Audio Structure Analysis

Meinard Müller, Christof Weiß
International Audio Laboratories Erlangen
meinard.mueller@audiolabs-erlangen.de, christof.weiss@audiolabs-erlangen.de
Book: Fundamentals of Music Processing

Meinard Müller
Fundamentals of Music Processing
Audio, Analysis, Algorithms, Applications
483 p., 249 illus., hardcover
ISBN: 978-3-319-21944-8
Springer, 2015

Accompanying website:
www.music-processing.de
# Book: Fundamentals of Music Processing

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Music Processing Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Music Representations</td>
</tr>
<tr>
<td>2</td>
<td>Fourier Analysis of Signals</td>
</tr>
<tr>
<td>3</td>
<td>Music Synchronization</td>
</tr>
<tr>
<td>4</td>
<td>Music Structure Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Chord Recognition</td>
</tr>
<tr>
<td>6</td>
<td>Tempo and Beat Tracking</td>
</tr>
<tr>
<td>7</td>
<td>Content-Based Audio Retrieval</td>
</tr>
<tr>
<td>8</td>
<td>Musically Informed Audio Decomposition</td>
</tr>
</tbody>
</table>

Meinard Müller  
Fundamentals of Music Processing  
Audio, Analysis, Algorithms, Applications  
483 p., 249 illus., hardcover  
ISBN: 978-3-319-21944-8  
Springer, 2015  

Accompanying website:  
www.music-processing.de
## Book: Fundamentals of Music Processing

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Music Processing Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Music Representations</td>
</tr>
<tr>
<td>2</td>
<td>Fourier Analysis of Signals</td>
</tr>
<tr>
<td>3</td>
<td>Music Synchronization</td>
</tr>
<tr>
<td>4</td>
<td>Music Structure Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Chord Recognition</td>
</tr>
<tr>
<td>6</td>
<td>Tempo and Beat Tracking</td>
</tr>
<tr>
<td>7</td>
<td>Content-Based Audio Retrieval</td>
</tr>
<tr>
<td>8</td>
<td>Musically Informed Audio Decomposition</td>
</tr>
</tbody>
</table>

Meinard Müller  
Fundamentals of Music Processing  
Audio, Analysis, Algorithms, Applications  
483 p., 249 illus., hardcover  
ISBN: 978-3-319-21944-8  
Springer, 2015

Accompanying website:  
www.music-processing.de
Chapter 4: Music Structure Analysis

4.1 General Principles
4.2 Self-Similarity Matrices
4.3 Audio Thumbnailing
4.4 Novelty-Based Segmentation
4.5 Evaluation
4.6 Further Notes

In Chapter 4, we address a central and well-researched area within MIR known as music structure analysis. Given a music recording, the objective is to identify important structural elements and to temporally segment the recording according to these elements. Within this scenario, we discuss fundamental segmentation principles based on repetitions, homogeneity, and novelty—principles that also apply to other types of multimedia beyond music. As an important technical tool, we study in detail the concept of self-similarity matrices and discuss their structural properties. Finally, we briefly touch the topic of evaluation, introducing the notions of precision, recall, and F-measure.
Music Structure Analysis

Example: Zager & Evans “In The Year 2525”
Music Structure Analysis

Example: Zager & Evans “In The Year 2525”
Music Structure Analysis

Example: Zager & Evans “In The Year 2525”
Music Structure Analysis

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Music Structure Analysis

Example: Folk Song Field Recording
(Nederlandse Liederenbank)
Music Structure Analysis

Example: Weber, Song (No. 4) from “Der Freischütz”
Music Structure Analysis

**General goal:** Divide an audio recording into temporal segments corresponding to musical parts and group these segments into musically meaningful categories.

**Examples:**
- Stanzas of a folk song
- Intro, verse, chorus, bridge, outro sections of a pop song
- Exposition, development, recapitulation, coda of a sonata
- Musical form ABACADA … of a rondo
Music Structure Analysis

**General goal:** Divide an audio recording into temporal segments corresponding to musical parts and group these segments into musically meaningful categories.

**Challenge:** There are many different principles for creating relationships that form the basis for the musical structure.

- **Homogeneity:** Consistency in tempo, instrumentation, key, ...
- **Novelty:** Sudden changes, surprising elements ...
- **Repetition:** Repeating themes, motives, rhythmic patterns,
Music Structure Analysis

Novelty

Homogeneity

Repetition
Overview

- Introduction
- Feature Representations
- Self-Similarity Matrices
- Novelty-Based Segmentation

Thanks:

- Clausen, Ewert, Kurth, Grohganz, ...
- Dannenberg, Goto
- Grosche, Jiang
- Paulus, Klapuri
- Peeters, Kaiser, ...
- Serra, Gómez, ...
- Smith, Fujinaga, ...
- Wiering, ...
- Wand, Sunkel, Jansen
- ...

Overview

- Introduction
- Feature Representations
- Self-Similarity Matrices
- Novelty-Based Segmentation

Thanks:
- Clausen, Ewert, Kurth, Grohganz, …
- Dannenberg, Goto
- Grosche, Jiang
- Paulus, Klapuri
- Peeters, Kaiser, …
- Serra, Gómez, …
- Smith, Fujinaga, …
- Wiering, …
- Wand, Sunkel, Jansen
- …
Feature Representation

**General goal:** Convert an audio recording into a mid-level representation that captures certain musical properties while suppressing other properties.

- Timbre / Instrumentation
- Tempo / Rhythm
- Pitch / Harmony
Feature Representation

**General goal:** Convert an audio recording into a mid-level representation that captures certain musical properties while suppressing other properties.

- Timbre / Instrumentation
- Tempo / Rhythm
- Pitch / Harmony
Feature Representation

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Feature Representation

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Feature Representation

**Example:** Brahms Hungarian Dance No. 5 (Ormandy)

Chroma (Harmony)

Feature extraction
Feature Representation

**Example:** Brahms Hungarian Dance No. 5 (Ormandy)

Chroma (Harmony)

- G minor
- G major
- G minor
Overview

- Introduction
- Feature Representations
- Self-Similarity Matrices
- Novelty-Based Segmentation
Self-Similarity Matrix (SSM)

**General idea:** Compare each element of the feature sequence with each other element of the feature sequence based on a suitable similarity measure.

→ Quadratic self-similarity matrix
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

**Example:** Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

**Example:** Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

**Example:** Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)
Self-Similarity Matrix (SSM)

Example: Brahms Hungarian Dance No. 5 (Ormandy)

Idealized SSM
Self-Similarity Matrix (SSM)

**Example:** Brahms Hungarian Dance No. 5 (Ormandy)

**Blocks:** Homogeneity

**Paths:** Repetition

**Corners:** Novelty
SSM Enhancement

Challenge: Presence of musical variations

- Fragmented paths and gaps
- Paths of poor quality
- Regions of constant (low) cost
- Curved paths

Idea: Enhancement of path structure
SSM Enhancement

Path Enhancement
SSM Enhancement

Path Enhancement

- Diagonal smoothing
SSM Enhancement

Path Enhancement

- Diagonal smoothing
- Multiple filtering
SSM Enhancement

Path Enhancement

- Diagonal smoothing
- Multiple filtering
- Thresholding (relative)
- Scaling & penalty
SSM Enhancement

Further Processing
- Path extraction
SSM Enhancement

Further Processing

- Path extraction
- Pairwise relations
SSM Enhancement

Further Processing

- Path extraction
- Pairwise relations
- Grouping (transitivity)
SSM Enhancement

Further Processing

- Path extraction
- Pairwise relations
- Grouping (transitivity)
SSM Enhancement

Example: Zager & Evans “In The Year 2525”
SSM Enhancement

**Example:** Zager & Evans “In The Year 2525”

Missing relations because of transposed sections
SSM Enhancement

**Example:** Zager & Evans “In The Year 2525”

Idea: Cyclic shift of one of the chroma sequences

One semitone up
SSM Enhancement

Example: Zager & Evans “In The Year 2525”

Idea: Cyclic shift of one of the chroma sequences

Two semitones up
SSM Enhancement

Example: Zager & Evans “In The Year 2525”

Idea: Overlay & Maximize ➔ Transposition-invariant SSM
Overview

- Introduction
- Feature Representations
- Self-Similarity Matrices
- Novelty-Based Segmentation
Novelty-Based Segmentation

General goals:

- Find instances where musical changes occur.
- Find transition between subsequent musical parts.

Idea (Foote):

Use checkerboard-like kernel function to detect corner points on main diagonal of SSM.
Novelty-Based Segmentation

Idea (Foote):

Use checkerboard-like kernel function to detect corner points on main diagonal of SSM.
Novelty-Based Segmentation

Idea (Foote):

Use checkerboard-like kernel function to detect corner points on main diagonal of SSM.
Novelty-Based Segmentation

Idea (Foote):

Use checkerboard-like kernel function to detect corner points on main diagonal of SSM.
Novelty-Based Segmentation

Idea (Foote):

Use checkerboard-like kernel function to detect corner points on main diagonal of SSM.
Novelty-Based Segmentation

Idea (Foote):

Use checkerboard-like kernel function to detect corner points on main diagonal of SSM.

Novelty function using
Novelty-Based Segmentation

Idea (Foote):

Use checkerboard-like kernel function to detect corner points on main diagonal of SSM.
Conclusions

Structure Analysis
Conclusions
Conclusions
Conclusions
Conclusions
Links

- SM Toolbox (MATLAB)
  http://www.audiolabs-erlangen.de/resources/MIR/SMtoolbox/

- MSAF: Music Structure Analysis Framework (Python)
  https://github.com/urinieto/msaf

- SALAMI Annotation Data

- LibROSA (Python)
  https://librosa.github.io/librosa/

- Evaluation: mir_eval (Python)
  https://craffel.github.io/mir_eval/

- Deep Learning: Boundary Detection
  Jan Schlüter (PhD thesis)