o(t_y)=o(t_x)
    - can be hidden in the matrix
  - Goal of HOS matrix
    - recover the missing values, emphasize the repetitions
  - Second order similarity matrix
  - Third order similarity matrix

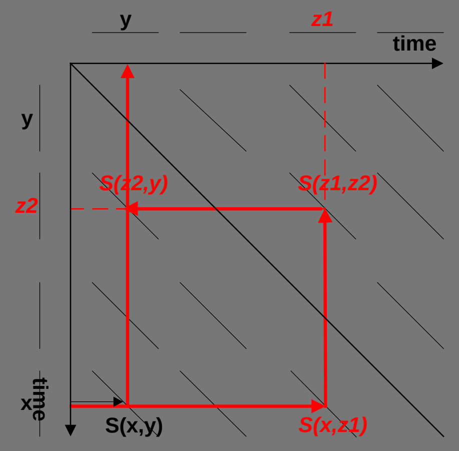
$$S_3(t_x,t_y) = \int_{t_{z1}} \int_{t_{z2}} S(t_x,t_{z1}) S(t_{z1},t_{z2}) S(t_{z2},t_y) dt_{z1} dt_{z2}$$



Higher order similarity matrix

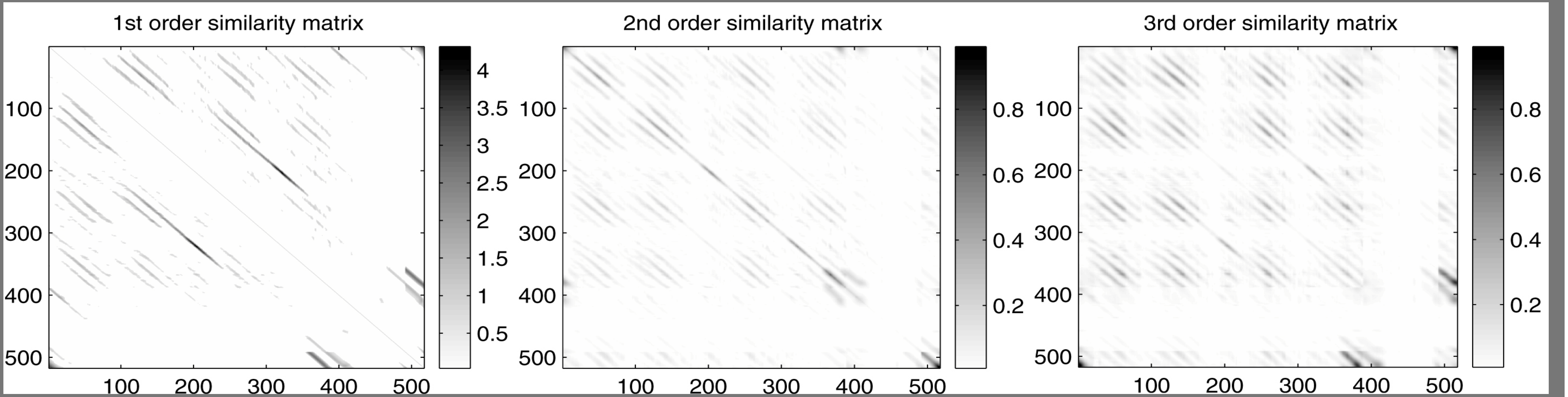
- Principle:
  - Transitivity rule:
    - if o(t_z)=o(t_x) and o(t_z)=o(t_y) then o(t_y)=o(t_x)
    - can be hidden in the matrix
  - Goal of HOS matrix
    - recover the missing values, emphasize the repetitions
  - Second order similarity matrix
  - Third order similarity matrix

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



Higher order similarity matrix

Example on “She Loves You” from The Beatles



Sequence representation

Introduction

- Extract from the HOS matrix the sequence representation
  - 1) in the matrix -> detect the diagonals (lines) -> form segments
  - 2) from the detected segments -> estimate the sequences
-> sequence representation

Sequence representation

Introduction

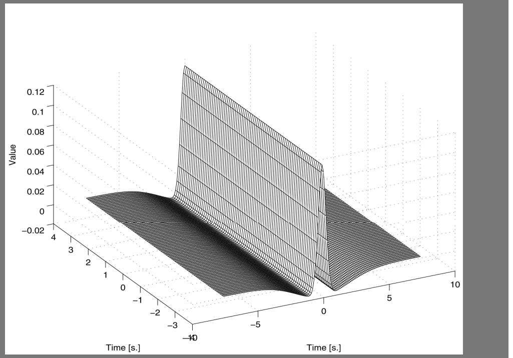
- Extract from the HOS matrix the sequence representation
  - 1) in the matrix -> detect the diagonals (lines) -> form segments
  - 2) from the detected segments -> estimate the sequences
-> sequence representation

- Definitions:
  - Diagonal (line):
    - possibly discontinuous set of points in the similarity (lag) matrix
  - Segment:
    - set of successive (continuous) times defined by a strating and ending times. Diagonal -> defines two segments (original + repetition)
  - Sequence:
    - set of segments representing similar information occuring at various times. Sequence = “mother” segment (most representative one) + set of times (indicates at which times the “mother” segment is instanciated).
  - Sequence representation:
    - set of sequences

Sequence representation

1. Matrix -> Diagonals (lines) -> Segments detection

- Reinforce the diagonals elements, removing non-diagonal elements
  - Matrix filtering
    - convert to lag-matrix $l_{ij} = t_i - t_j$
    - horizontal high-pass filter:
      combination of two opposed sign gaussian functions
    - vertical low-pass filter:
      simple averaging filter (length=8s)



Sequence representation

1. Matrix -> Diagonals (lines) -> Segments detection

- Reinforce the diagonals elements, removing non-diagonal elements
  - Matrix filtering
    - convert to lag-matrix $l_{ij} = t_i - t_j$
    - horizontal high-pass filter:
      combination of two opposed sign gaussian functions
    - vertical low-pass filter:
      simple averaging filter (length=8s)

- Segment detection
  - Using of Goto [ICASSP 2003] method
    - (-) does not allow to detect repetitions with time variations
    - (+) fast and most of the time reliable
    - Method:
      - peak detection in the summation over the time-axis of the lag-matrix
      - for each detected peak, analysis of the constant-lag time-segment

Sequence estimation

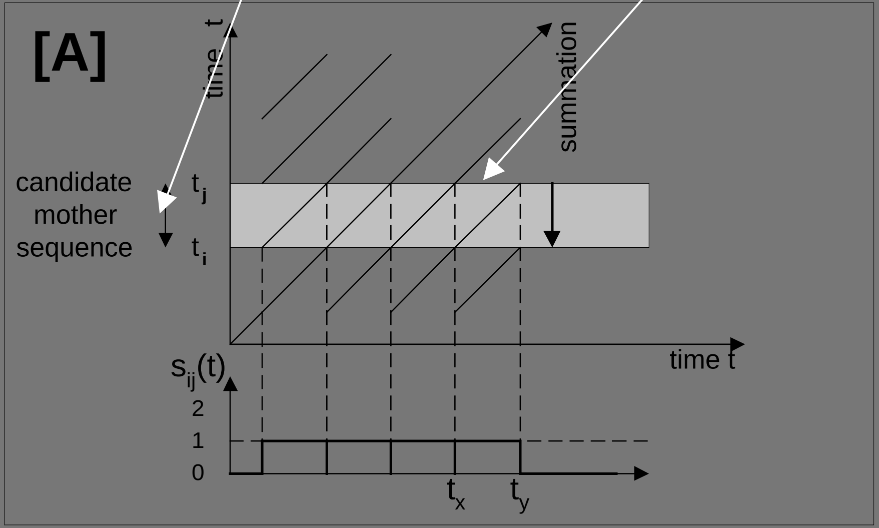
2) Segments -> Sequence estimation

- Goal ?
  - represent all the segments detected in the matrix
    using the smallest possible set of sequences (mother segments and repetitions times)
- How ?
  - For each candidate mother segment
    -> measure how well it would explain the observed segments

Sequence estimation

2) Segments -> Sequence estimation

- Goal ?
  - represent all the segments detected in the matrix
    using the smallest possible set of sequences (mother segments and repetitions times)
- How ?
  - For each candidate mother segment
    -> measure how well it would explain the observed segments
  - segment similarity matrix: $S_{seg}(t_i,t_j)$
  - m_{ij} candidate mother segment defined by T_i T_j
    -> m_{ij} defines a corridor in the segment similarity matrix



Sequence estimation

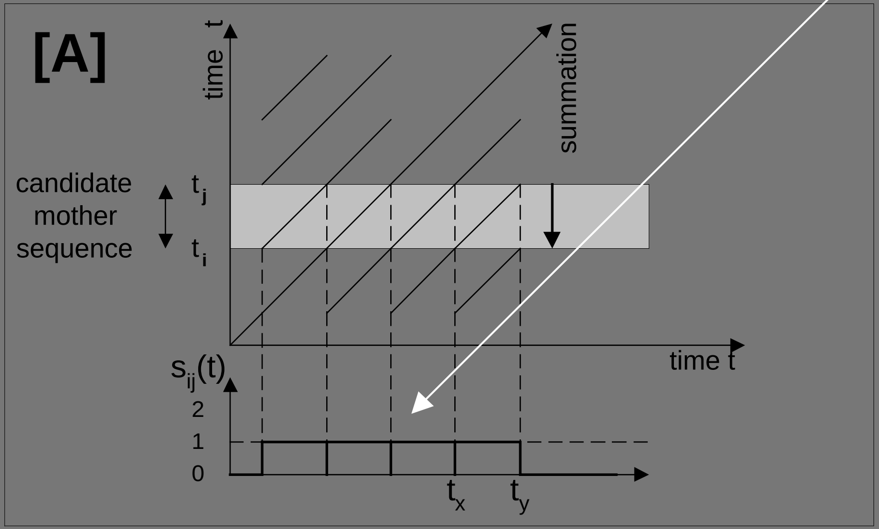
2) Segments -> Sequence estimation

- summation over the length of the corridor

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

- summation over the width of the corridor
  - indicates how many segments cross simultaneously the corridor

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



Sequence estimation

2) Segments -> Sequence estimation

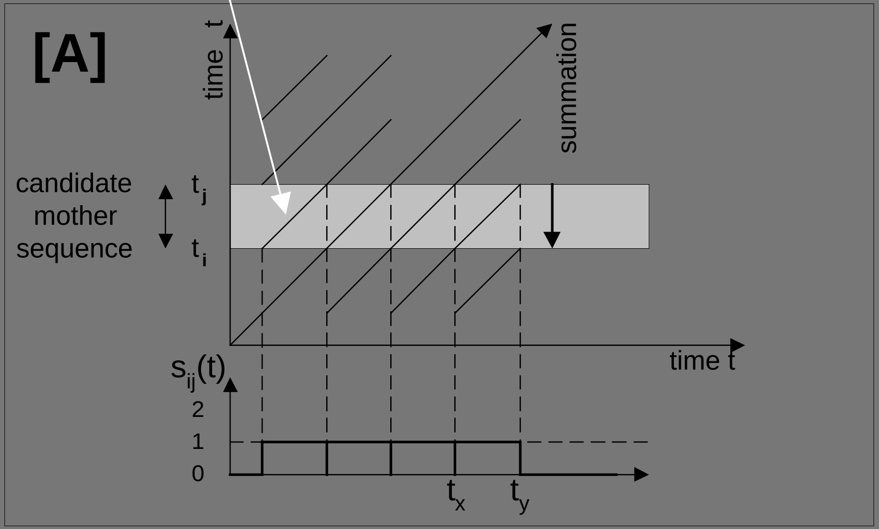
- summation over the length of the corridor →

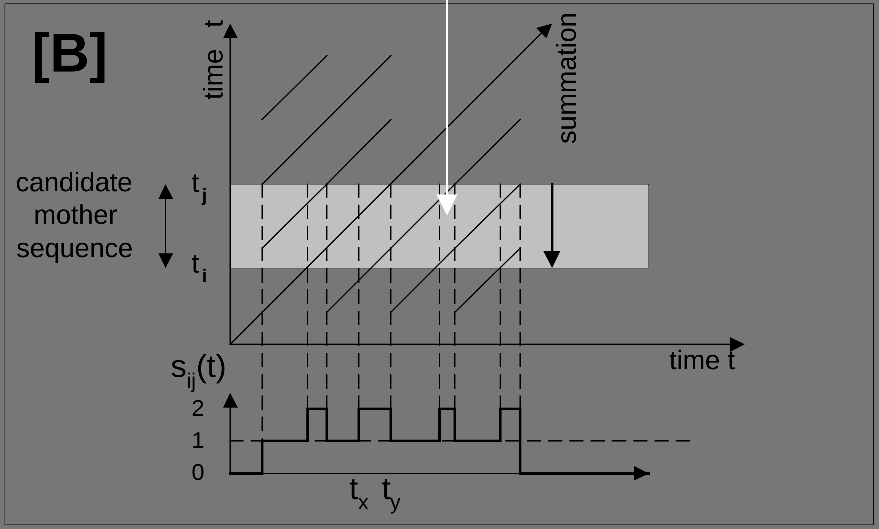
$$\sigma(\tau) = \sum_{t=1}^{T} S_{seg}(\tau, t)$$

- summation over the width of the corridor ↓
  - indicates how many segments cross simultaneously the corridor

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

- If one segment crosses the corridor simultaneously … If two segments …





Sequence estimation

2) Segments -> Sequence estimation

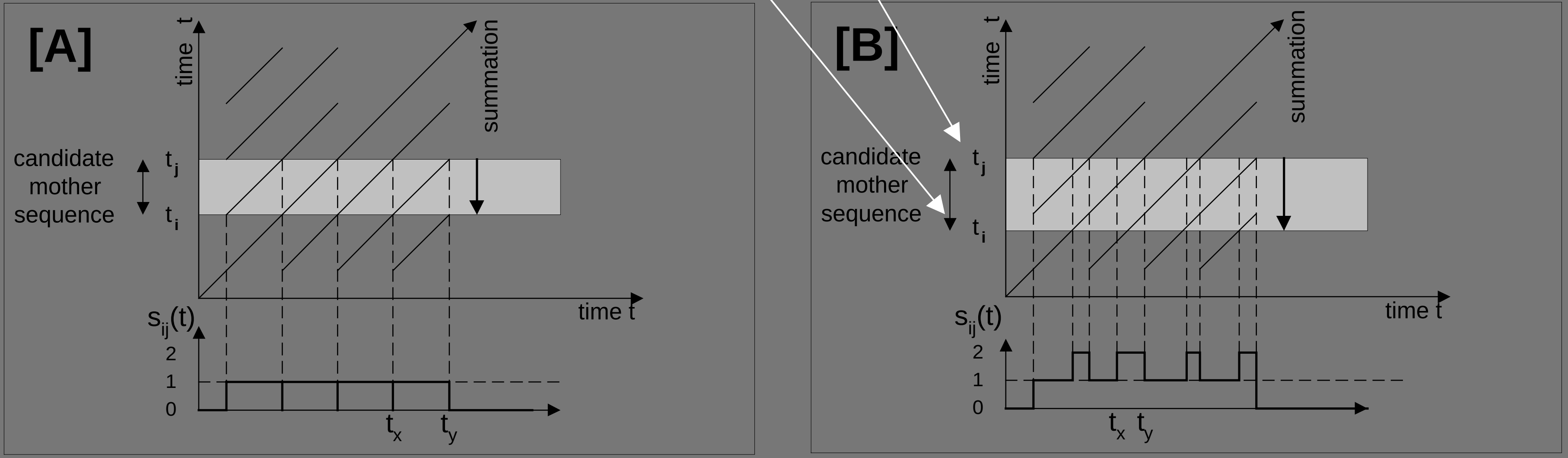
- summation over the length of the corridor →

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

- summation over the width of the corridor ↓
  - indicates how many segments cross simultaneously the corridor

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

- If one segment crosses the corridor simultaneously … If two segments …
- First condition:
  $s_{ij}(t)$ must be <=1 -> change T_i and T_j to achieve that



Sequence estimation

2) Segments -> Sequence estimation

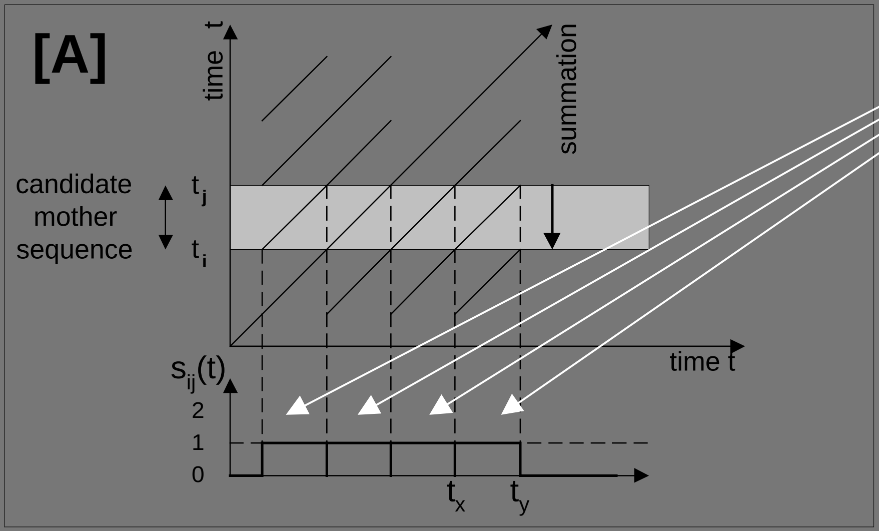
- summation over the length of the corridor →

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

- summation over the width of the corridor ↓
  - indicates how many segments cross simultaneously the corridor

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

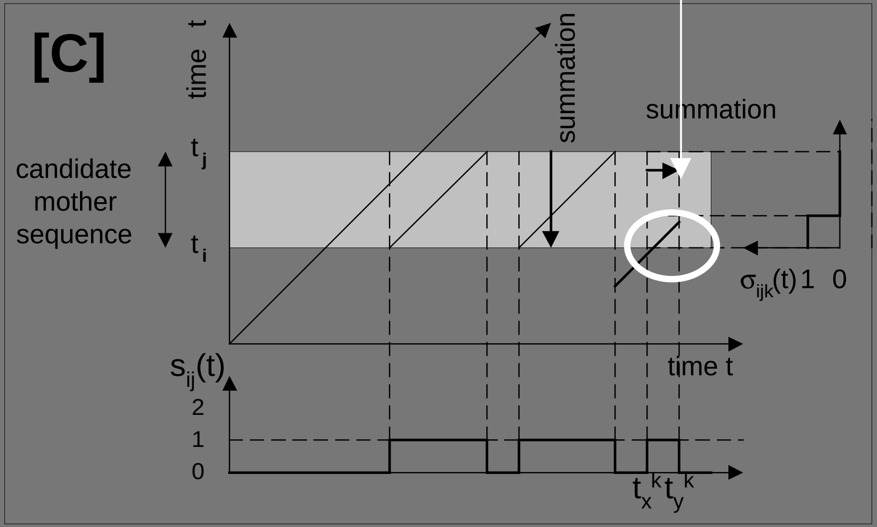
- If one segment crosses the corridor simultaneously ... If two segments …
- First condition: $s_{ij}(t)$ must be <=1 -> reduce T_i and T_j to achieve that
- $s_{ij}(t)$ is used for segmentation -> k segments $[t_x^k, t_y^k]$

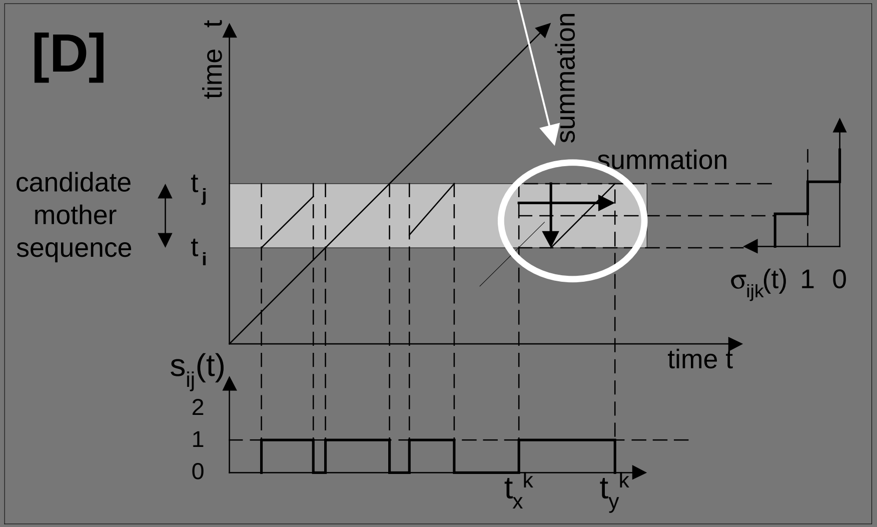


Sequence estimation

2) Segments -> Sequence estimation

- Second condition:
  - $s_{ij}(t)$ <=1 is not enough to guarantee that $[t_x^k, t_y^k]$ is an instantiation of the mother segment T_i T_j
  - could also be a part (beginning or ending) of another sequence
  - could also be the succession of two non-overlapping segments





Sequence estimation

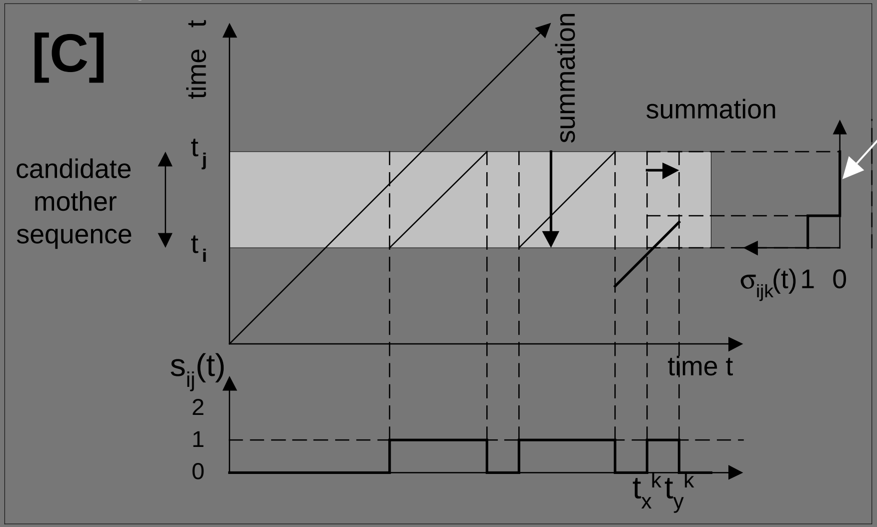
2) Segments -> Sequence estimation

- Second condition:
  - $s_{ij}(t)$ <=1 is not enough to guarantee that $[t_x^k, t_y^k]$ is an instantiation of the mother segment T_i T_j
  - could also be a part (beginning or ending) of another sequence
  - could also be the succession of two non-overlapping segments

- second score specific to each interval k:

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

- Second condition:
  $s_{ijk}(t)$ must be =1 for all k -> reduce T_i and T_j to achieve that



Sequence estimation

2) Segments -> Sequence estimation

- Best fit approach:
  - T_i and T_j should be modified until $s_{ij}(t)<=1$ and $s_{ijk}(t)=1$ for all k
  - But this could lead to unnecessary corridor width reduction
  - best fit approach between reduction of errors and reduction of corridor-width
    - SEE PAPER FOR DETAILS

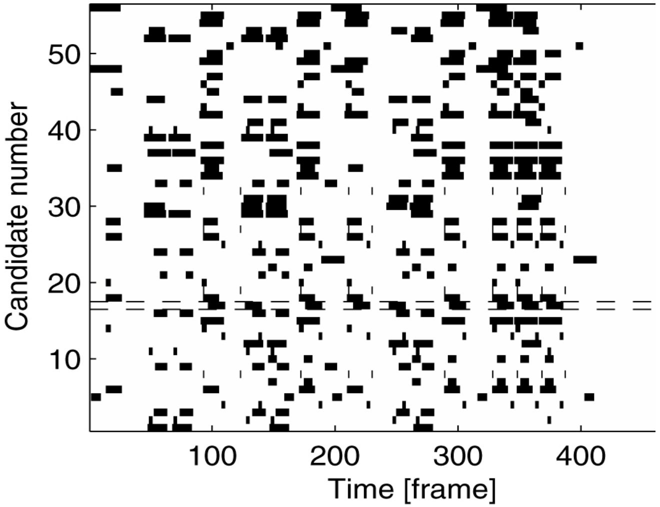
- How is the corridor reduced ?
  - increase T_i or decrease T_j ?
  - each interval k may require a different solution
    - vote process: each interval vote either to increase T_i or decrease T_j

- Score computation
  - a score is attribute to each candidate mother segment
    - represents the likelihood that this mother segment explains the observed segments
    - score ? defined as the sum of the lengths of all explained segment $[t_x^k, t_y^k]$
  - choose the condidate mother segment with the maximum likelihood

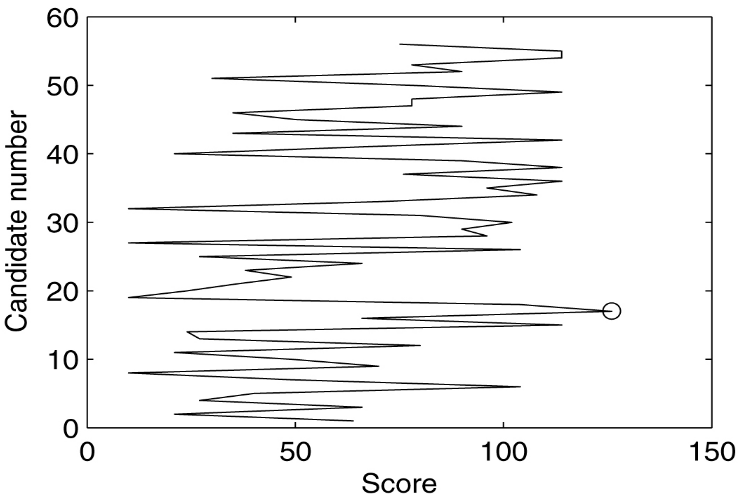
Sequence estimation

2) Segments -> Sequence estimation

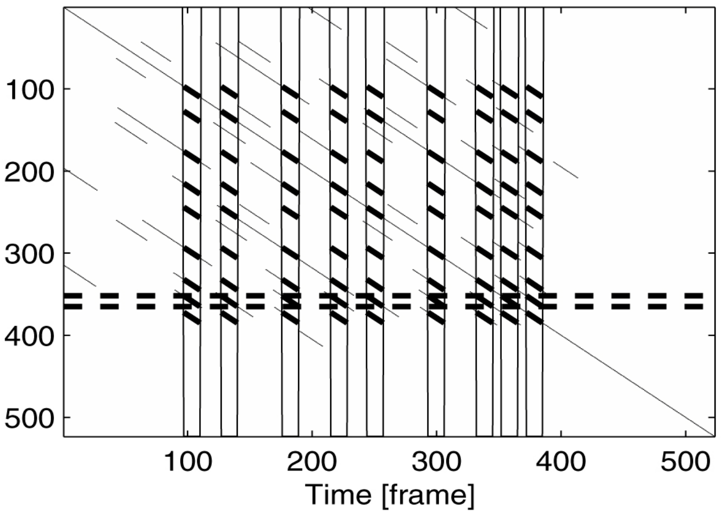
Example on “She Loves You” from The Beatles



For each candidate mother sequence, the temporal segments explained



Score of each candidate mother sequence



Segment similarity matrix + Final corridor width and segments expained by the winning candidate mother sequence

Sequence estimation

2) Segments -> Sequence estimation

- Best fit approach:
  - T_i and T_j should be modified until $s_{ij}(t)<=1$ and $s_{ijk}(t)=1$ for all k
  - But this could lead to unnecessary corridor width reduction
  - best fit approach between reduction of errors and reduction of corridor-width
    - SEE PAPER FOR DETAILS
- How is the corridor reduced ?
  - increase T_i or decrease T_j ?
  - each interval k may require a different solution
    - vote process: each interval vote either to increase T_i or decrease T_j
- Score computation
  - a score is attribute to each candidate mother segment
    - represents the likelihood that this mother segment explains the observed segments
    - score ? defined as the sum of the lengths of all explained segment $[t_x^k, t_y^k]$
  - choose the condidate mother segment with the maximum likelihood
- Segment cancellation
  - cancel the values of the segment similarity matrix inside the corridor of the selected mother segment
- The process is repeated for the detection of the next mother segment
- T_i and T_j can be any starting and ending time but in practice T_i and T_j are chosen among the detected segments

Evaluation

- Test set
  - MPEG-7 Test Set (2003)
    - state annotations
    - sequence annotations: 11 tracks (up to 7 melodies)
  - QMUL test set ?

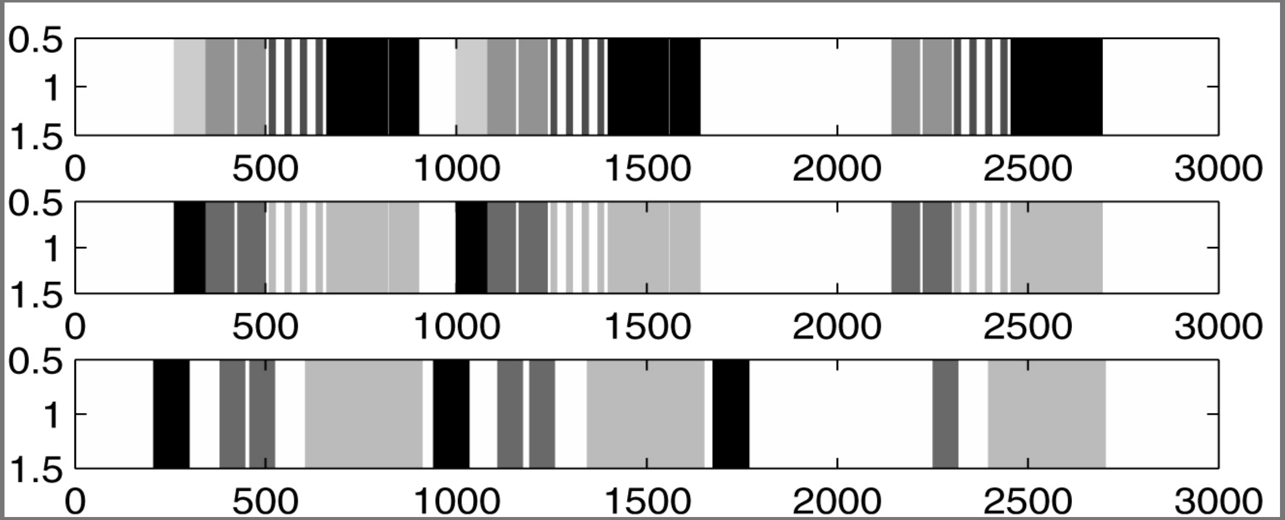
- Performance measure
  - ABCD -> AB CD or A BCD or AB- -BC- -CD
  - number of labels may differ
    - annotation tends to split melodies into sub-melodies (according to lyrics)
    - estimation tends to merge successive repeated melodies into a single one
    - allow mapping several annotated sequences to a unique estimated sequence
  - mapping between annotated a(i) and estimated labels e(j): k(j)=i
    - based on correlation between $a_i(t)$ and $e_j(t)$

- Score computation

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

Evaluation

Example on “Smells like teen spirit” from Nirvana



Annotated

Mapped annotations

Detected

Evaluation

- Results:
  - choice of features:
    - best results with combined features

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

Evaluation

- Results:
  - choice of features:
    - best results with combined features
  - choice of the method (1st order SM or 2nd order SM)
    - only improvement for
Brubeck,
Moby “Natural Blues”,
Oasis,
Pink Floyd
    - worst for
Morisette,
Nirvana
    - Why ? always the same chord progression -> to many segments

Track name	Number of segments to be detected				Combined features		HOS	
Alanis Morisette "Head over feet"	*3*				**61,9**	*3*	35,7	*2*
Dave Brubeck "Take Five"	*2*				44,1	*3*	**68,8**	*3*
Moby "Natural Blues"	*4*				24,5	*3*	**33,4**	*3*
Moby "Why does my heart"	*3*				**43,8**	*3*	26,6	*3*
Nirvana "Smells like teen spirit"	*4*				**73,1**	*3*	28,4	*3*
Oasis "Wonderwall"	*7*				36,0	*2*	**52,6**	*3*
Pink Floyd "The Wall"	*3*				37,6	*3*	**40,9**	*3*
Pink Martini "Je ne veux pas travailler"	*6*				45,6	*3*	35,7	*2*
Beatles "Hard days night"	*4*				**84,3**	*3*	76,0	*3*
Beatles "Love Me do"	*4*				**84,9**	*2*	72,1	*2*
Beatles "She loves you"	*4*				**66,8**	*3*	44,4	*3*
Average score					**54,8**		**46,8**	

Conclusion and future works

- Conclusion: we have proposed
  - Simultaneous use of several similarity matrix (timbre, harmony)
    - -> Improve the results
  - High-Order Similarity Matrix
    - -> only brings improvement in few cases
  - New method for sequence representation
    which allows to solve the problem on a global way
    - -> robust and much faster than usual DTW approach

Conclusion and future works

- Conclusion: we have proposed
  - Simultaneous use of several similarity matrix (timbre, harmony)
    - -> Improve the results
  - High-Order Similarity Matrix
    - -> only brings improvement in few cases
  - New method for sequence representation which allows to solve the problem on a global way
    - -> robust and much faster than usual DTW approach

- Future works
  - Most errors come from the segment detection (not from the sequence estimation)
    - -> adapt the sequence estimation algorithm to work directly on the similarity matrix (no need for segment estimation)
  - The detected starting time of segments do not match the annotated one
    - introduce information about voice presence, beat/ measure positions
  - Study other evaluation measures