

**Selected Topics in Deep Learning for
Audio, Speech, and Music Processing**

Introduction to Music Processing

Meinard Müller

International Audio Laboratories Erlangen
meinard.mueller@audiolabs-erlangen.de

26.04.2021

Group Meinard Müller

- Frank Zalkow
- Sebastian Rosenzweig
- Michael Krause
- Yigitcan Özer
- Peter Meier (extern)



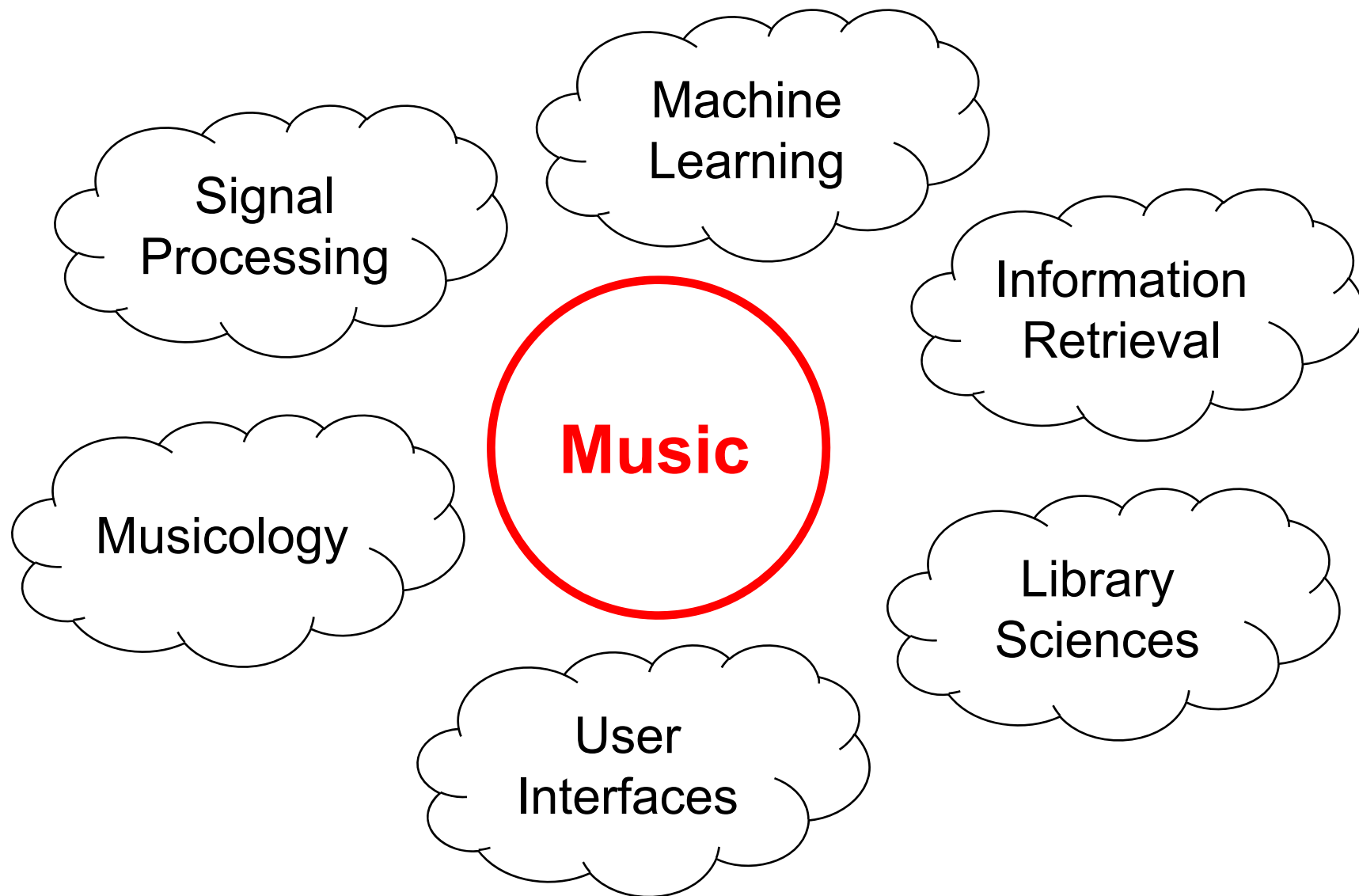
- Christian Dittmar
- Christof Weiß
- Stefan Balke
- Jonathan Driedger
- Thomas Prätzlich
- ...



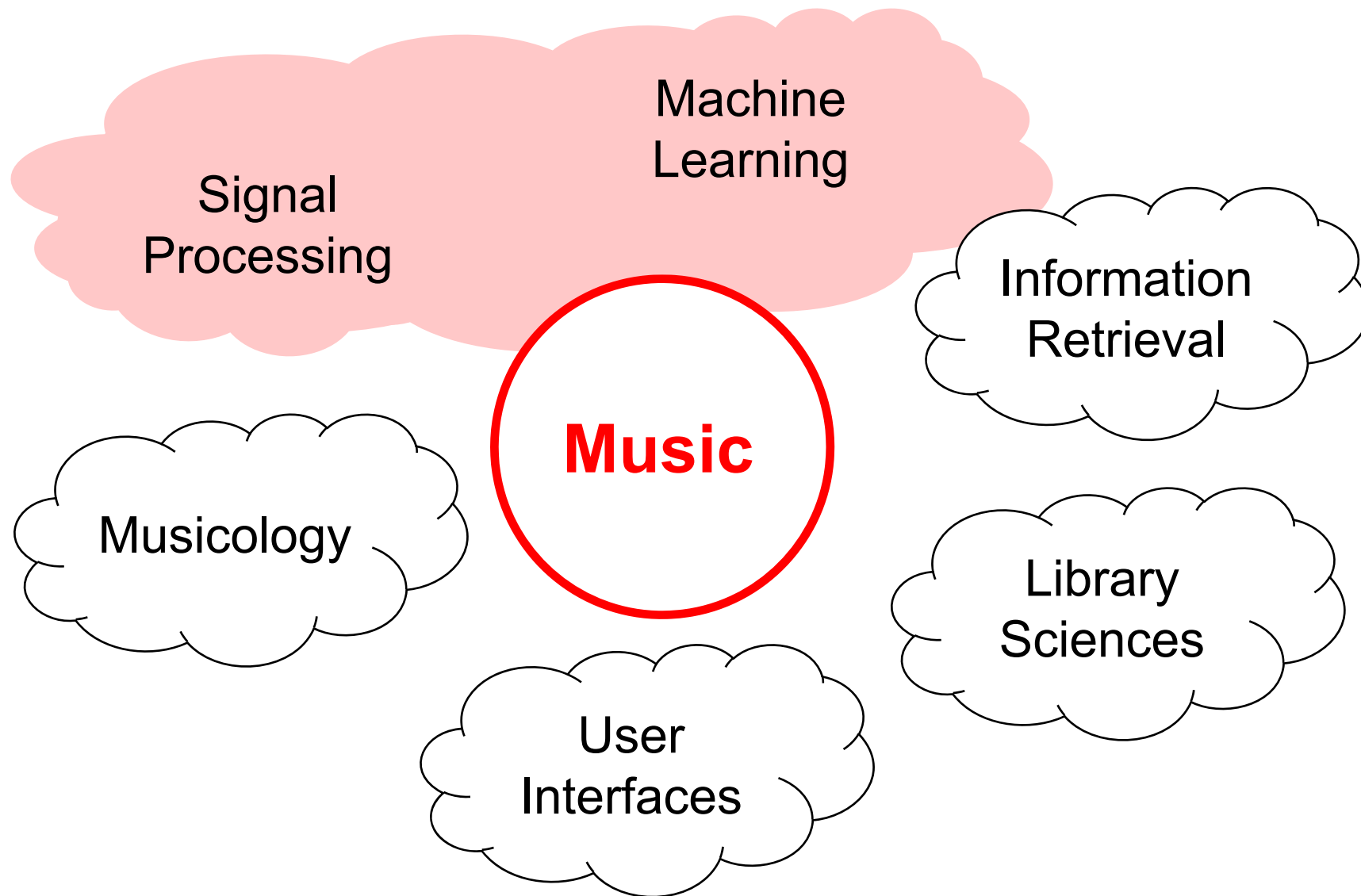
Music



Music Information Retrieval (MIR)



Music Information Retrieval (MIR)

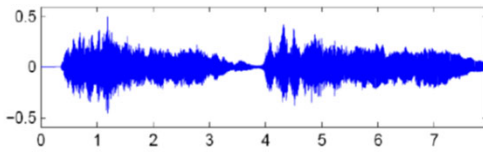


Music Information Retrieval (MIR)

Sheet Music (Image)



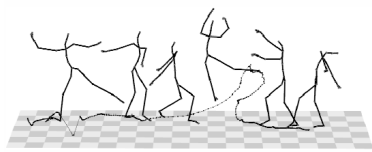
CD / MP3 (Audio)



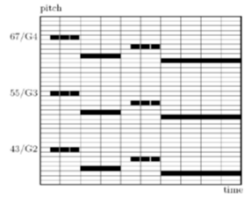
MusicXML (Text)

```
<note>  
  <pitch>  
    <step>E</step>  
    <alter>-1</alter>  
    <octave>4</octave>  
  </pitch>  
  <duration>2</duration>  
  <type>half</type>  
</note>
```

Dance / Motion (Mocap)



MIDI



Singing / Voice (Audio)



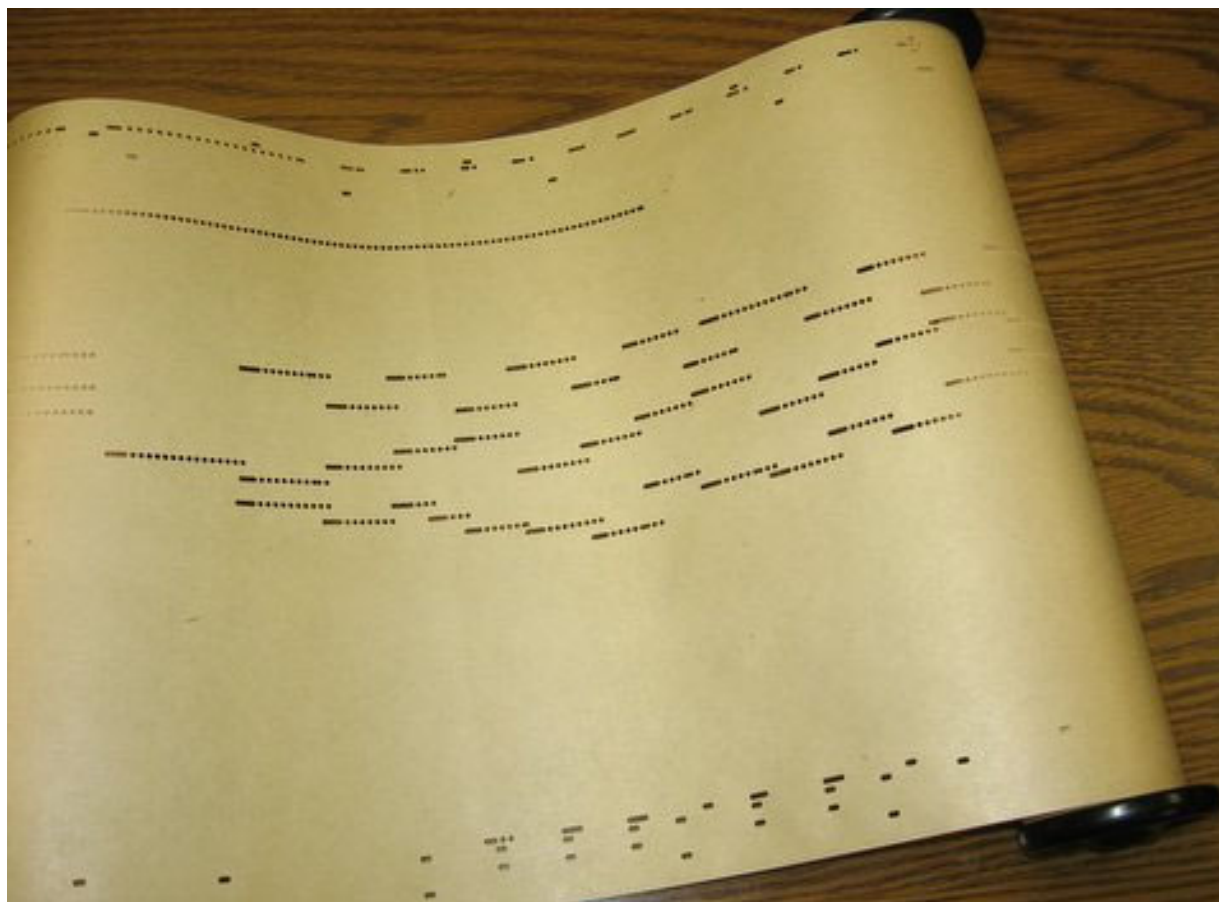
Music Film (Video)



Music Literature (Text)



Piano Roll Representation



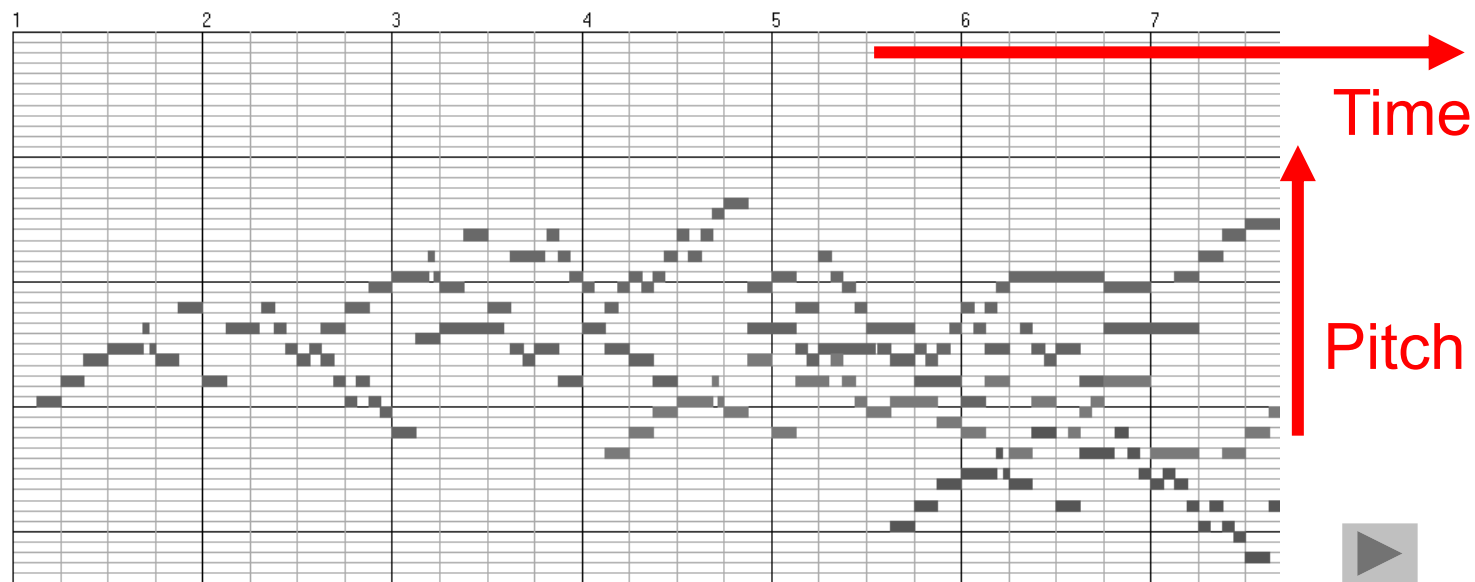
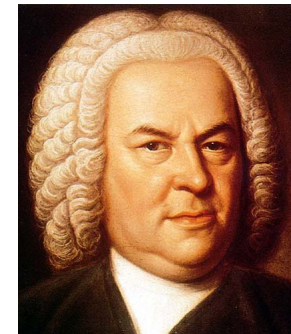
Player Piano (1900)



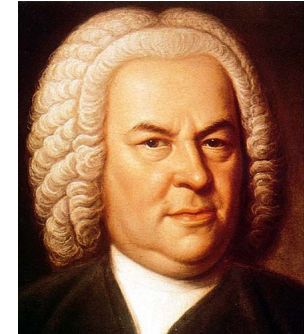
Piano Roll Representation (MIDI)

J.S. Bach, C-Major Fuge

(Well Tempered Piano, BWV 846)



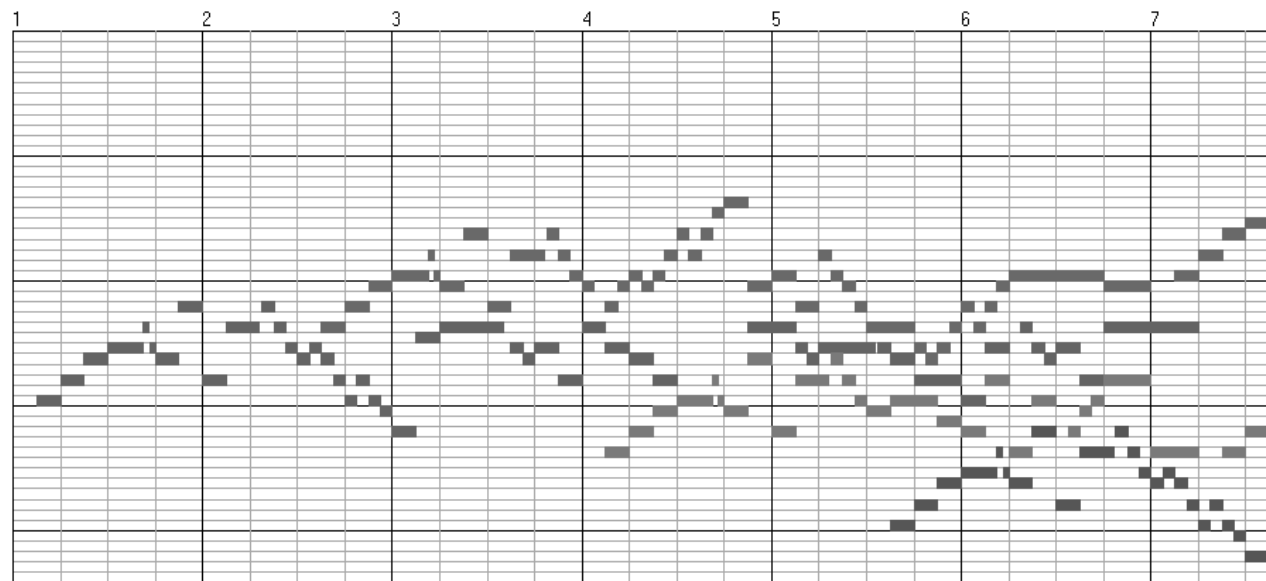
Piano Roll Representation (MIDI)



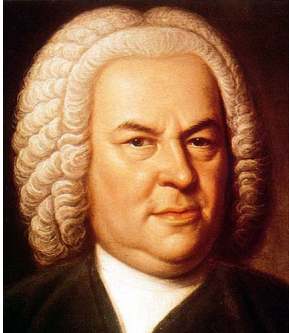
Query:



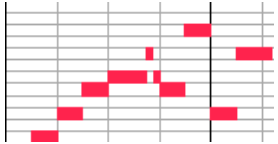
Goal: Find all occurrences of the query



Piano Roll Representation (MIDI)

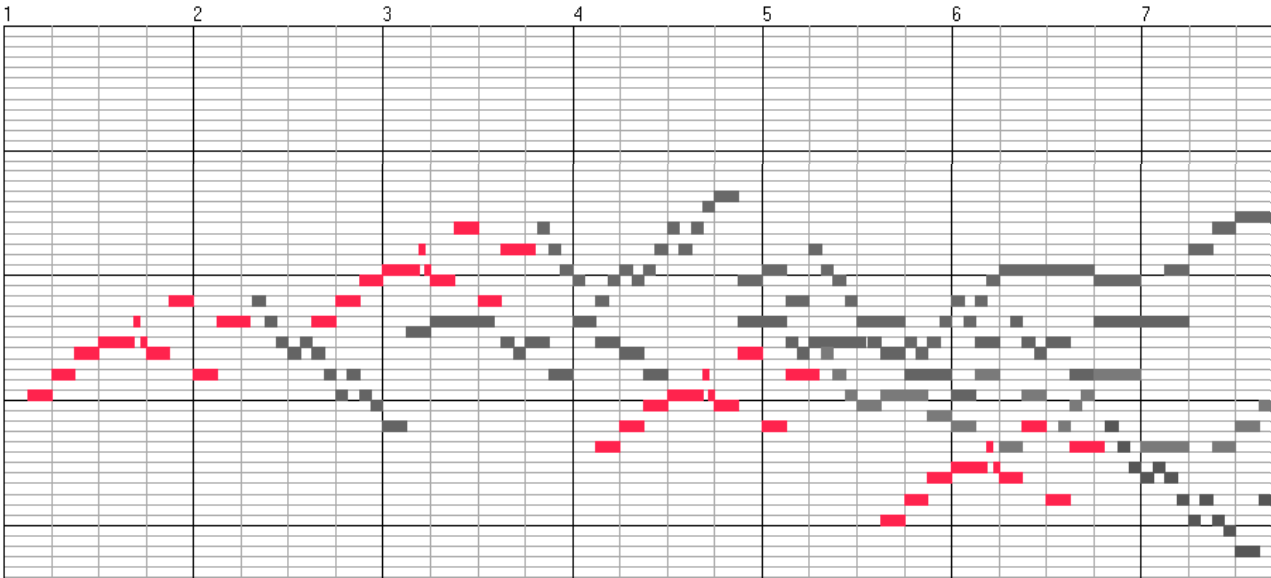


Query:



Goal: Find all occurrences of the query

Matches:

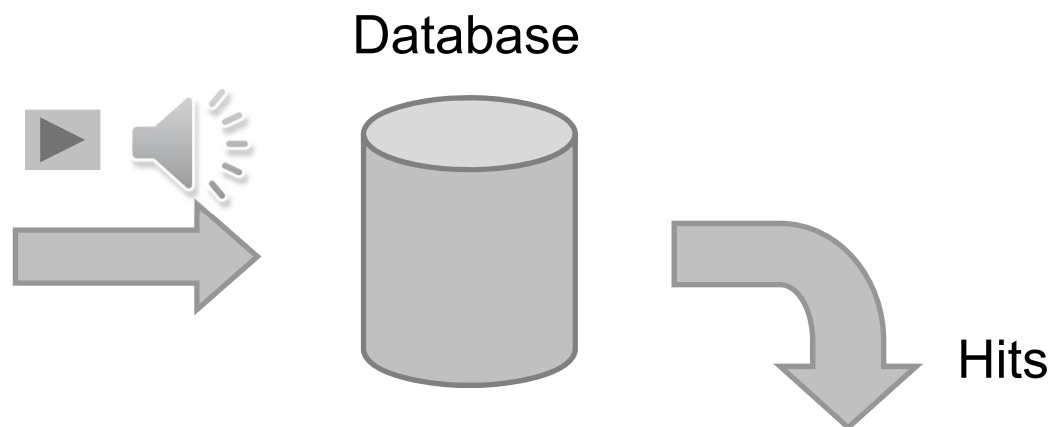


Music Retrieval

Query



The query box contains three visual representations of a musical query: a musical score with treble and bass staves, a blue waveform, and a piano roll.



Retrieval tasks:

Audio identification

Audio matching

Version identification

Category-based music retrieval

Bernstein (1962)
Beethoven, Symphony No. 5

Beethoven, Symphony No. 5:

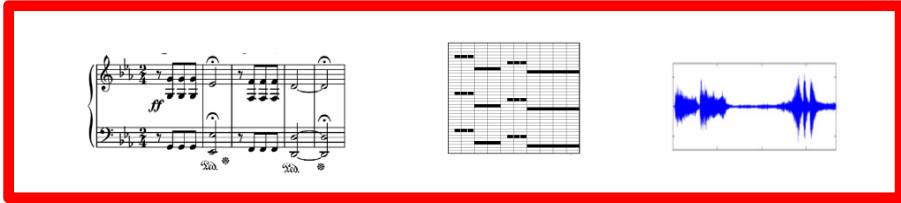
- Bernstein (1962)
- Karajan (1982)
- Gould (1992)

- Beethoven, Symphony No. 9
- Beethoven, Symphony No. 3
- Haydn Symphony No. 94



Music Retrieval

Modalities



Retrieval tasks:

Audio identification

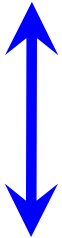
Audio matching

Version identification

Category-based music retrieval

Specificity

High specificity



Low specificity

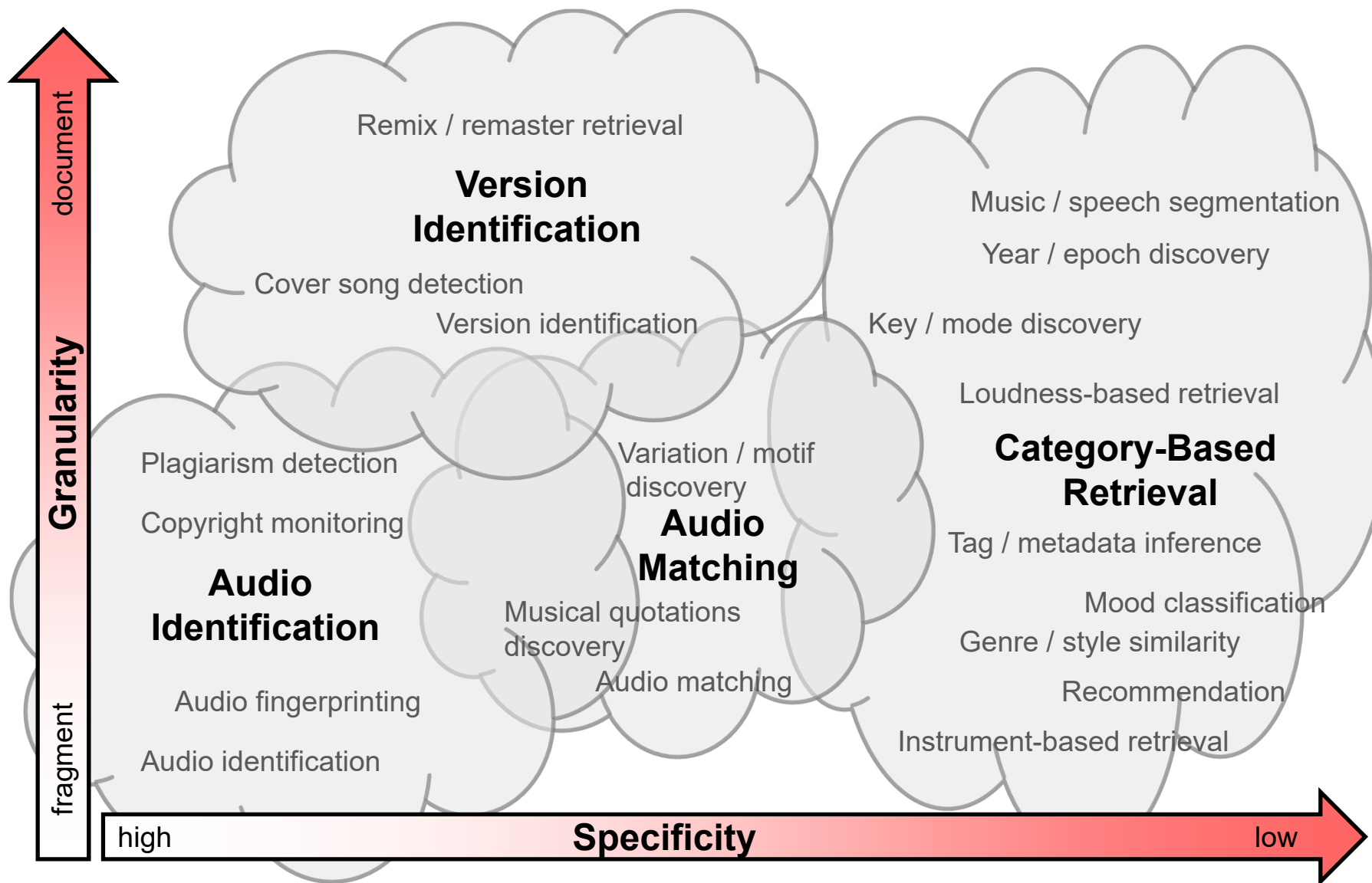
Granularity

Fragment-based retrieval



Document-based retrieval

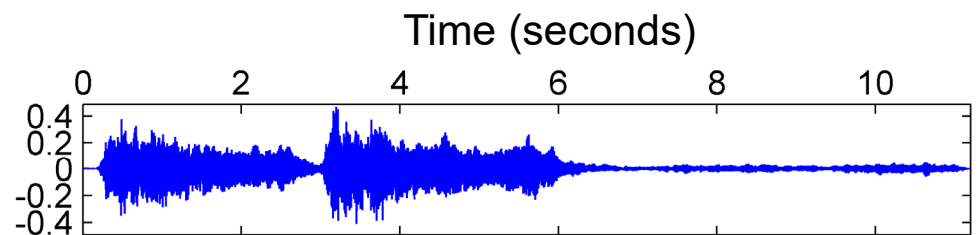
Music Retrieval



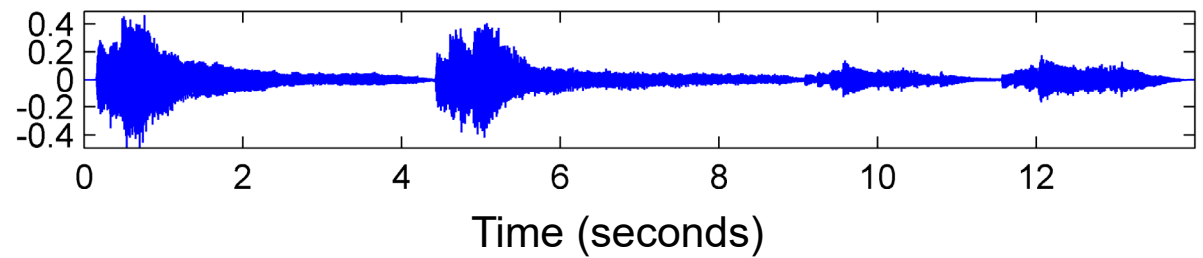
Music Synchronization: Audio-Audio

Beethoven's Fifth

Karajan 



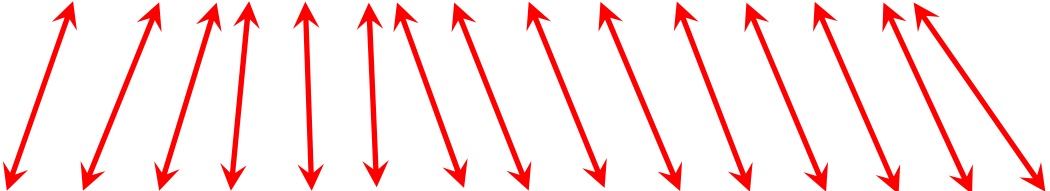
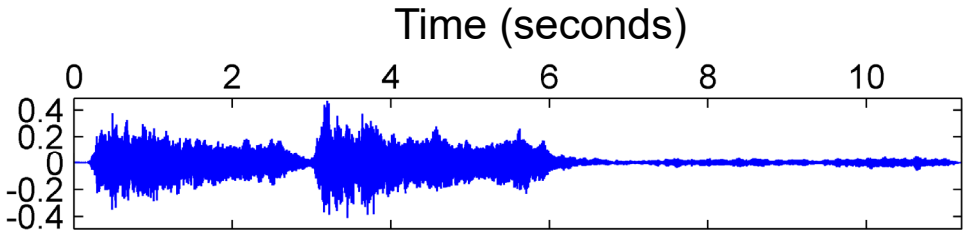
Gould 



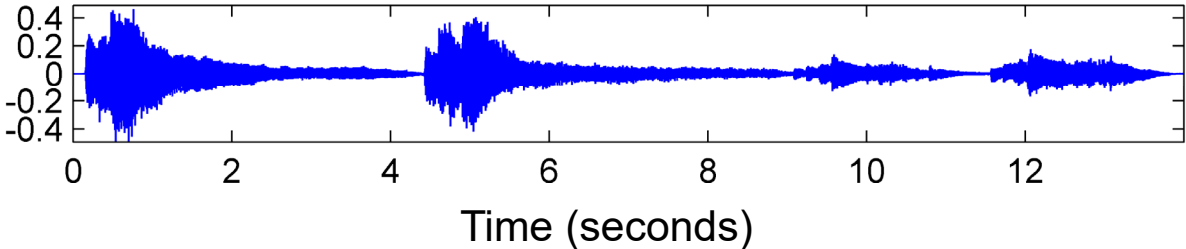
Music Synchronization: Audio-Audio

Beethoven's Fifth

Karajan



Gould



Application: Interpretation Switcher

The screenshot shows a software window titled "Interpretation Switcher" for the piece "Beethoven, Op067-1_Symphony5". The interface features four horizontal progress bars, each representing a different interpretation: "midi", "Bernstein", "Sawallisch", and "Scherbakov". Each bar is divided into three segments: blue, red, and green. The "midi" bar has the shortest segments, while "Bernstein" has the longest. The "Sawallisch" bar has a noticeably longer blue segment compared to the others. Each bar is accompanied by a play button icon and a timestamp of "00:00.00".

On the right side of the window, there is a list of checked checkboxes for the interpretations: "midi", "Bernstein", "Sawallisch", and "Scherbakov". Below this list is a "Deselect all" button.

The bottom of the window contains a control bar with the following elements from left to right: a radio button menu with "Absolute" selected (indicated by a checkmark), "Relative", and "Reference"; a play button icon; a square stop button icon; a "Movement selection" button with a double-headed arrow icon; an "Interval Repeat" checkbox; and an "Info" button with a question mark icon.



Music Synchronization: Audio-Audio

Task

Given: Two different audio recordings (two versions) of the same underlying piece of music.

Goal: Find for each position in one audio recording the **musically** corresponding position in the other audio recording.

Music Synchronization: Audio-Audio

Traditional Engineering Approach:

1.) Feature extraction

- Robust to variations (e.g., instrumentation, timbre, dynamics)
- Discriminative (e.g., capturing harmonic, melodic, tonal aspects)

➡ **Chroma features**

2.) Temporal alignment

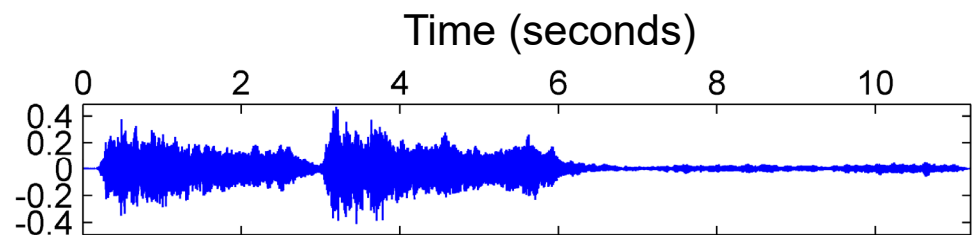
- Capturing local and global tempo variations
- Trade-off: Robustness vs. accuracy
- Efficiency

➡ **Dynamic time warping (DTW)**

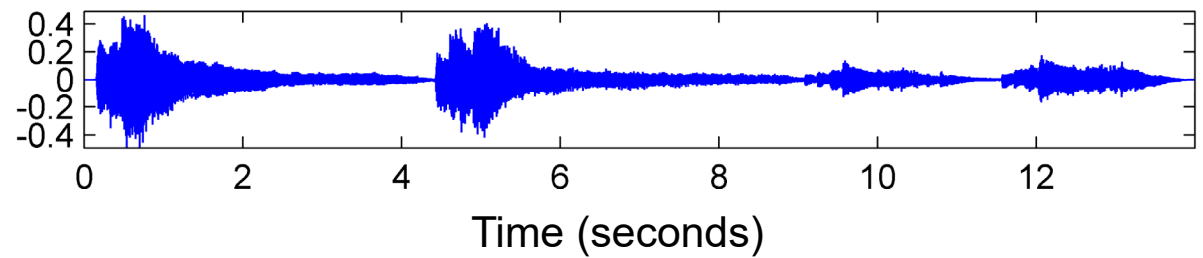
Music Synchronization: Audio-Audio

Beethoven's Fifth

Karajan 



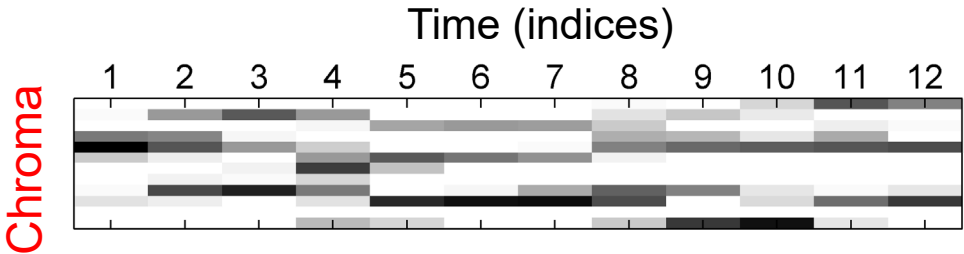
Gould 



Music Synchronization: Audio-Audio

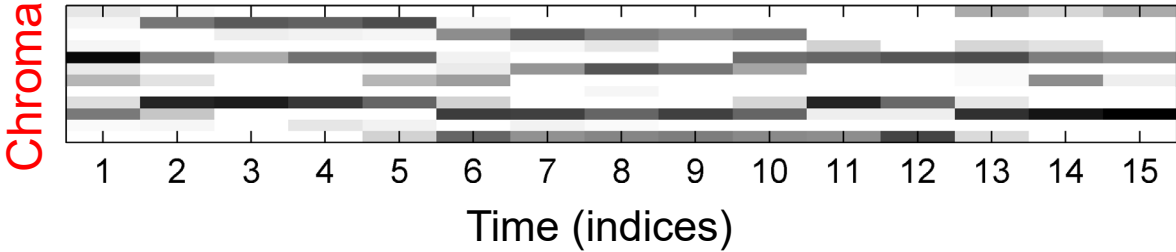
Beethoven's Fifth

Karajan





Time–chroma representations

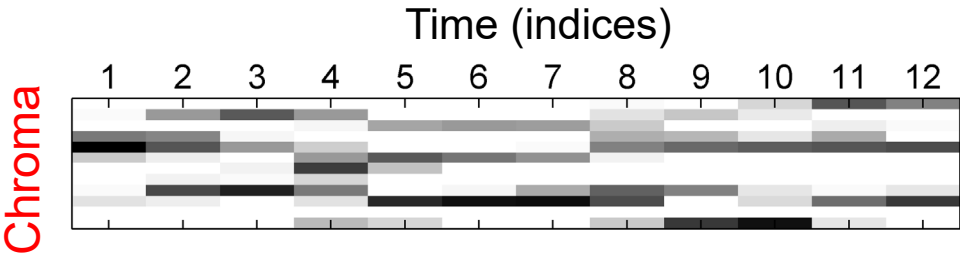
Gould



Music Synchronization: Audio-Audio

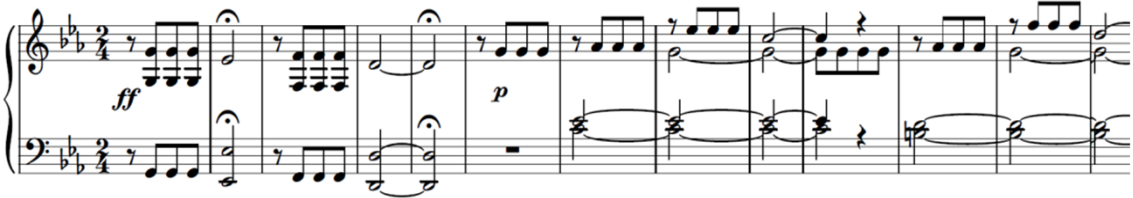
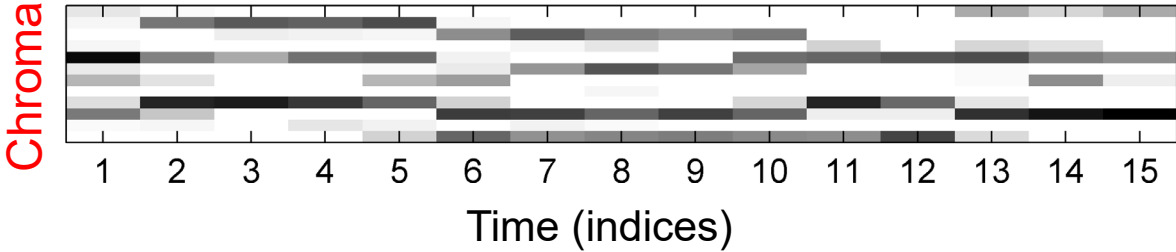
Beethoven's Fifth

Karajan 






Time–chroma representations

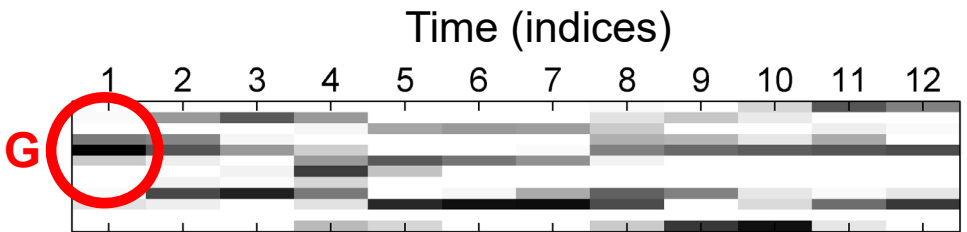
Gould 



Music Synchronization: Audio-Audio


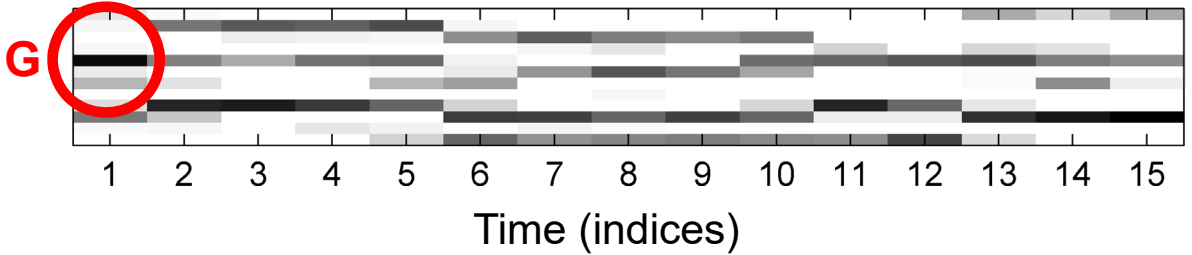
Beethoven's Fifth

Karajan 






Time-chroma representations

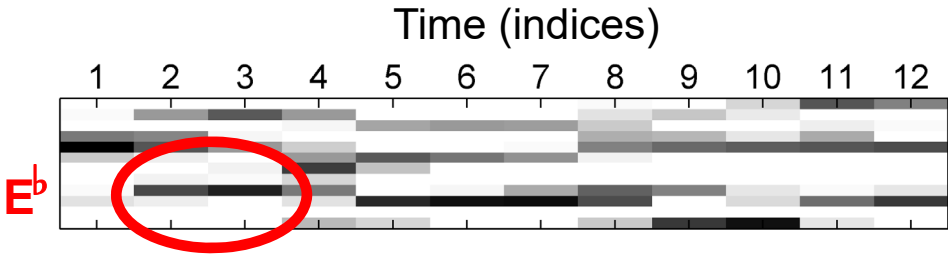
Gould 




Music Synchronization: Audio-Audio

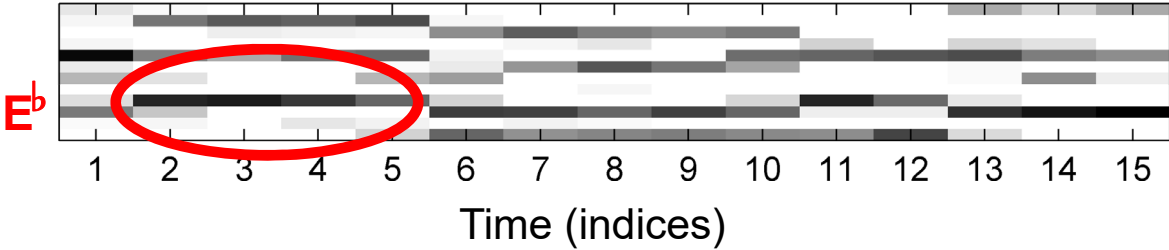
Beethoven's Fifth

Karajan 




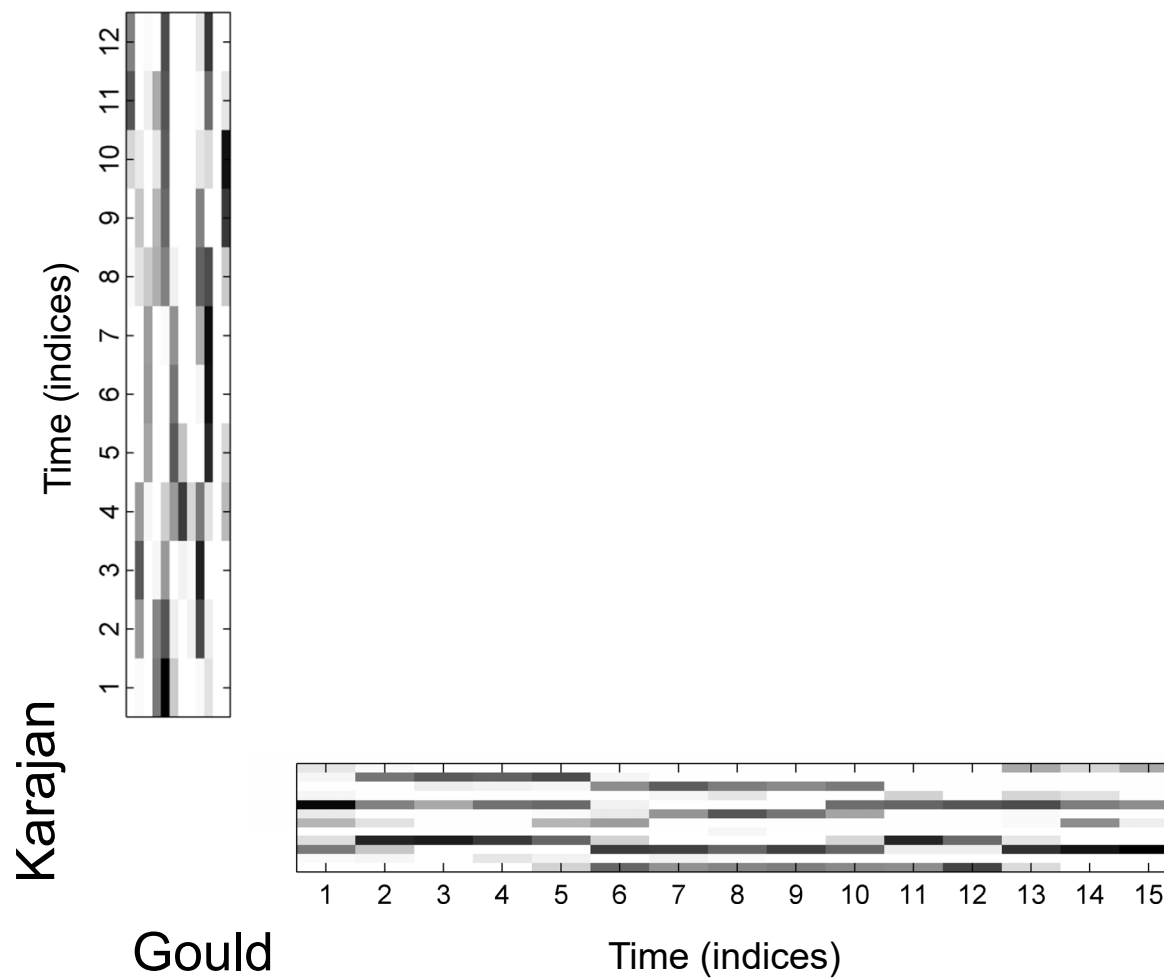
Time–chroma representations

Gould 

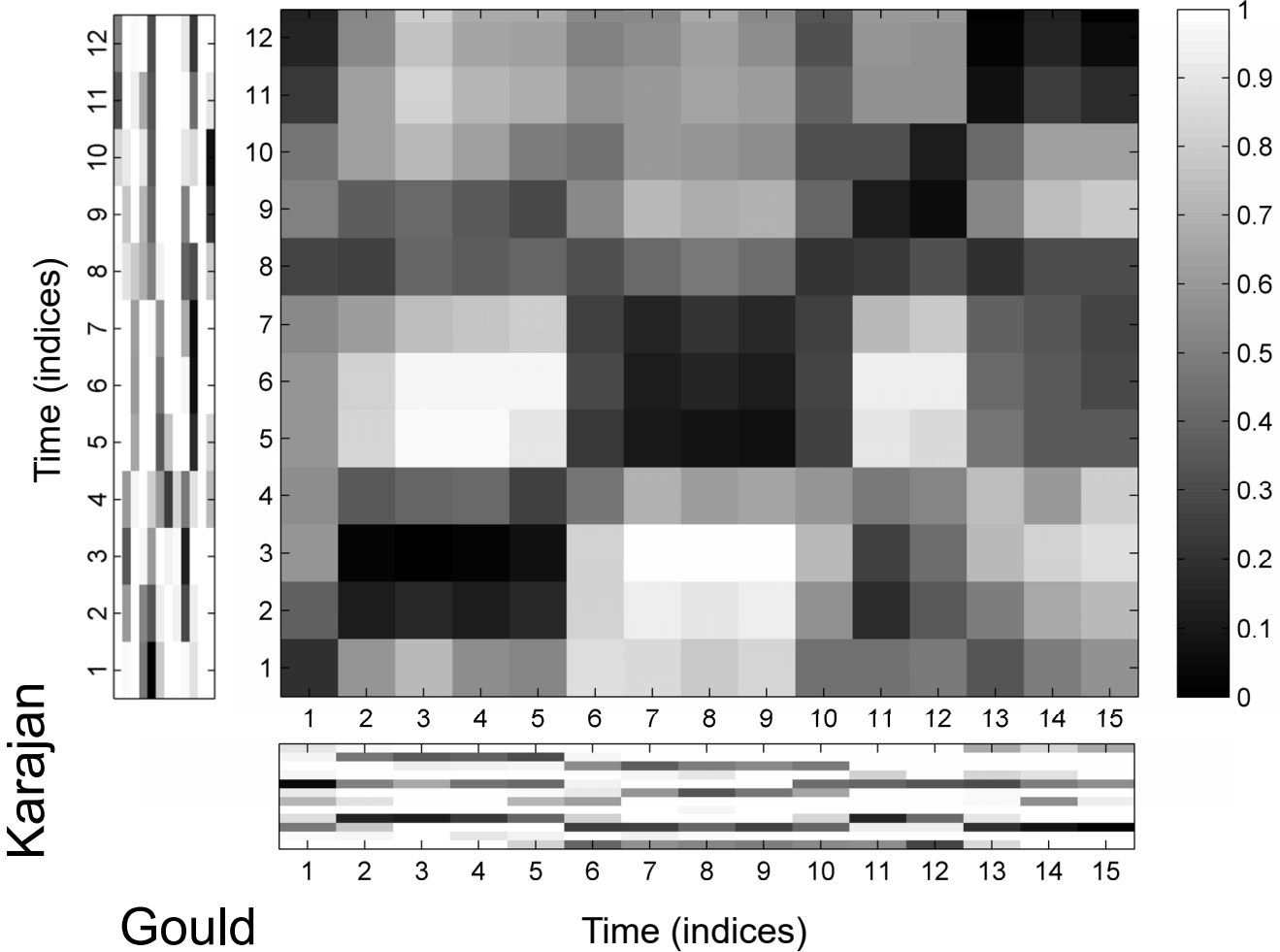
ff *p*

Music Synchronization: Audio-Audio



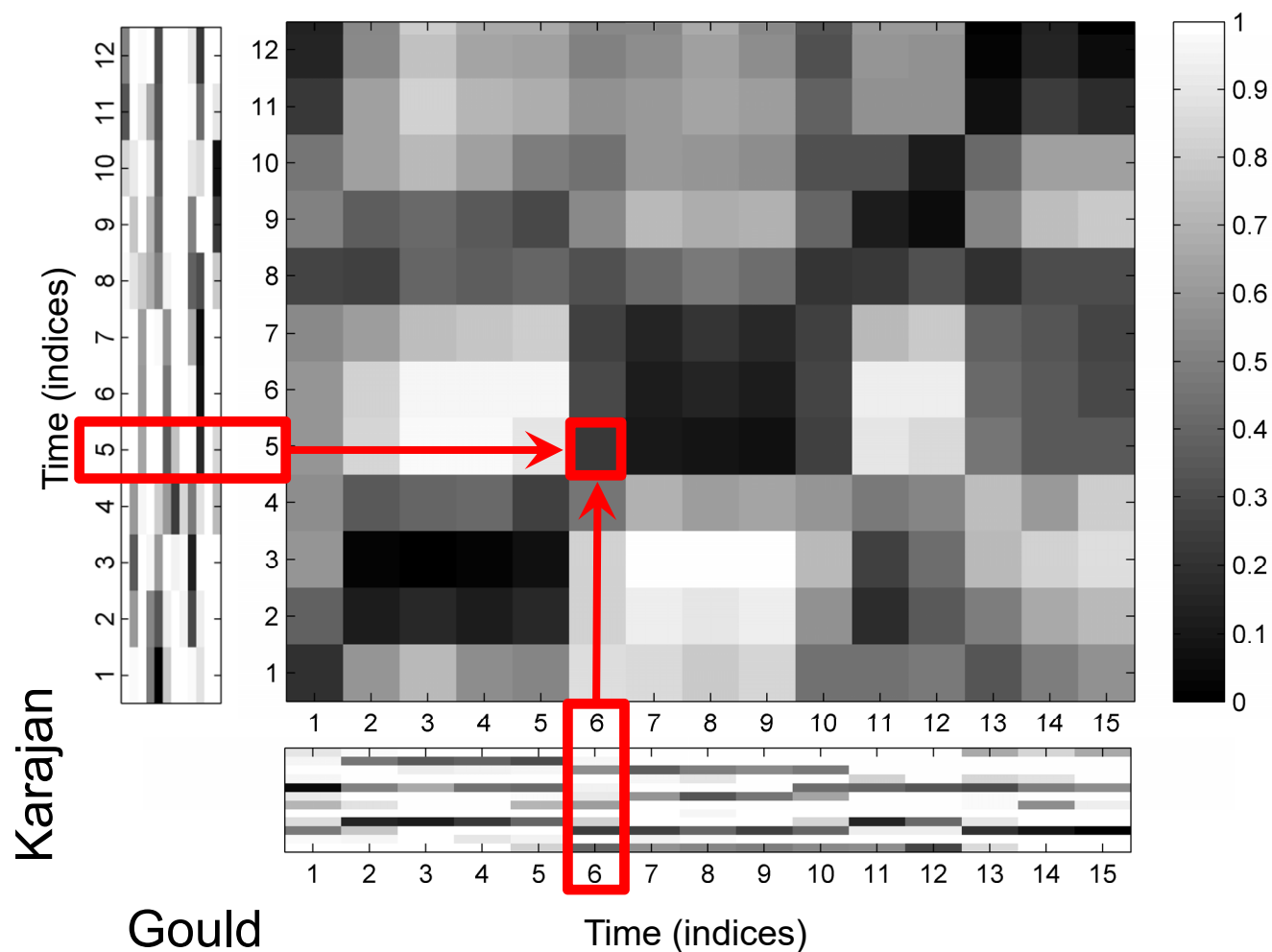
Music Synchronization: Audio-Audio

Cost matrix



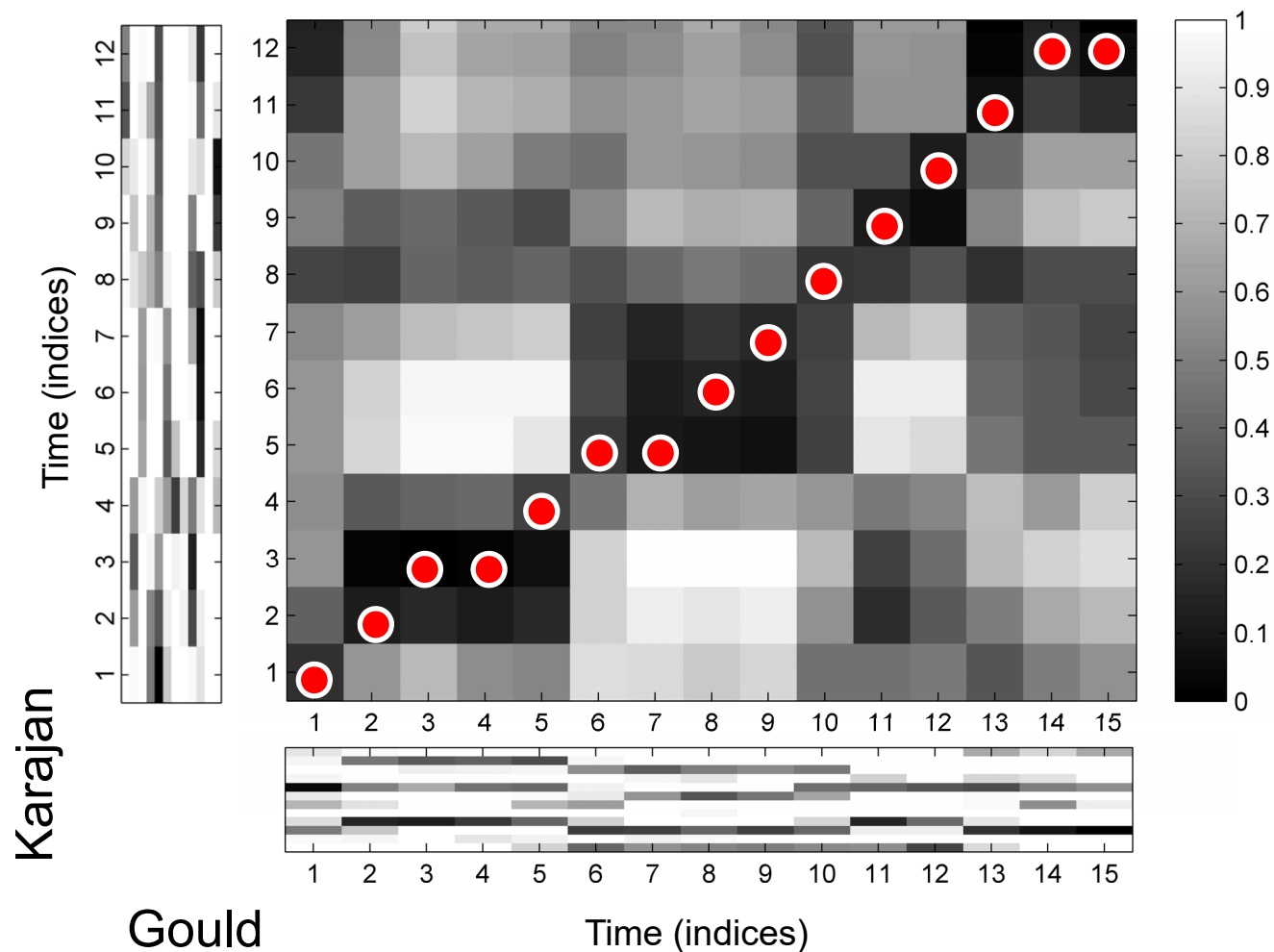
Music Synchronization: Audio-Audio

Cost matrix



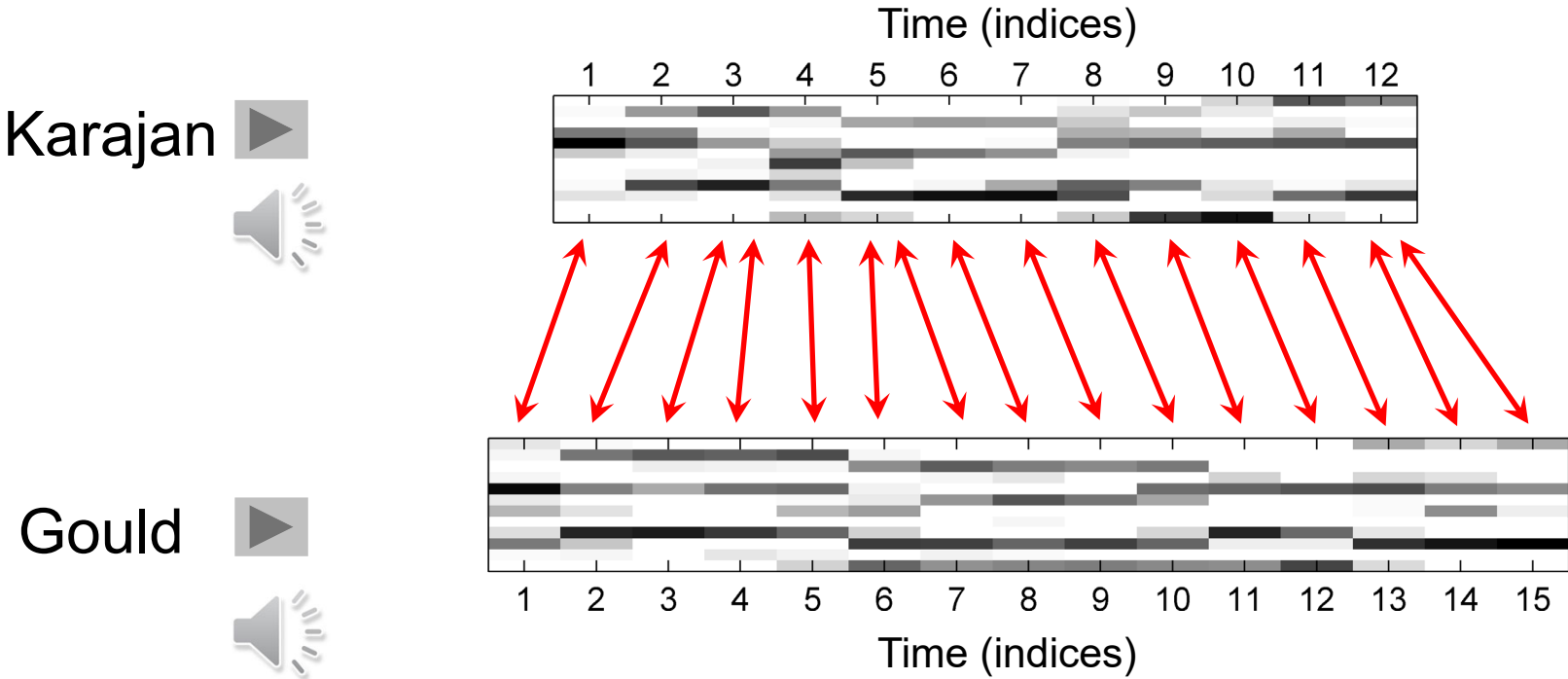
Music Synchronization: Audio-Audio

Cost-minimizing warping path



Music Synchronization: Audio-Audio

Optimal alignment (cost-minimizing warping path)



Music Synchronization: Audio-Audio

Deep Learning Approaches:

- Learn audio features from data
 - Should be able to achieve high alignment accuracy
 - Should be robust to performance variations
 - Musical relevance?
- Alignment problem
 - Pre-aligned data for training
 - Part of loss function → differentiability?

**Lecture 9: Connectionist
Temporal Classification (CTC)
Loss with Applications to
Theme-Based Music Retrieval**

Music Synchronization: Image-Audio

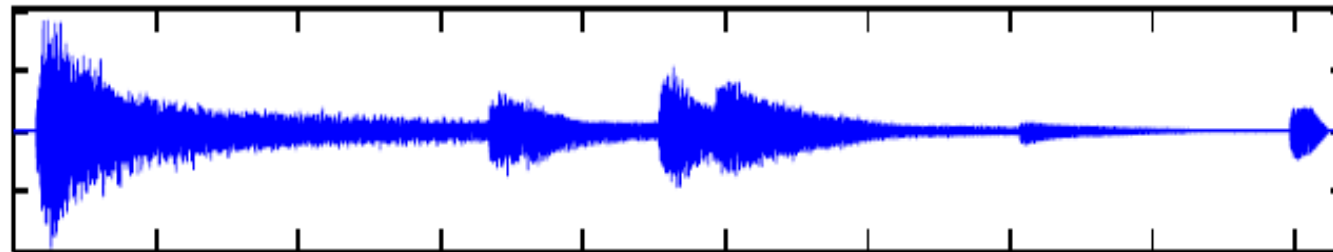
Image

Grave.

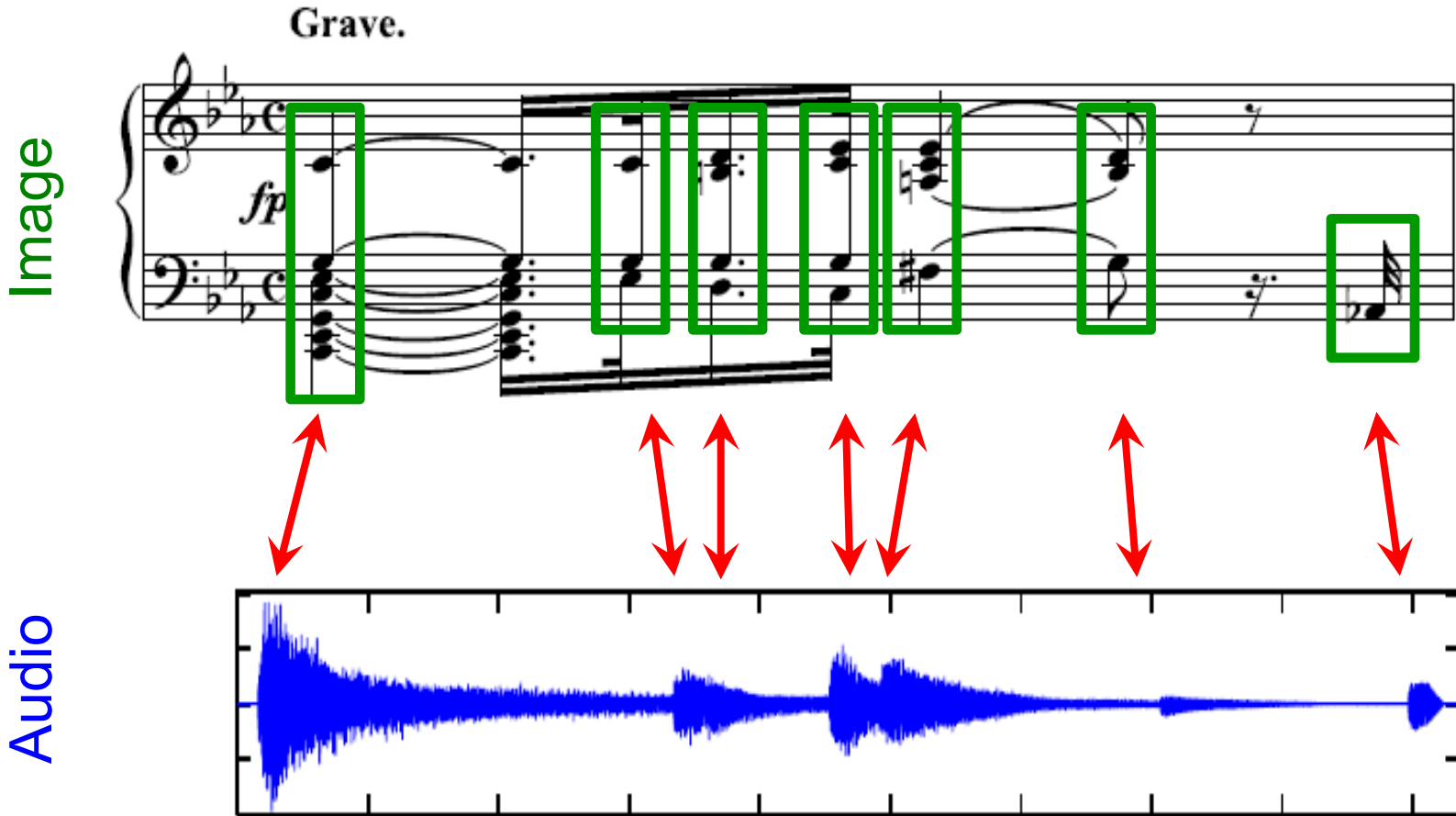


The image shows a musical score for piano, marked "Grave." and "fp". The score is written in G major (one sharp) and common time (C). It consists of two staves: a treble clef staff and a bass clef staff. The music features a slow, somber tempo with a focus on sustained chords and melodic lines. The treble staff begins with a half note chord (G4, B4, D5) followed by a series of chords and a melodic line. The bass staff features a series of chords and a melodic line, with a prominent bass line. The score is marked "fp" (fortissimo) and "Grave." (grave).

Audio



Music Synchronization: Image-Audio



Application: Score Viewer

The image displays a music application interface with two main windows: ScoreViewer and AudioViewer.

ScoreViewer: This window shows a digital score for Beethoven's Piano Sonata no. 8 in C minor, op. 13, "Pathétique", Rondo (Allegro). The score is displayed in a standard musical notation format. Below the score, there are navigation controls: Track: 29 / 54, Bar: 1 / 211, and Page: 159 / 285. There are also buttons for "Score Following On", "Play", and "Stop".

AudioViewer: This window shows a track listing for "Beethoven - Piano Sonatas - Alfred Brendel". The listing includes the following tracks and durations:

Track	Track Name	Duration
03	Sonata no.1 in F minor, op.2 no.1 / Menuetto (Allegretto)	3:24
04	Sonata no.1 in F minor, op.2 no.1 / Prestissimo	5:32
05	Sonata no.2 in A major, op.2 no.2 / Allegro vivace	7:15
06	Sonata no.2 in A major, op.2 no.2 / Largo appassionato	6:28
07	Sonata no.2 in A major, op.2 no.2 / Scherzo (Allegretto)	3:30
08	Sonata no.2 in A major, op.2 no.2 / Rondo (Grazioso)	7:03
09	Sonata no.8 in C minor, op.13 "Pathétique" / Allegro di molto e con brio	9:40
10	Sonata no.8 in C minor, op.13 "Pathétique" / Adagio cantabile	5:17
11	Sonata no.8 in C minor, op.13 "Pathétique" / Rondo (Allegro)	4:30

Below the track listing, there are navigation controls: Disc: 1 / 11, Track: 11 / 11, and Time: 00:00.00 / 4:30.35. There are also buttons for "Play" and "Stop".

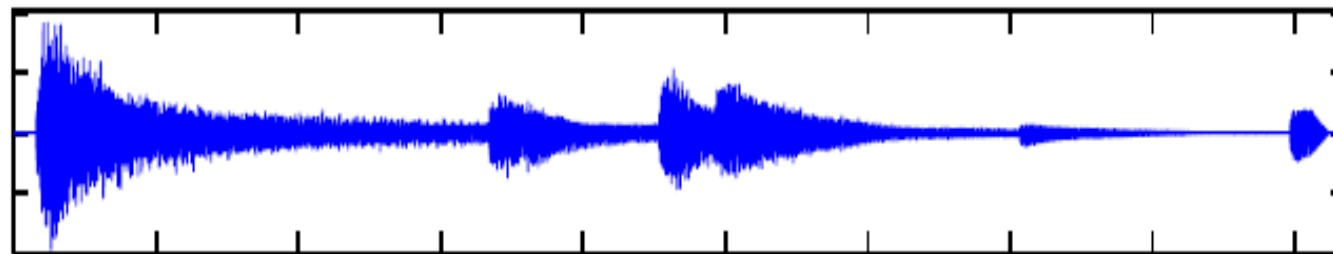


How to make the data comparable?

Image



Audio



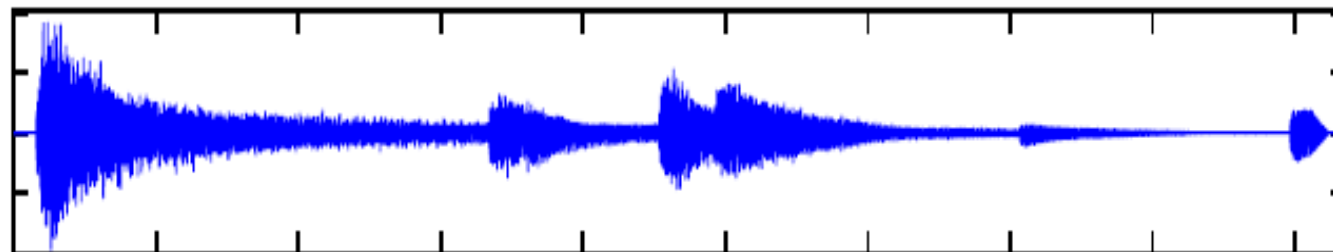
How to make the data comparable?

Image Processing: Optical Music Recognition

Image



Audio



How to make the data comparable?

Image Processing: Optical Music Recognition

Image



Audio



Audio Processing: Fourier Analysis



How to make the data comparable?

Image Processing: Optical Music Recognition

Image



Audio

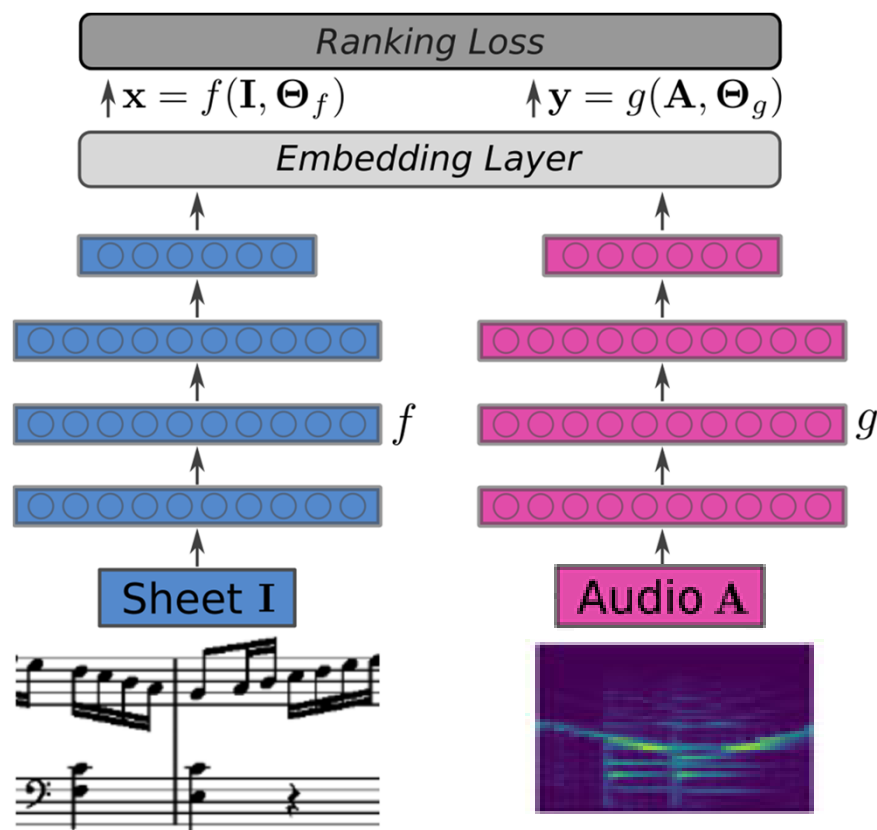


Audio Processing: Fourier Analysis



Music Synchronization: Image-Audio

Deep Learning Approach:

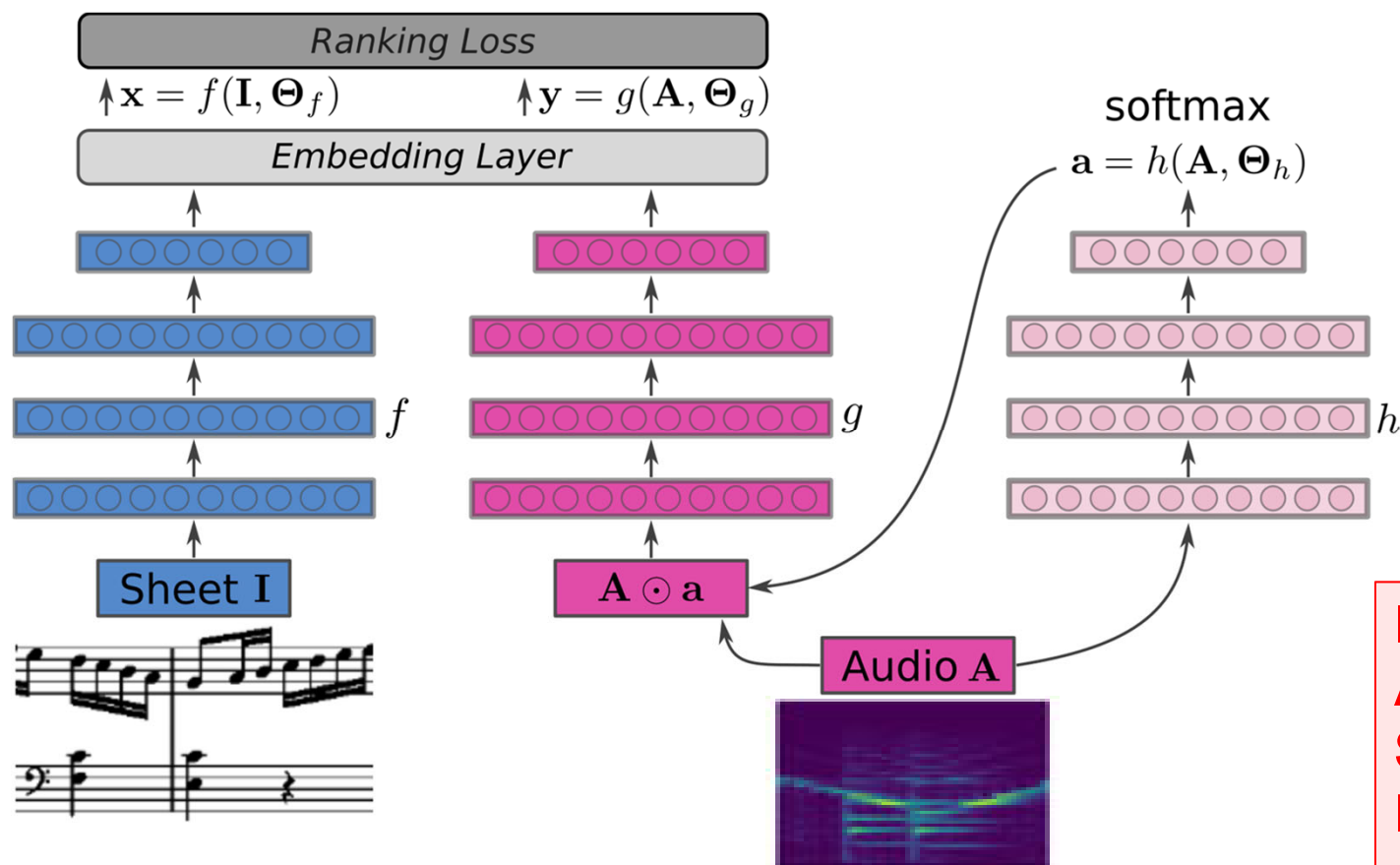


- Cross-modal embedding
- Requires corresponding snippets of audio and sheet music for training
- Triplet Loss function
 $\max(0, d(x^a, y^p) - d(x^a, y^n) + \alpha)$
- Problem very hard
 - Performance variations
 - Layout variations

Dorfer, Schlüter, Vall, Korzeniowski, Widmer.
End-to-End Cross-Modality Retrieval with
CCA Projections and Pairwise Ranking Loss.
International Journal of Multimedia
Information Retrieval, 2018.

Music Synchronization: Image-Audio

Deep Learning Approach: Soft Attention Mechanism



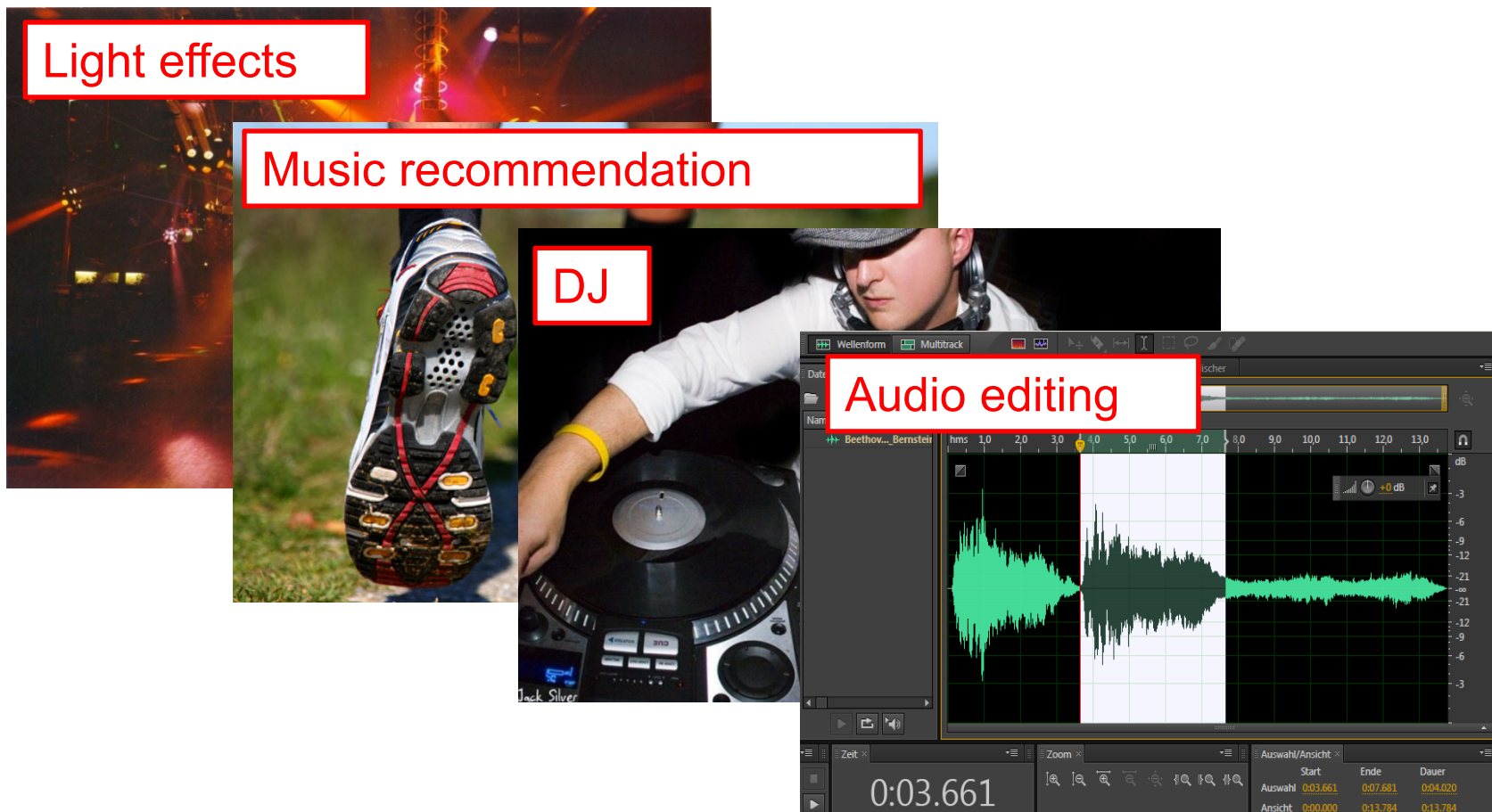
**Lecture 7:
Attention in
Sound Source
Localization
and Speaker
Extraction**

Music Processing

Coarse/Relative Level	Fine/Absolute Level
What do different versions or instances have in common?	What are the characteristics of a specific version or instance?
Provide coarse description: What makes up a piece of music?	Capture nuances and subtleties: What makes music come alive?
Identify despite of differences	Identify the differences
Example tasks: Music Retrieval Genre Classification Global Tempo Estimation	Example tasks: Music Transcription Performance Analysis Local Tempo Estimation

Tempo Estimation and Beat Tracking

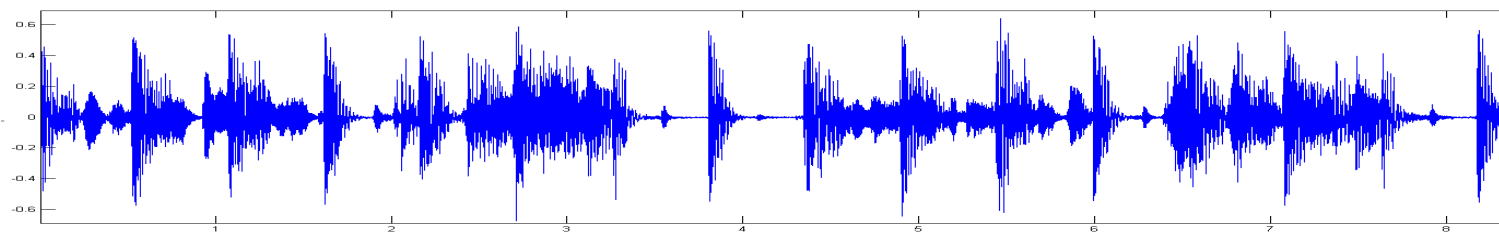
Basic task: “Tapping the foot when listening to music”



Tempo Estimation and Beat Tracking

Basic task: “Tapping the foot when listening to music”

Example: Queen – Another One Bites The Dust



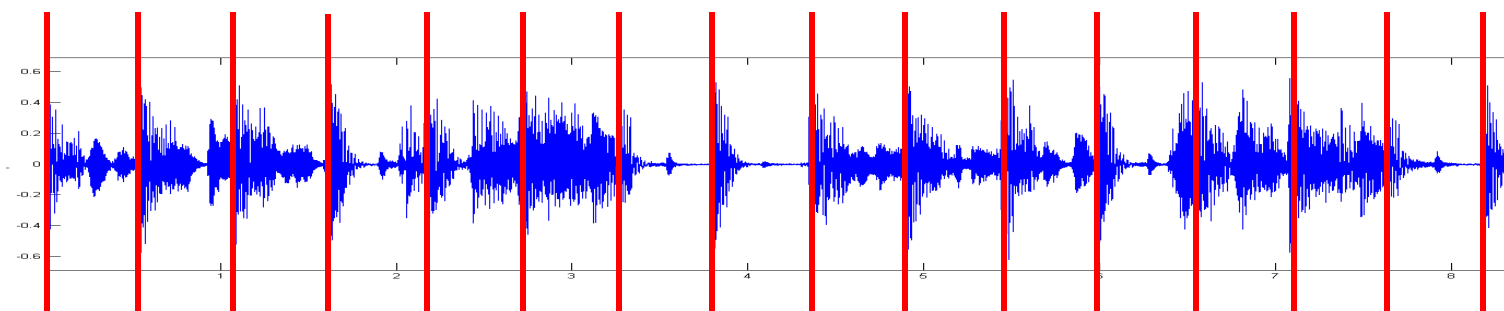
Time (seconds)



Tempo Estimation and Beat Tracking

Basic task: “Tapping the foot when listening to music”

Example: Queen – Another One Bites The Dust



Time (seconds)



Tempo Estimation and Beat Tracking

Example: Chopin – Mazurka Op. 68-3

Pulse level: Quarter note

Tempo: ???



Tempo Estimation and Beat Tracking

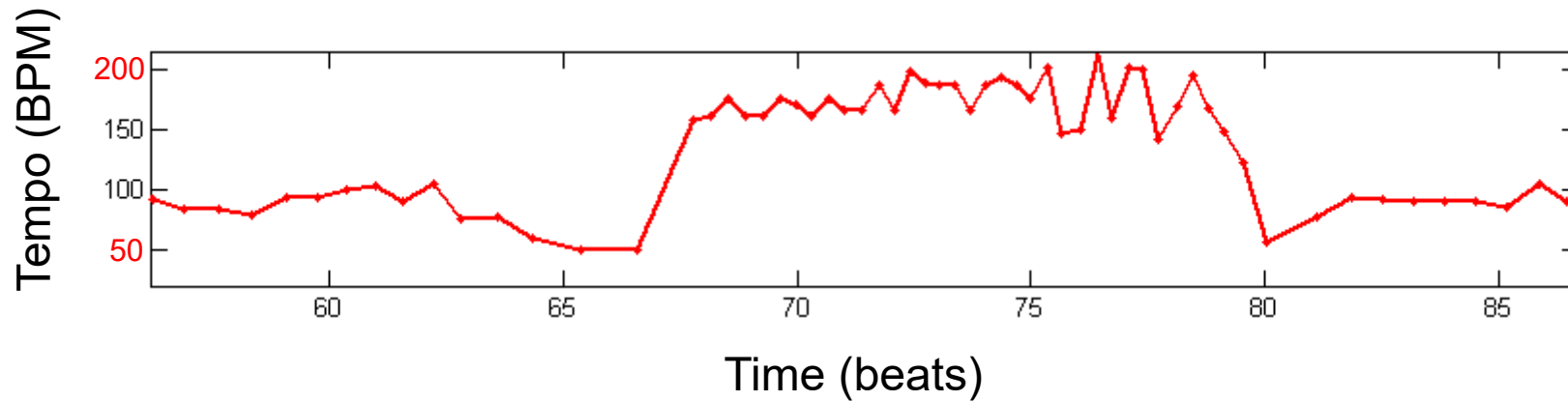
Example: Chopin – Mazurka Op. 68-3

Pulse level: Quarter note

Tempo: **50-200 BPM**



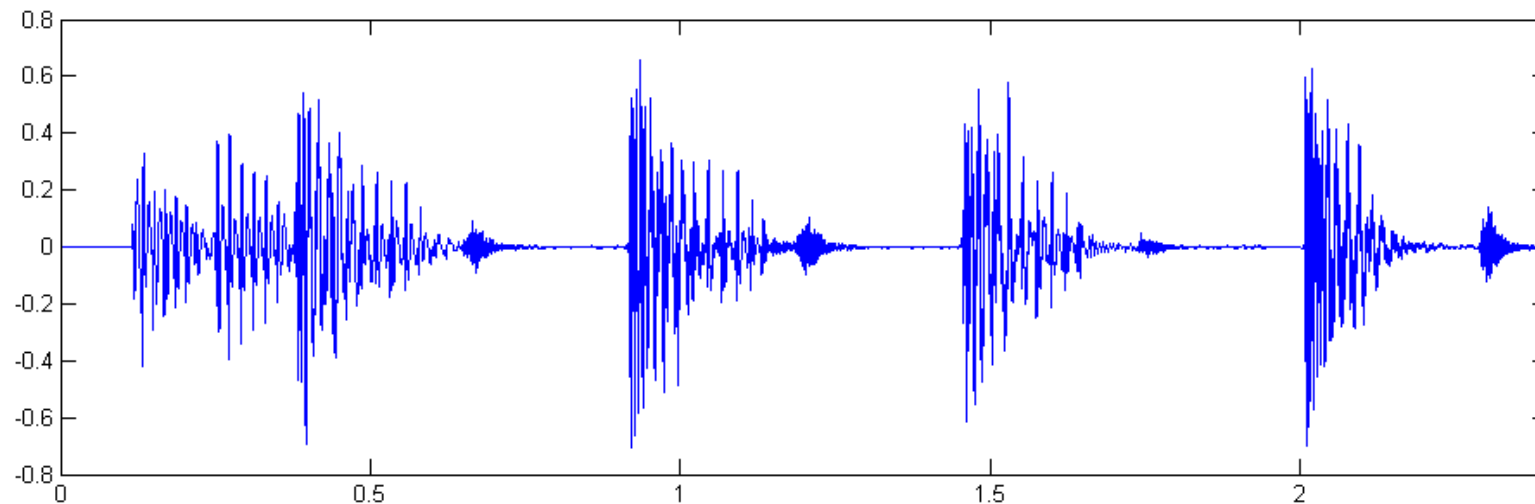
Tempo curve



Tempo Estimation and Beat Tracking

Tasks

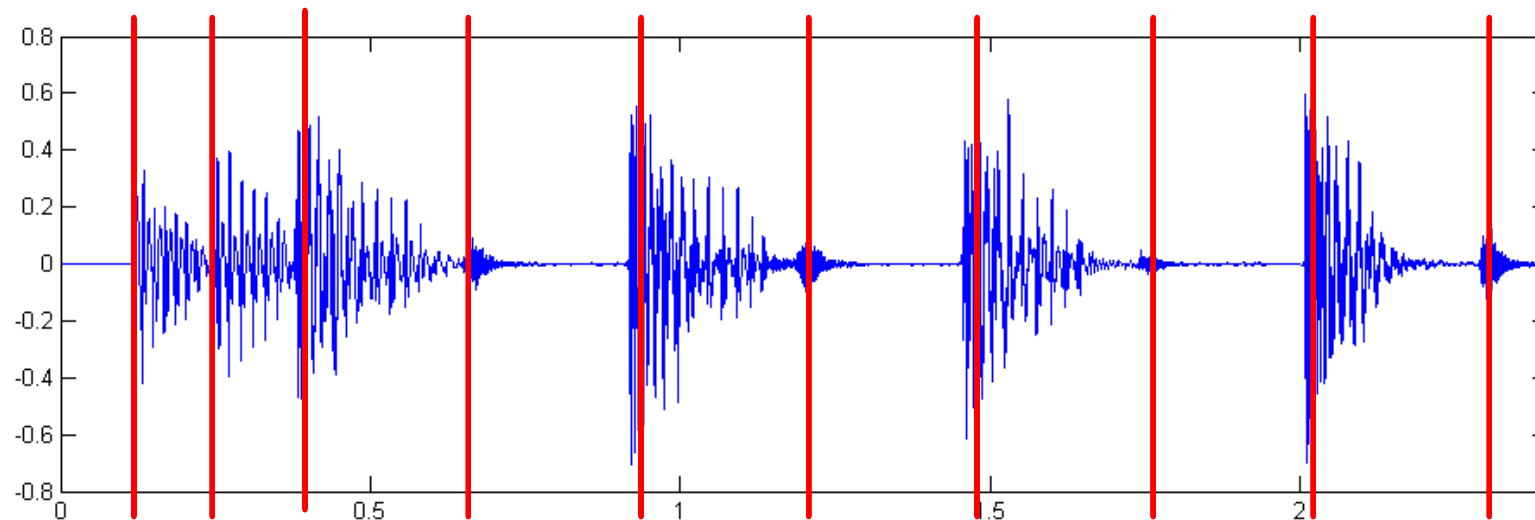
- Onset detection
- Beat tracking
- Tempo estimation



Tempo Estimation and Beat Tracking

Tasks

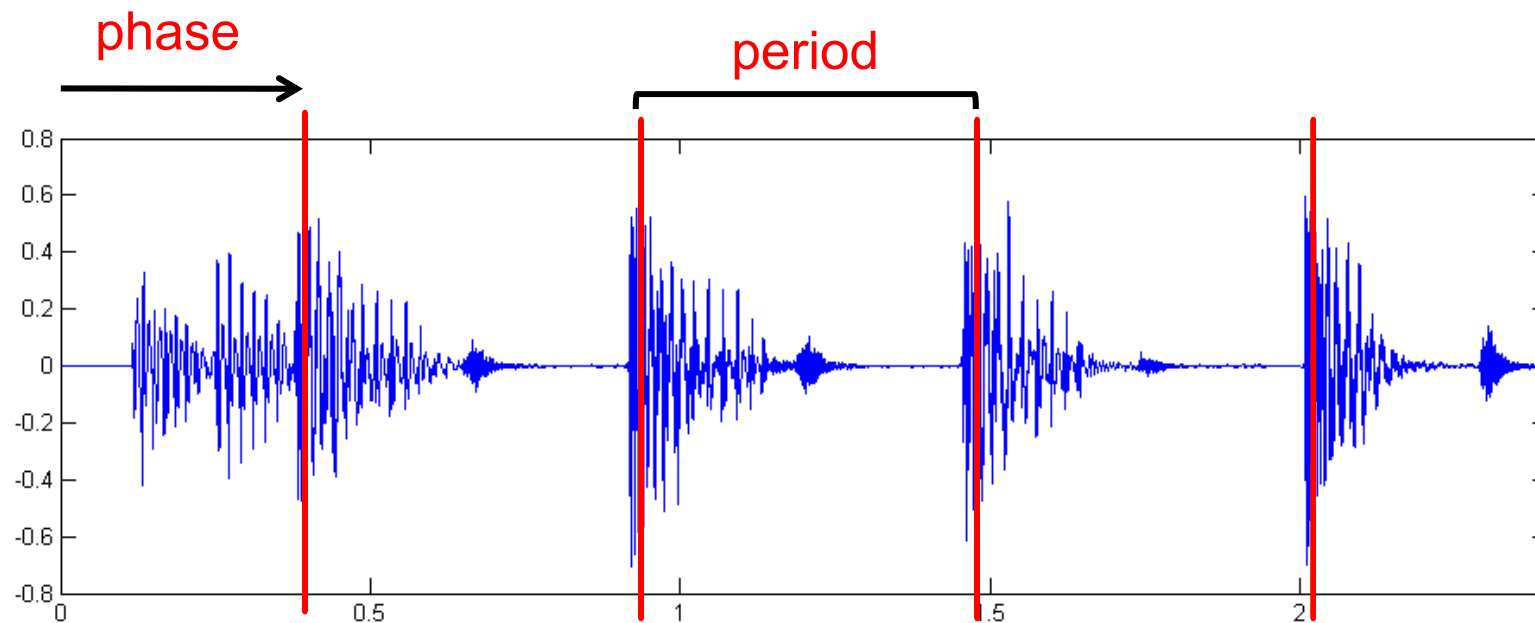
- Onset detection
- Beat tracking
- Tempo estimation



Tempo Estimation and Beat Tracking

Tasks

- Onset detection
- **Beat tracking**
- Tempo estimation



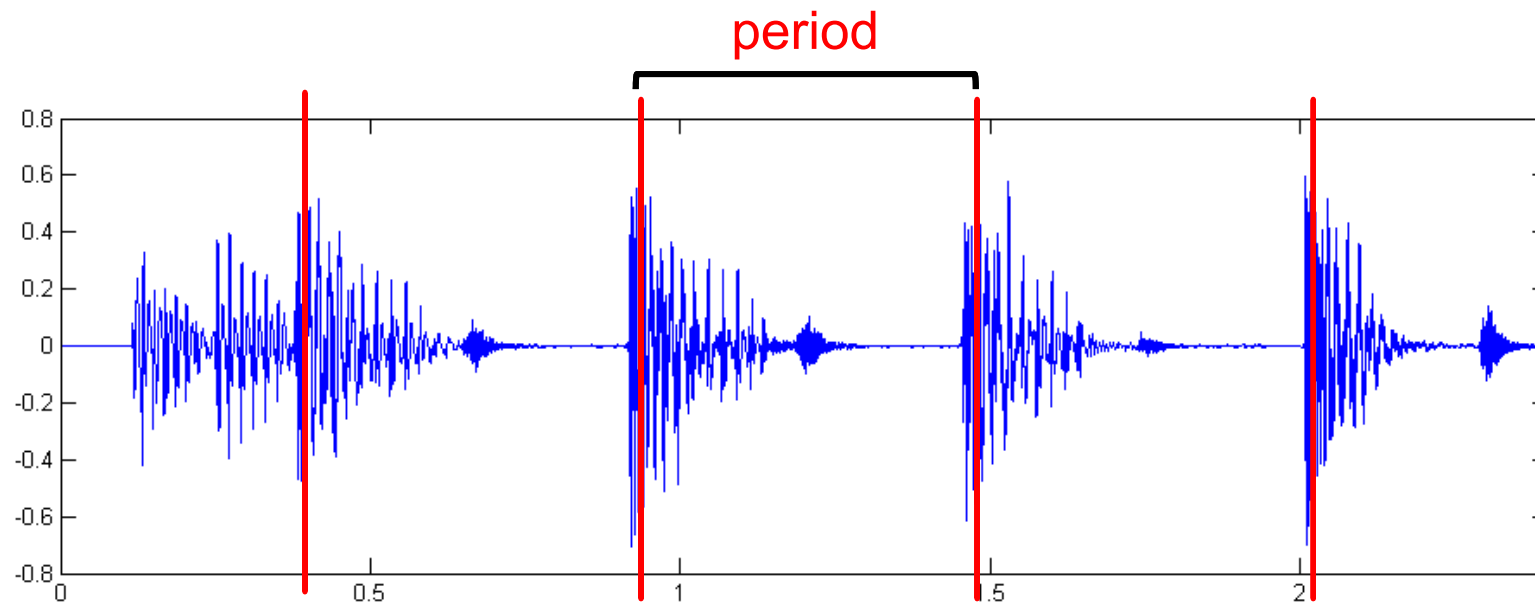
Tempo Estimation and Beat Tracking

Tasks

- Onset detection
- Beat tracking
- Tempo estimation

Tempo := $60 / \text{period}$

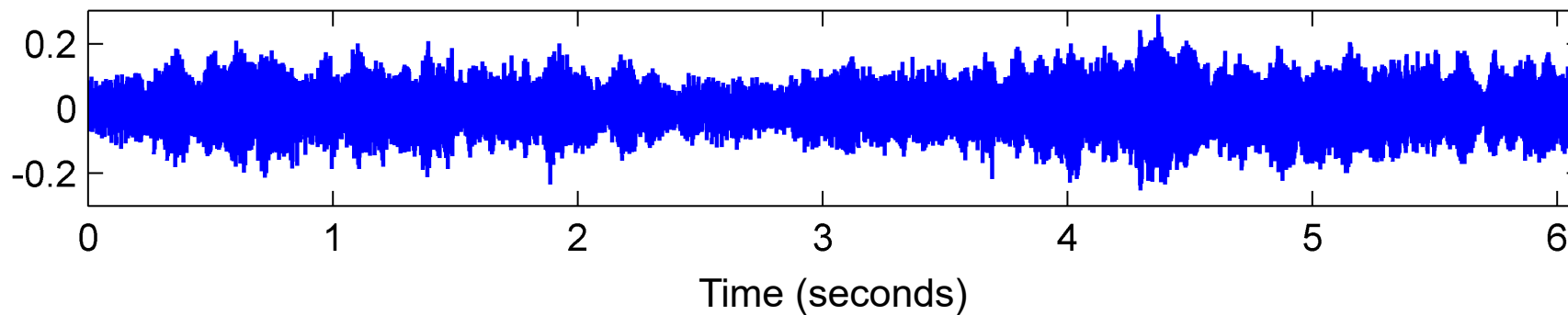
Beats per minute (BPM)



Onset Detection (Spectral Flux)



Audio recording

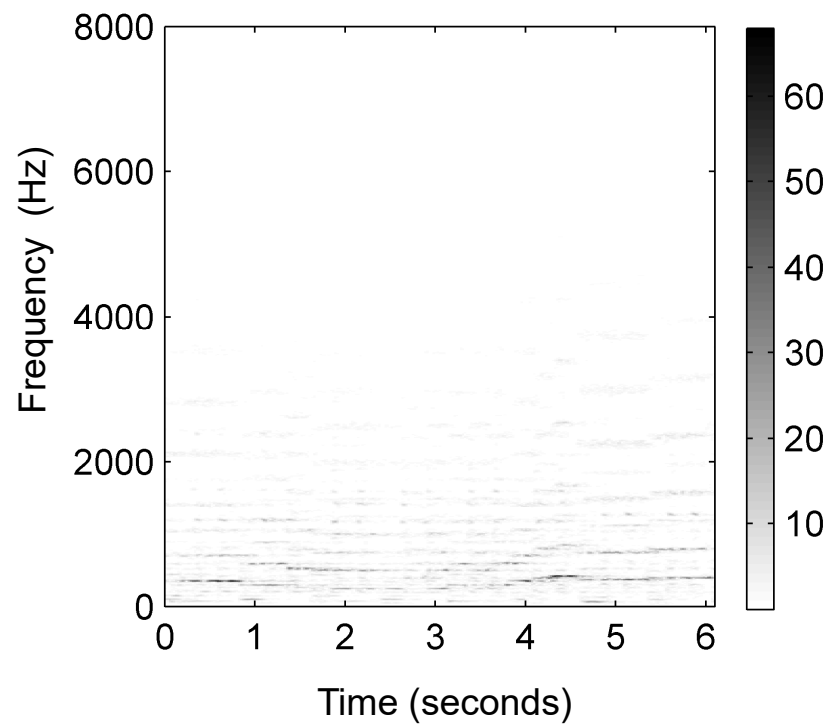


Onset Detection (Spectral Flux)

Steps:

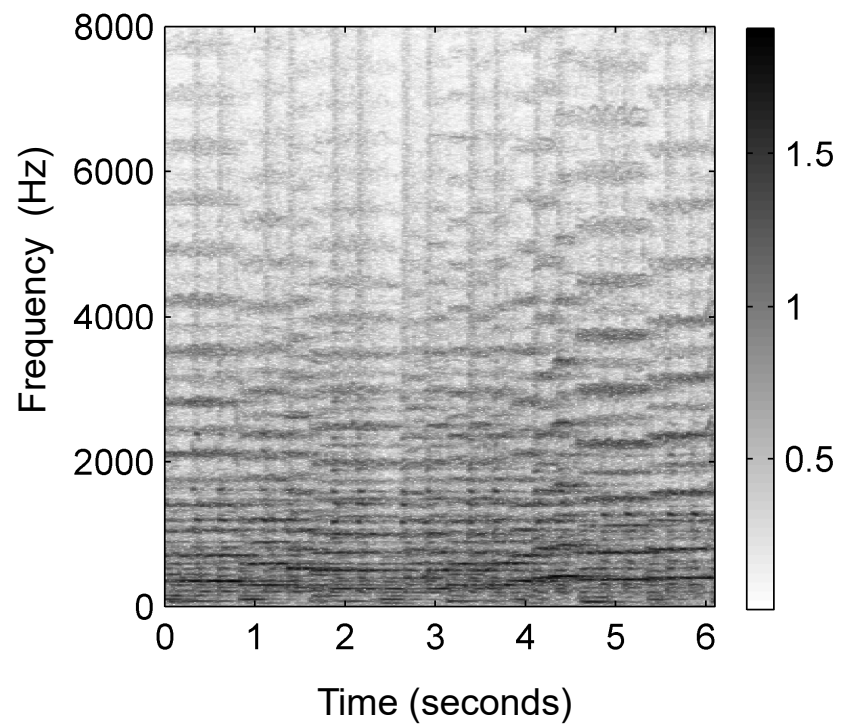
1. Spectrogram

Magnitude spectrogram $|X|$



Onset Detection (Spectral Flux)

Compressed spectrogram Y

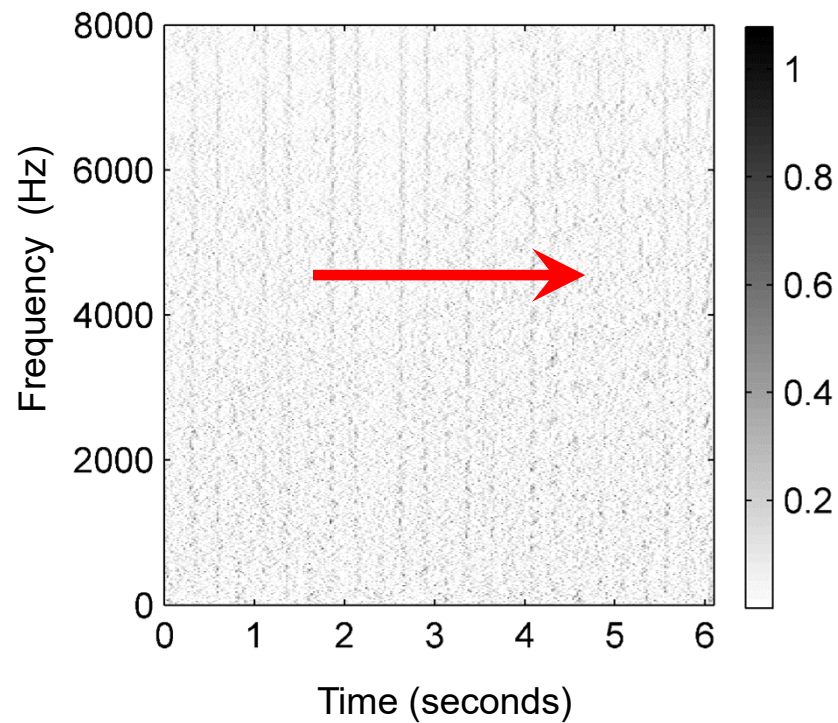


Steps:

1. Spectrogram
2. Logarithmic compression

Onset Detection (Spectral Flux)

Spectral difference



Steps:

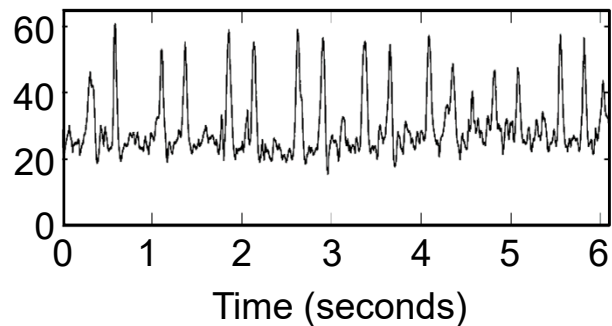
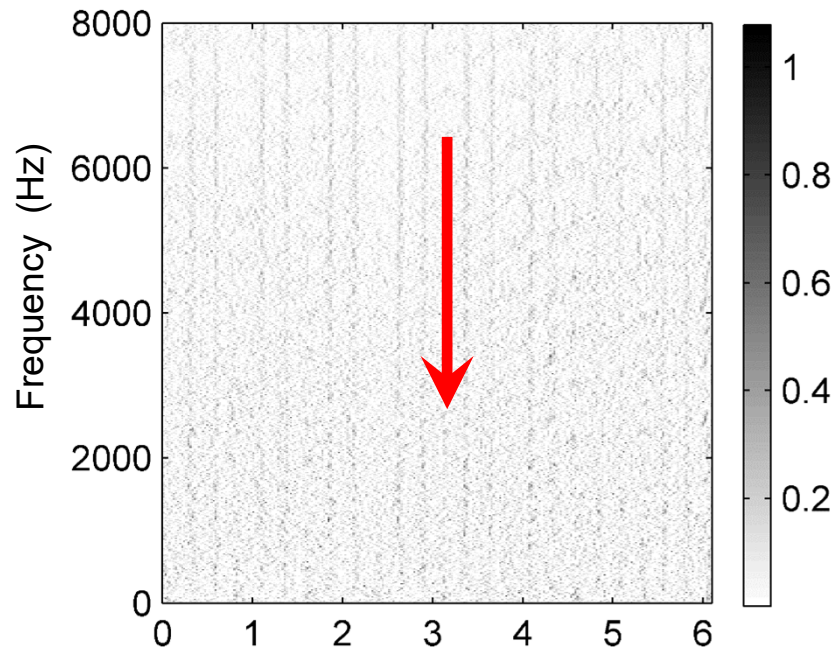
1. Spectrogram
2. Logarithmic compression
3. Differentiation & half wave rectification

Onset Detection (Spectral Flux)

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation & half wave rectification
4. Accumulation

Spectral difference



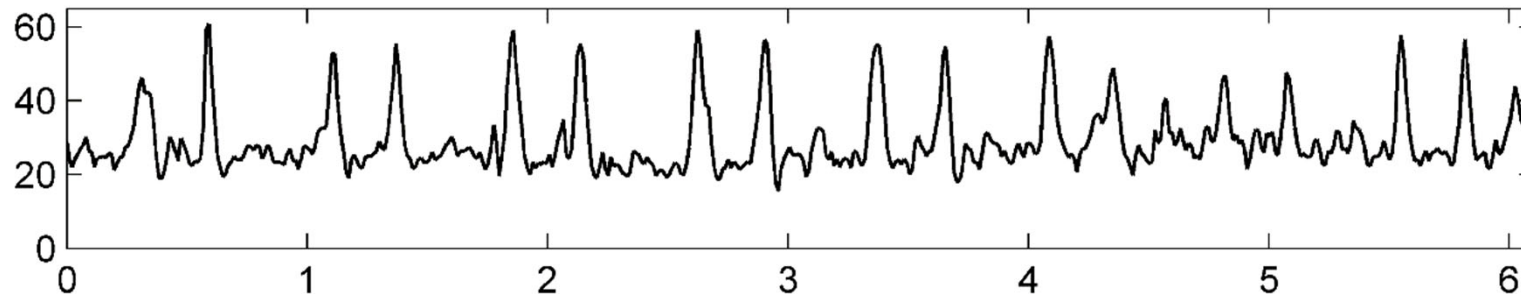
Novelty curve

Onset Detection (Spectral Flux)

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation & half wave rectification
4. Accumulation

Novelty function



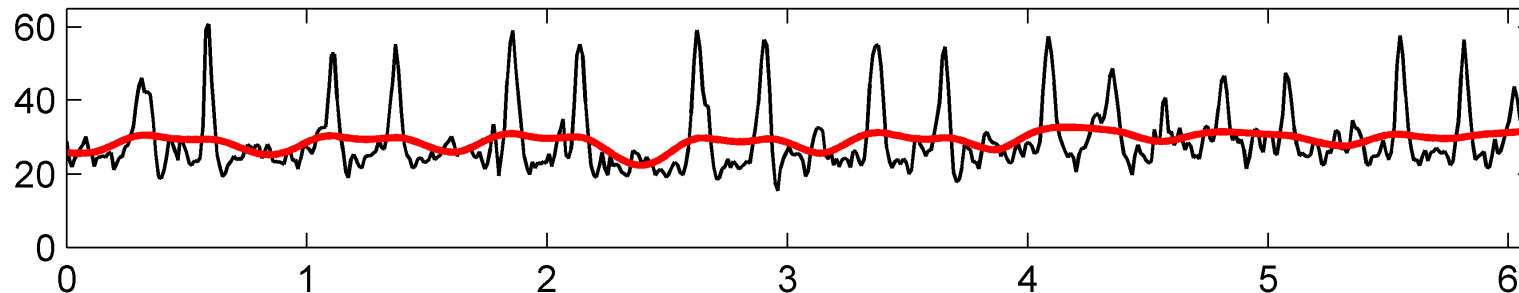
Onset Detection (Spectral Flux)

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation & half wave rectification
4. Accumulation
5. Normalization

Novelty function

Substraction of local average

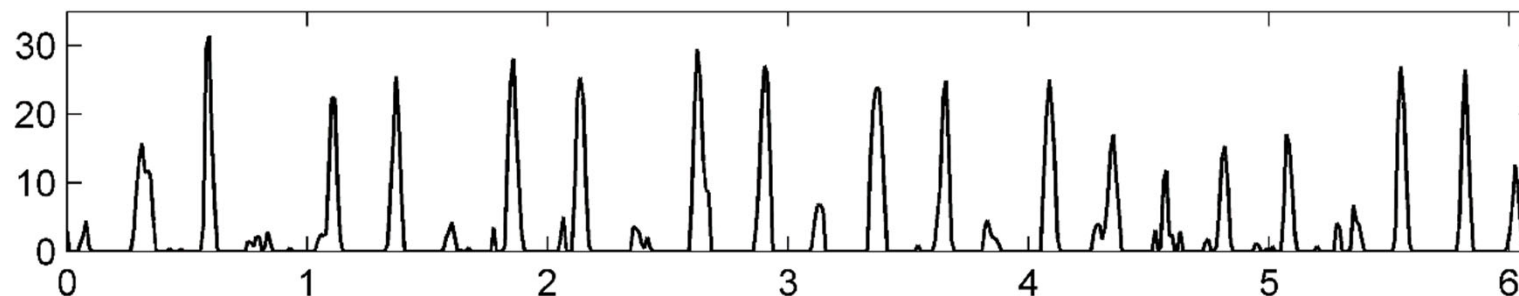


Onset Detection (Spectral Flux)

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation & half wave rectification
4. Accumulation
5. Normalization

Normalized novelty function



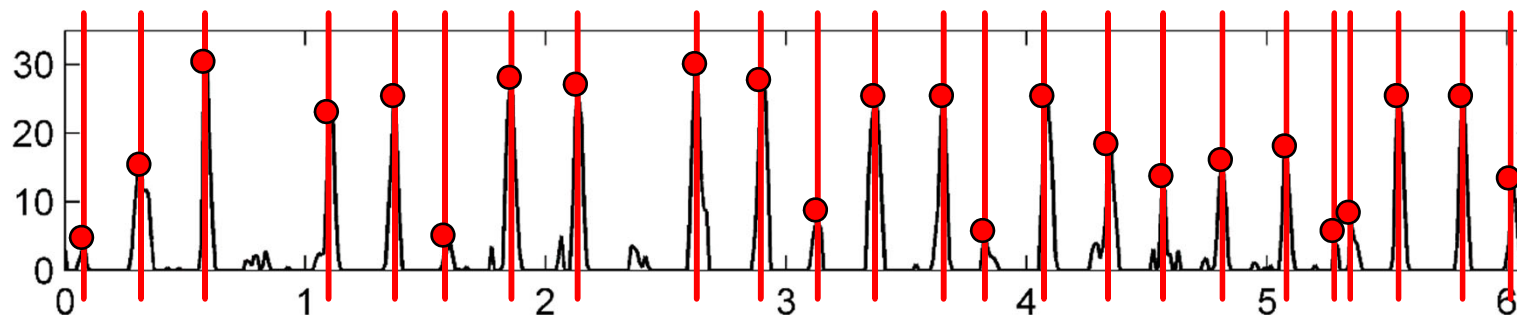
Onset Detection (Spectral Flux)

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation & half wave rectification
4. Accumulation
5. Normalization

Normalized novelty function

Peak positions indicate beat candidates



Onset Detection (Spectral Flux)

Deep Learning Approaches:

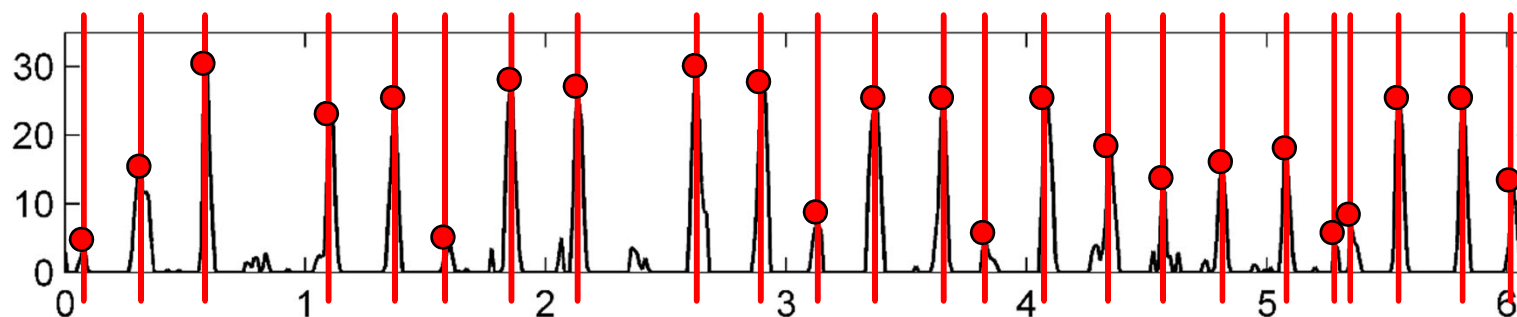
1. Input representation
2. Sigmoid activation
3. Convolution & rectified linear unit (ReLU)
4. Pooling
5. Convolution & ReLU

Steps:

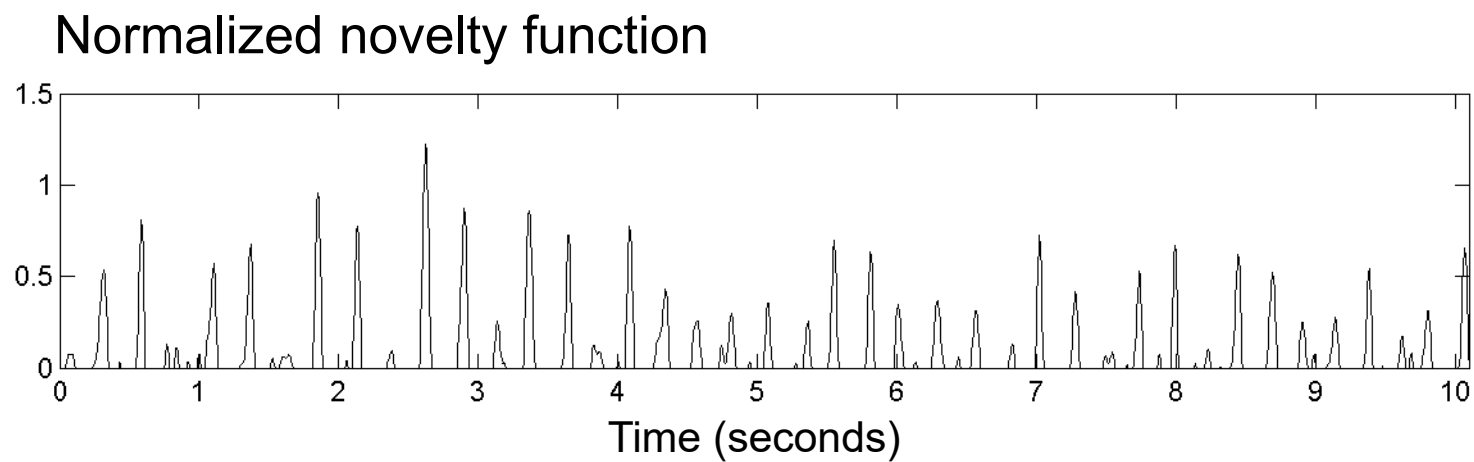
1. Spectrogram
2. Logarithmic compression
3. Differentiation & half wave rectification
4. Accumulation
5. Normalization

Normalized novelty function

Peak positions indicate beat candidates

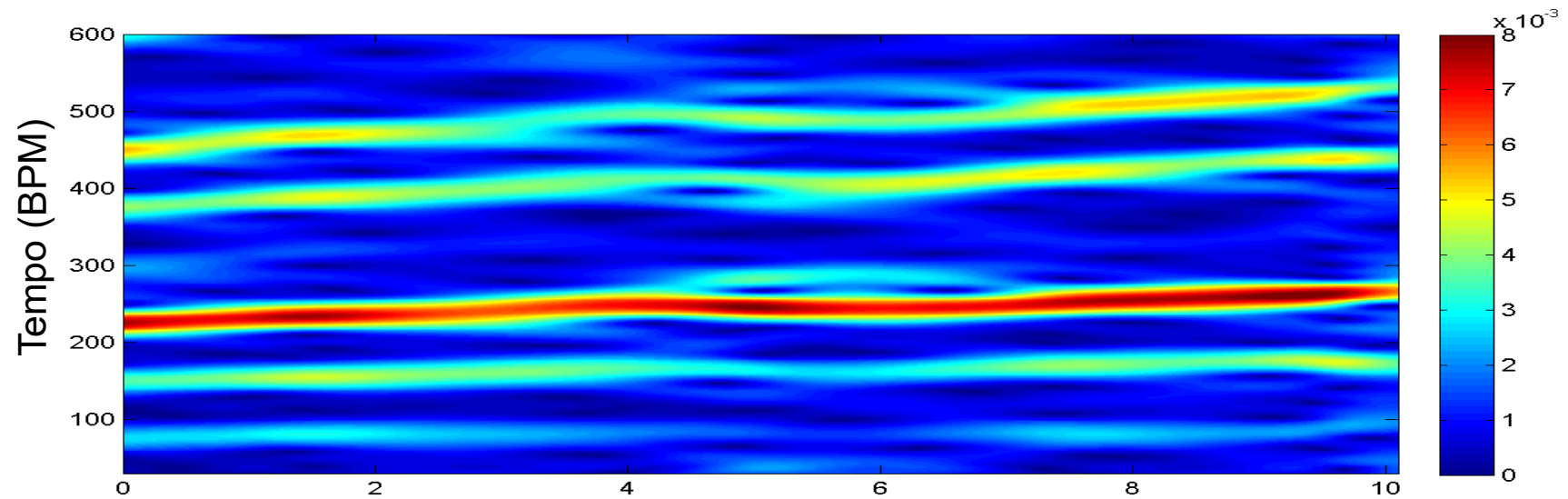


Local Pulse and Tempo Tracking

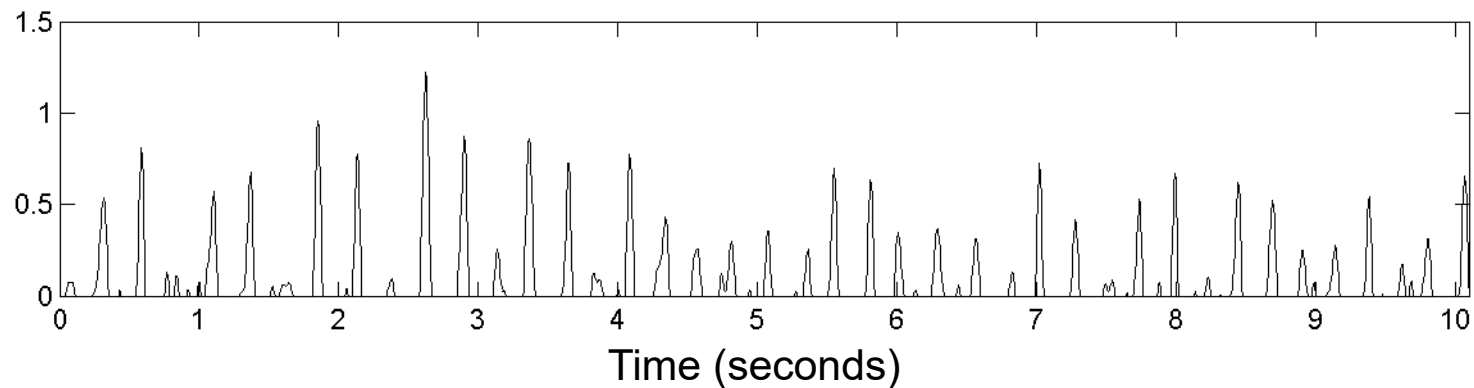


Local Pulse and Tempo Tracking

Fourier temogram (STFT of novelty function)

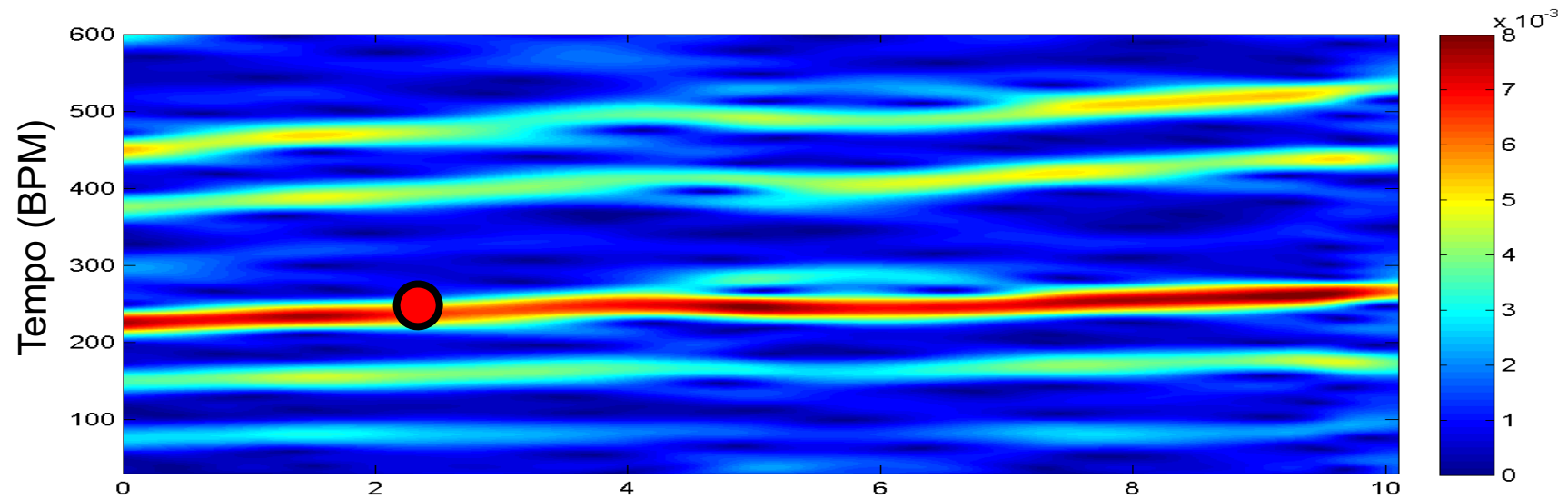


Normalized novelty function

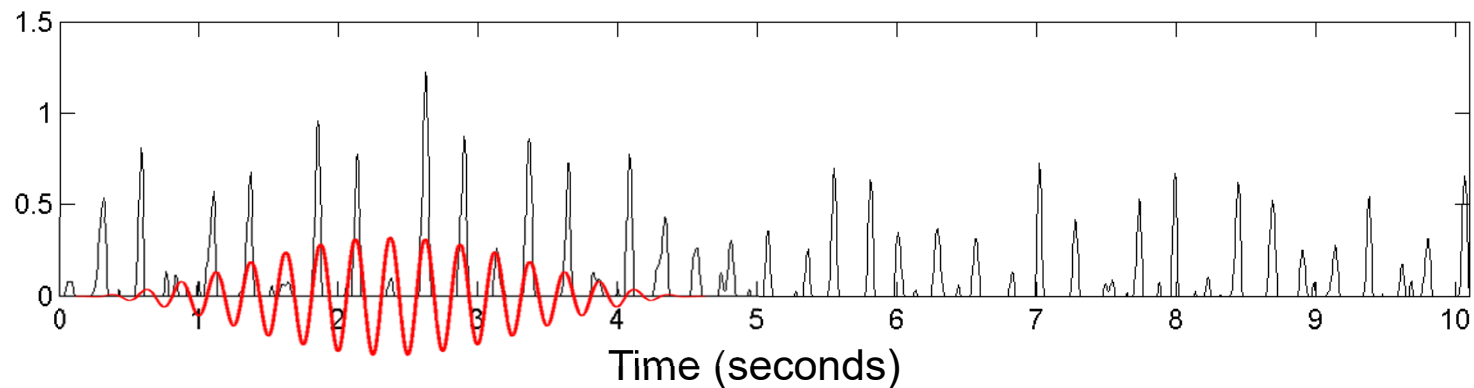


Local Pulse and Tempo Tracking

Fourier temogram (STFT of novelty function)

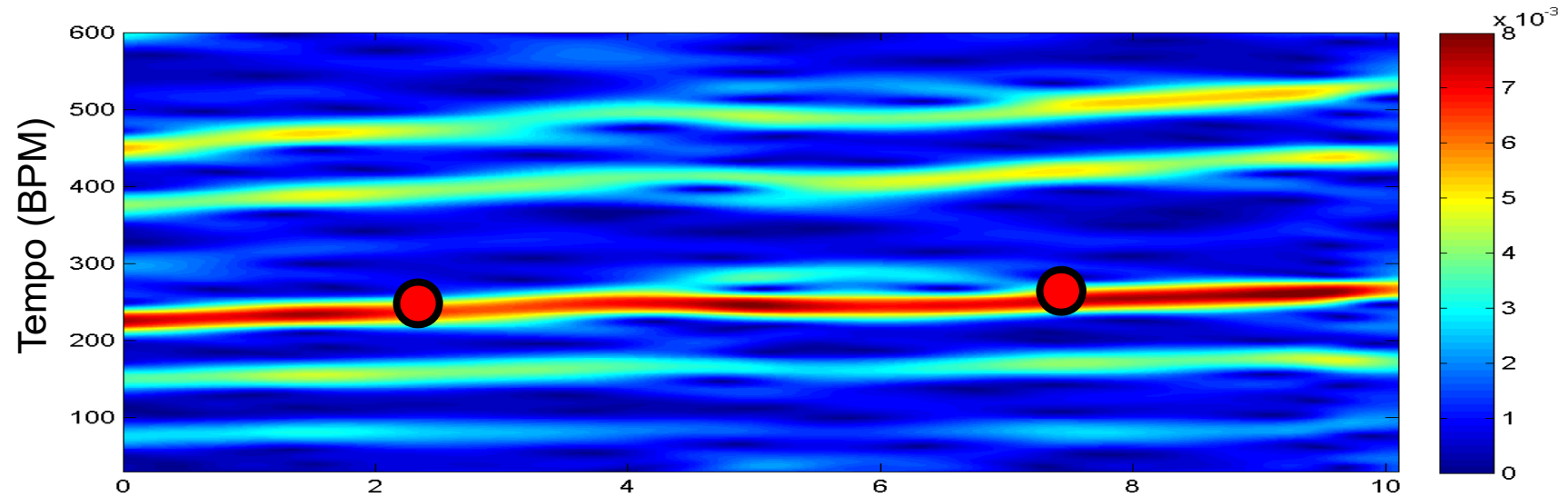


Optimizing local periodicity kernel

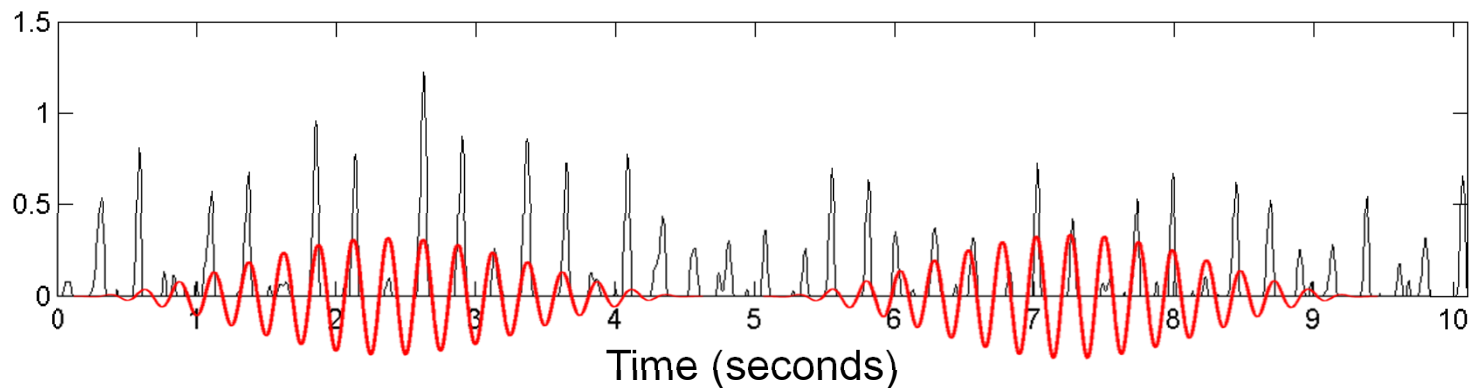


Local Pulse and Tempo Tracking

Fourier temogram (STFT of novelty function)

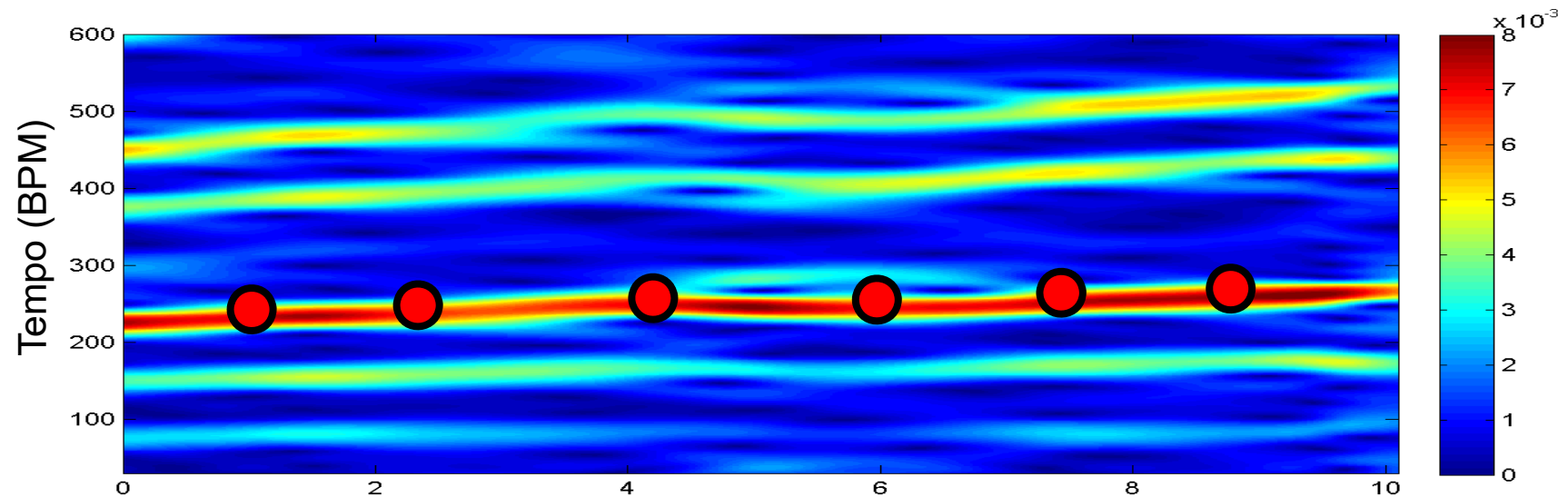


Optimizing local periodicity kernel

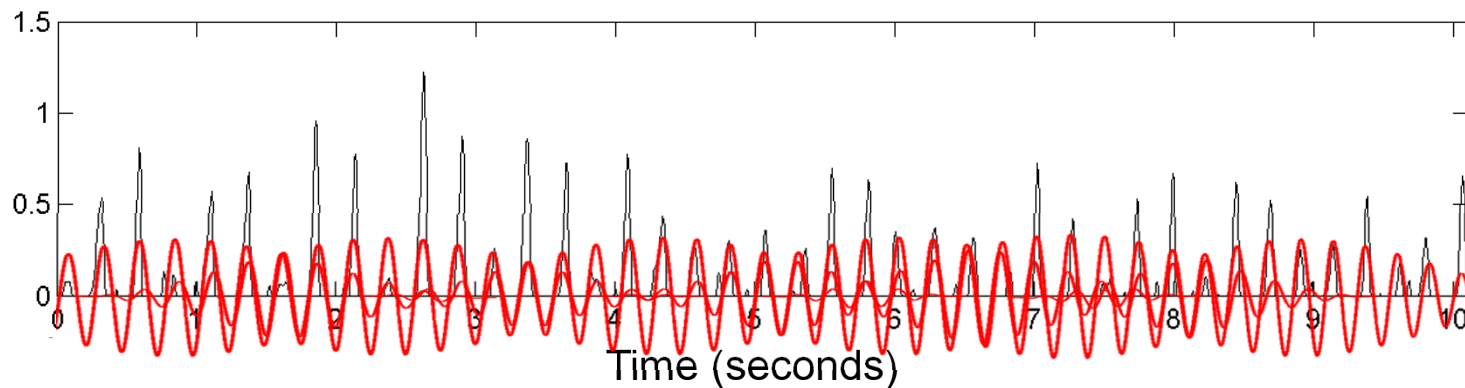


Local Pulse and Tempo Tracking

Fourier temogram (STFT of novelty function)

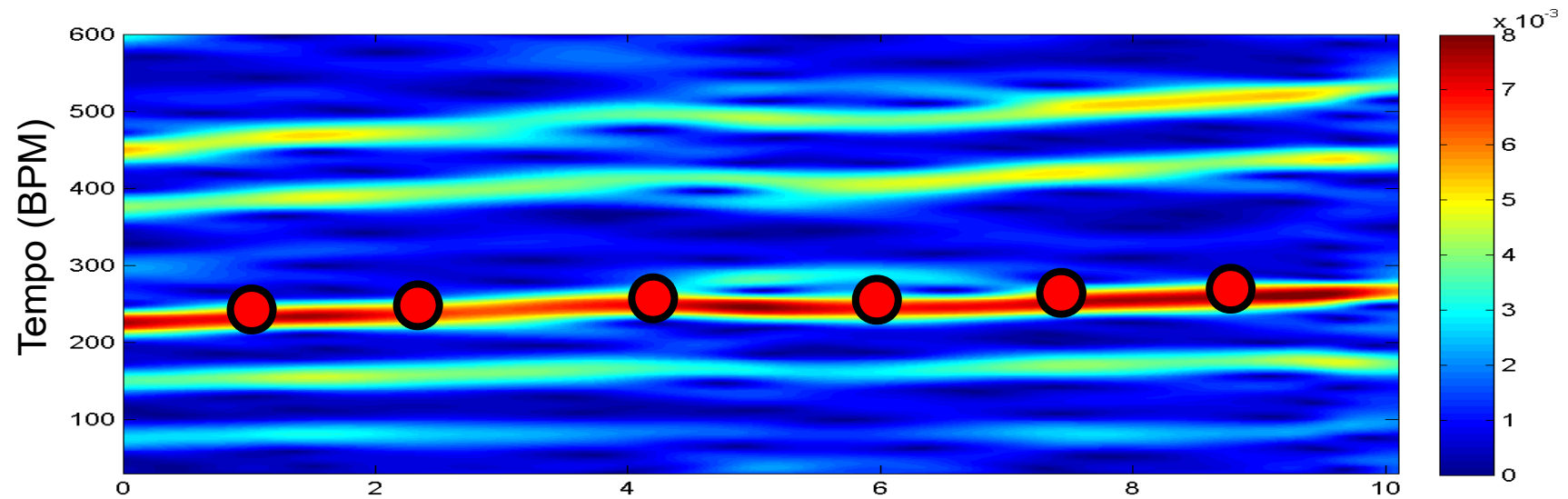


Optimizing local periodicity kernel

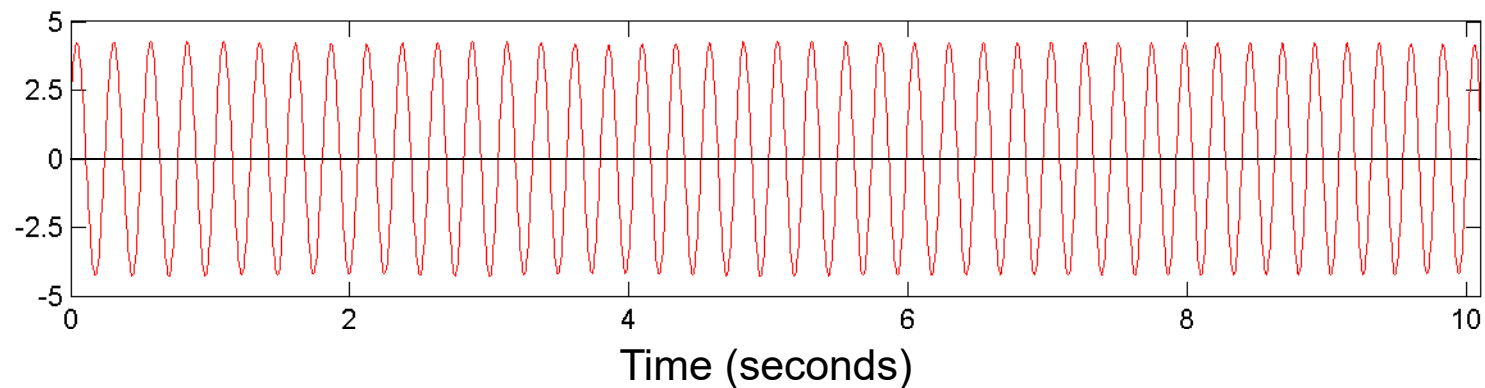


Local Pulse and Tempo Tracking

Fourier temogram (STFT of novelty function)

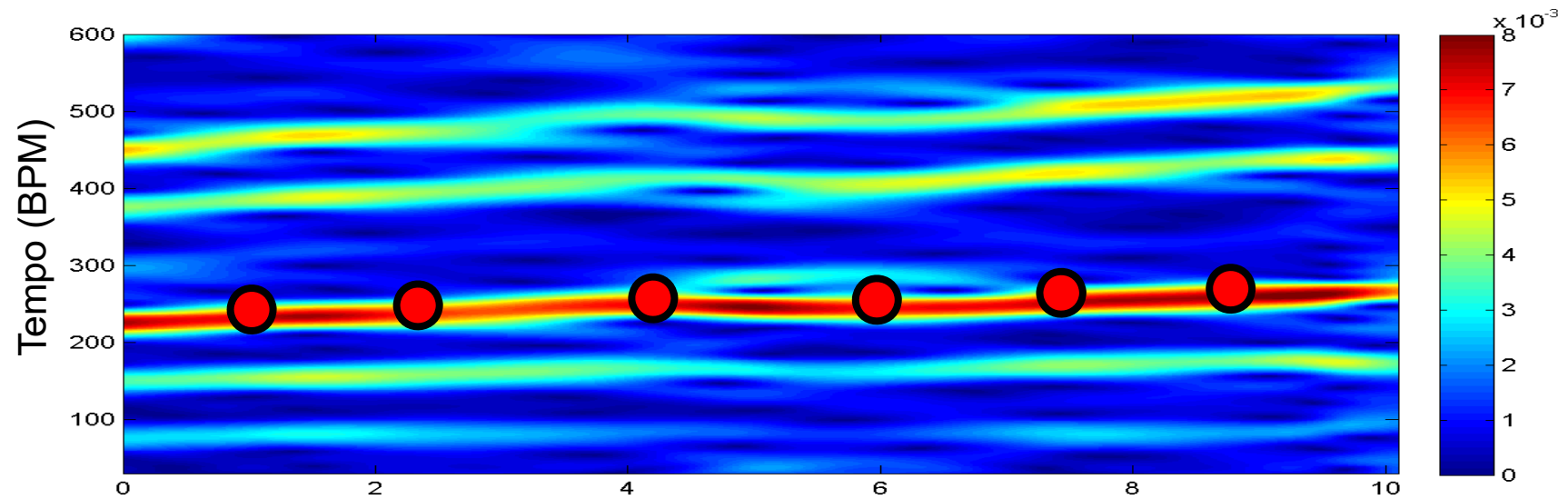


Accumulation of kernels

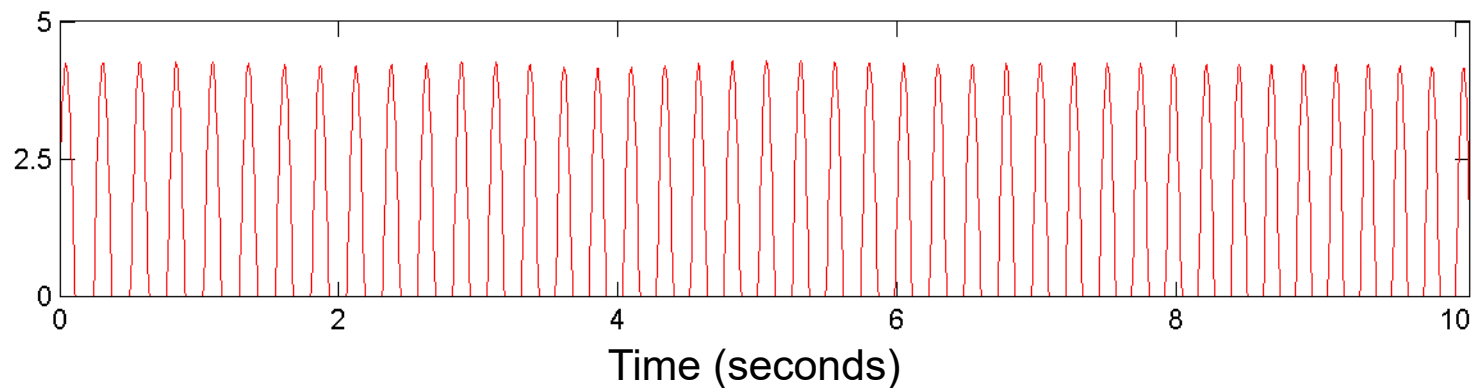


Local Pulse and Tempo Tracking

Fourier temogram (STFT of novelty function)

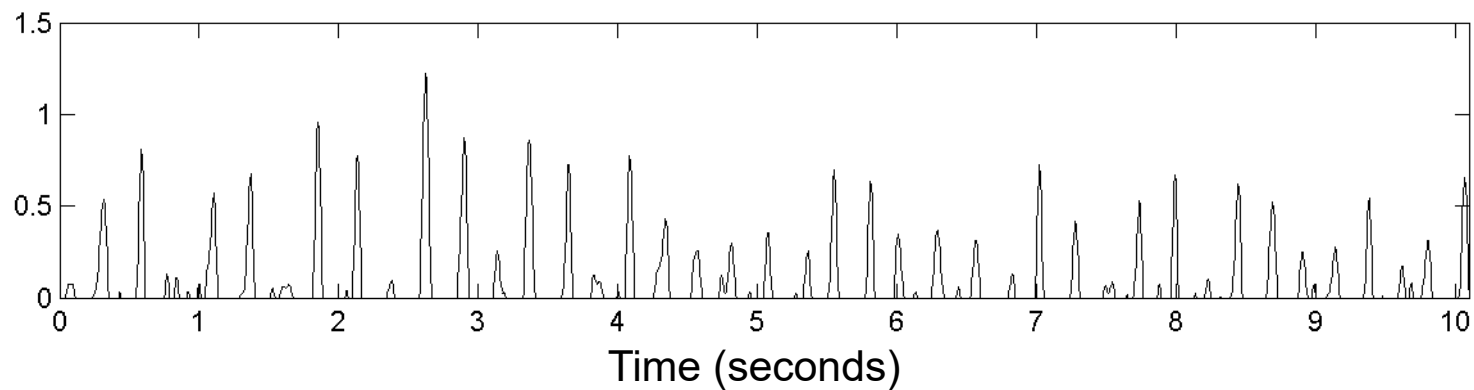


Halfwave rectification

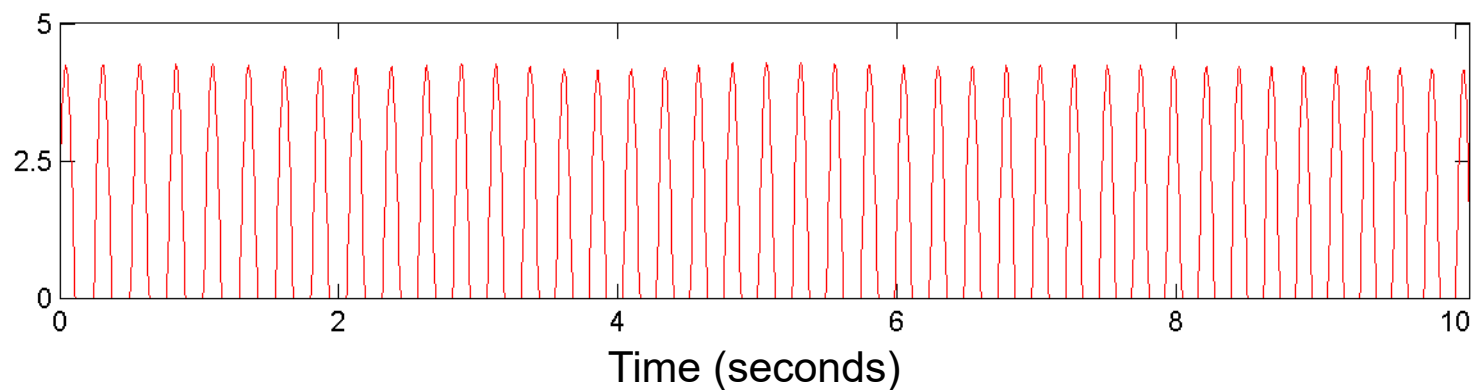


Local Pulse and Tempo Tracking

Novelty Curve



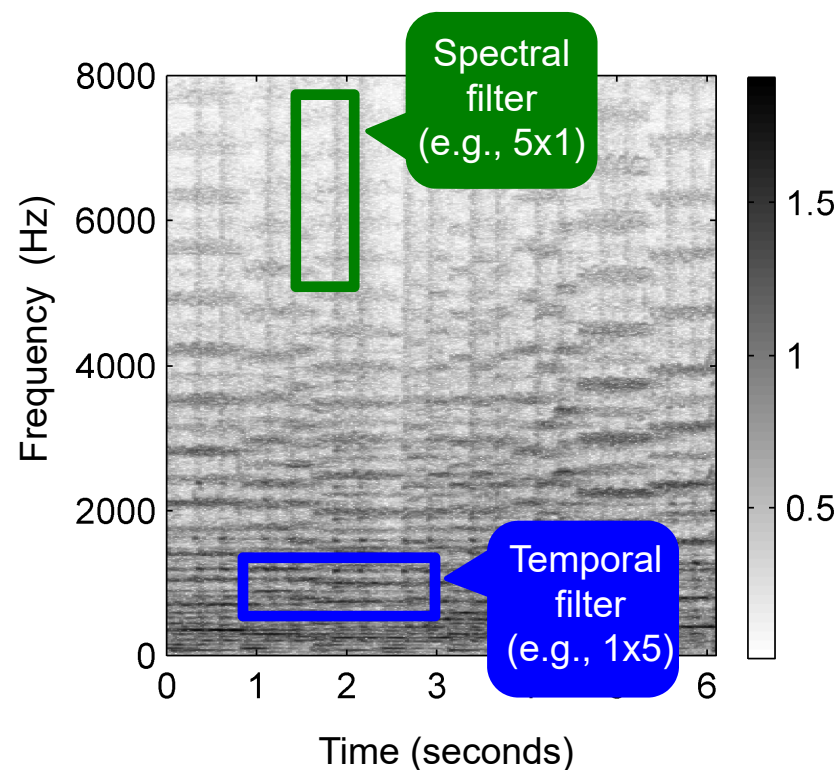
Predominant Local Pulse (PLP)



Local Pulse and Tempo Tracking

Deep Learning Approaches:

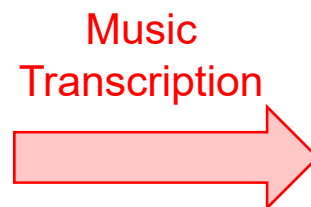
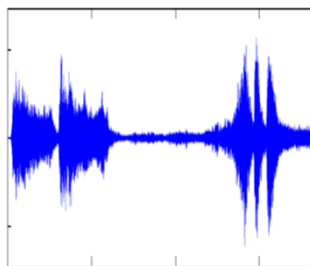
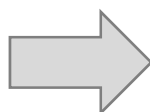
- End-to-end approach
 - Input: Short audio snippets
 - Output: Tempo value
- DL architecture inspired by traditional engineering
 - Layers and activation functions
 - Shape of convolutional kernels



Schreiber, Müller: A Single-Step Approach to Musical Tempo Estimation Using a Convolutional Neural Network, ISMIR 2018.

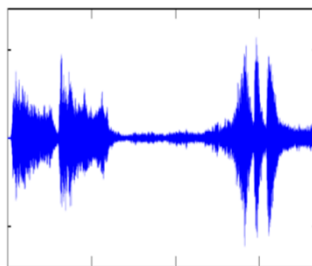
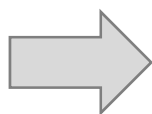
Automatic Music Transcription

Task: Convert a music recording into sheet music

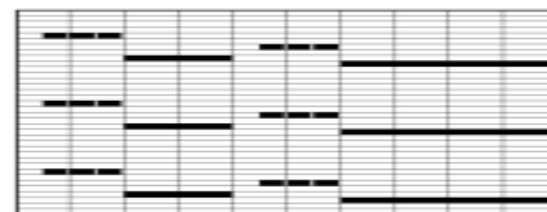


Automatic Music Transcription

Task: Convert a music recording into sheet music
(or another symbolic music representation)

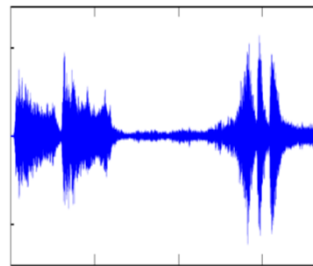
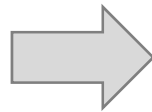


Music
Transcription



Automatic Music Transcription

Task: Convert a music recording into sheet music
(or another symbolic music representation)



Music
Transcription



Multitask learning for estimating

- pitches,
- note onsets & offsets,
- beat & measure positions,
- musical voices & instrumentation,
- pedalling, dynamics, ...

Why is Music Processing Challenging?

Example: Chopin, Mazurka Op. 63 No. 3



Mazurka.

F. CHOPIN. Op. 63, № 3.

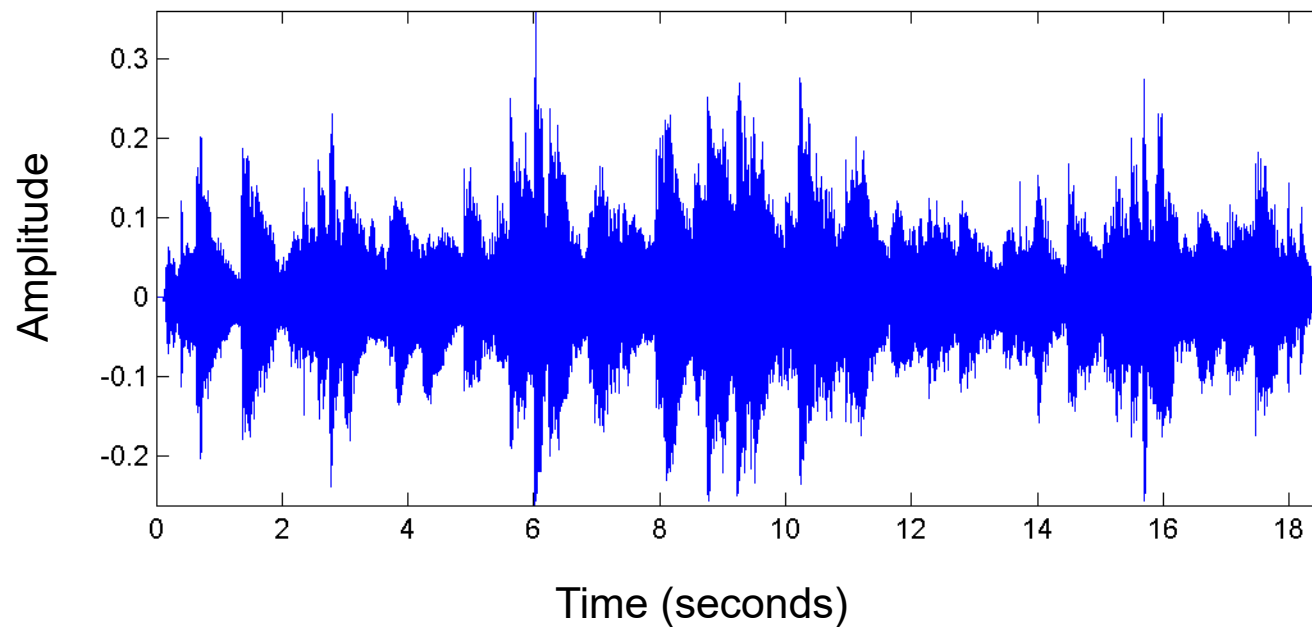
41. Allegretto. *p*

The image shows two systems of musical notation for measures 41-50 of Chopin's Mazurka Op. 63 No. 3. The first system covers measures 41-46, and the second system covers measures 47-50. The music is in 3/4 time with a key signature of three sharps (F#, C#, G#). The tempo is marked 'Allegretto' and the dynamics are 'p' (piano). The notation includes treble and bass staves with various musical symbols such as slurs, accents, and fingerings. The bass line features a repeating rhythmic pattern of eighth notes, marked with 'Ped.' and asterisks. The melody in the treble staff includes triplets and slurs. The second system continues the piece with similar notation and a final measure marked with a double bar line.

Why is Music Processing Challenging?

Example: Chopin, Mazurka Op. 63 No. 3

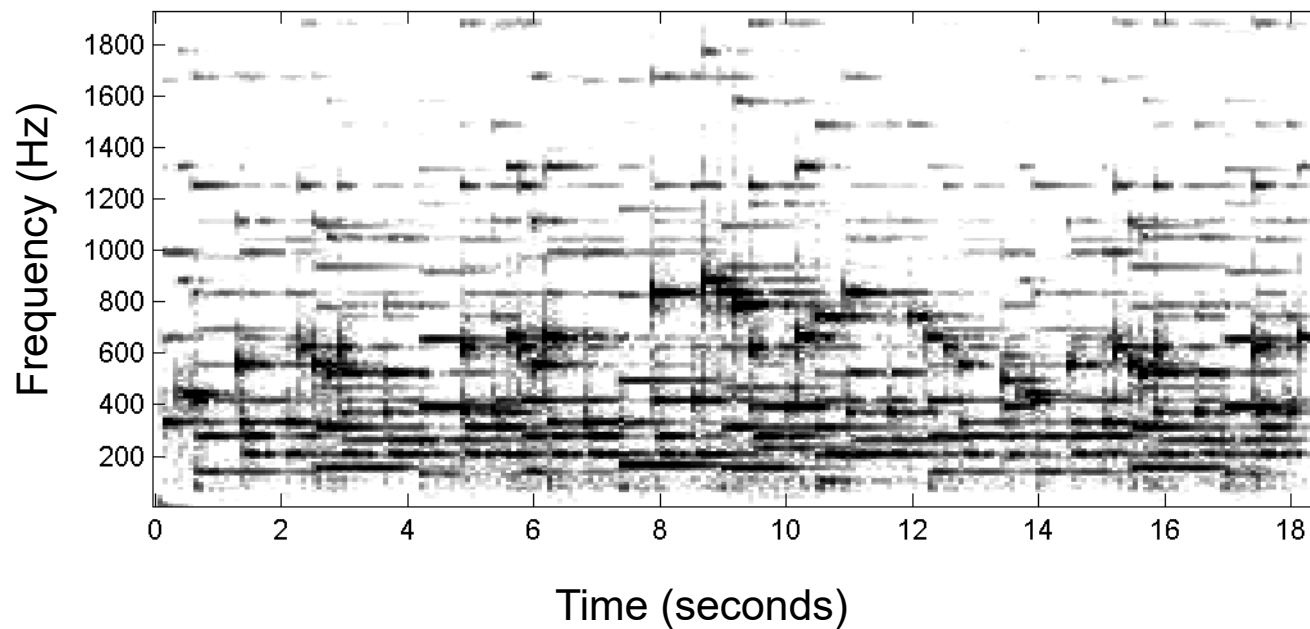
- Waveform



Why is Music Processing Challenging?

Example: Chopin, Mazurka Op. 63 No. 3

- Waveform / Spectrogram



Why is Music Processing Challenging?

Example: Chopin, Mazurka Op. 63 No. 3

- Waveform / Spectrogram
- Performance
 - Tempo
 - Dynamics
 - Note deviations
 - Sustain pedal

Why is Music Processing Challenging?

Example: Chopin, Mazurka Op. 63 No. 3

- Waveform / Spectrogram



- Performance

- Tempo
- Dynamics
- Note deviations
- Sustain pedal

A musical score for Chopin's Mazurka Op. 63 No. 3, showing two systems of music. The score is annotated with performance information: blue highlights for the main melody, red highlights for an additional melody line, and yellow highlights for the accompaniment. Fingerings and dynamics like 'p' and 'f' are also present.

- Polyphony



Main Melody



Additional melody line



Accompaniment

Source Separation

- Decomposition of audio stream into different sound sources
- Central task in digital signal processing
- “Cocktail party effect”

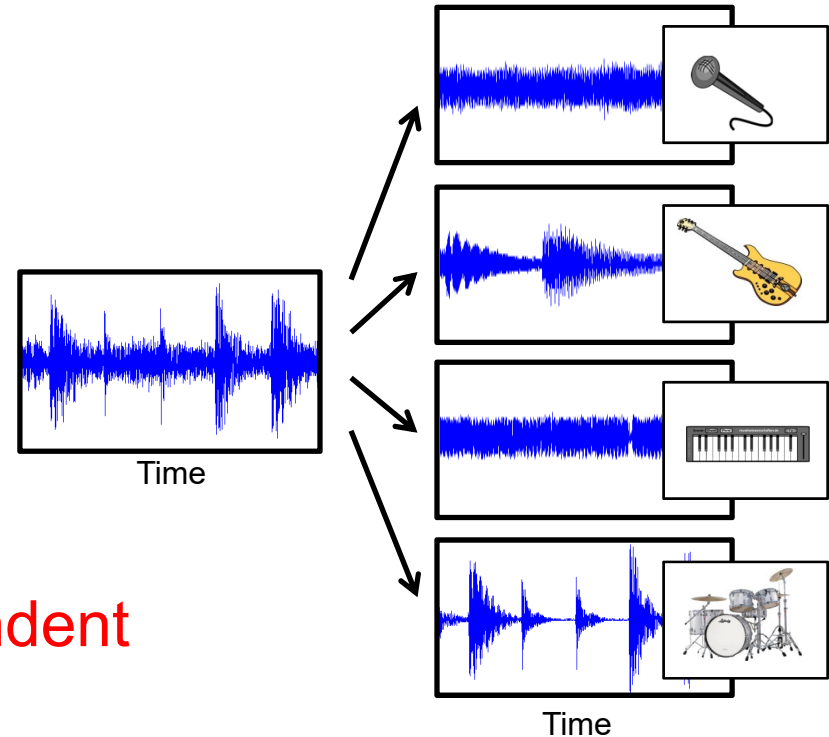


Source Separation

- Decomposition of audio stream into different sound sources
- Central task in digital signal processing
- “Cocktail party effect”
- Several input signals
- Sources are assumed to be statistically independent

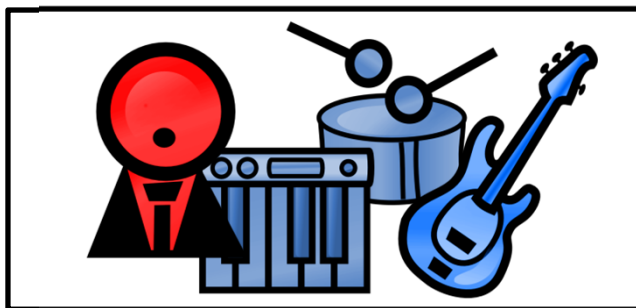
Source Separation (Music)

- Main melody, accompaniment, drum track
- Instrumental voices
- Individual note events
- Only mono or stereo
- Sources are often highly dependent

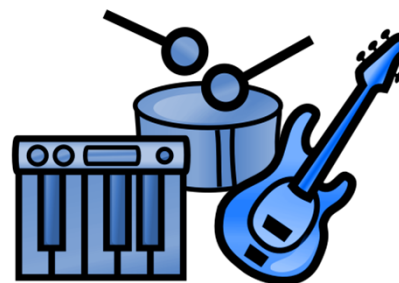


Singing Voice Extraction

Original Recording



Singing voice

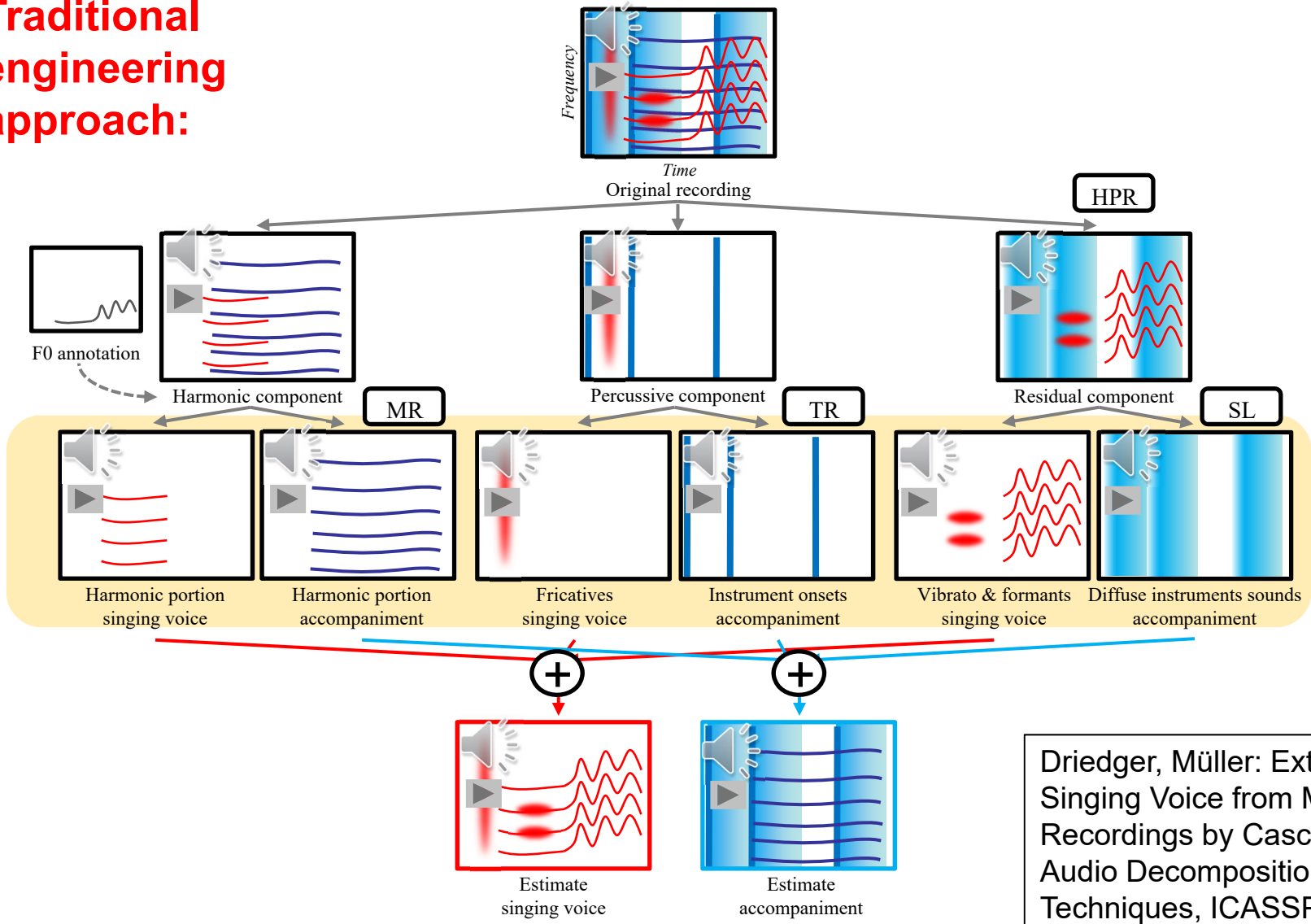


Accompaniment



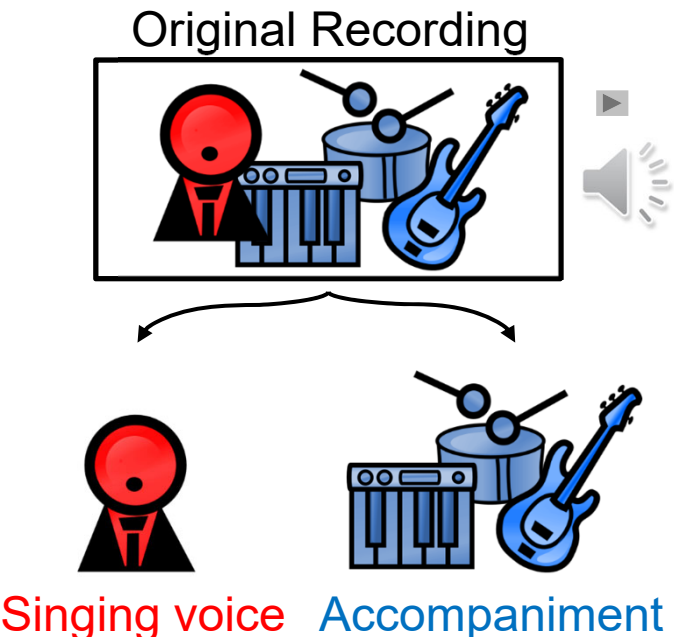
Singing Voice Extraction

Traditional engineering approach:



Driedger, Müller: Extracting Singing Voice from Music Recordings by Cascading Audio Decomposition Techniques, ICASSP 2015.

Singing Voice Extraction



**Deep learning
has lead to
breakthrough**

**Lecture 5:
Music
Source
Separation**

Reference voices:



Engineering approach:



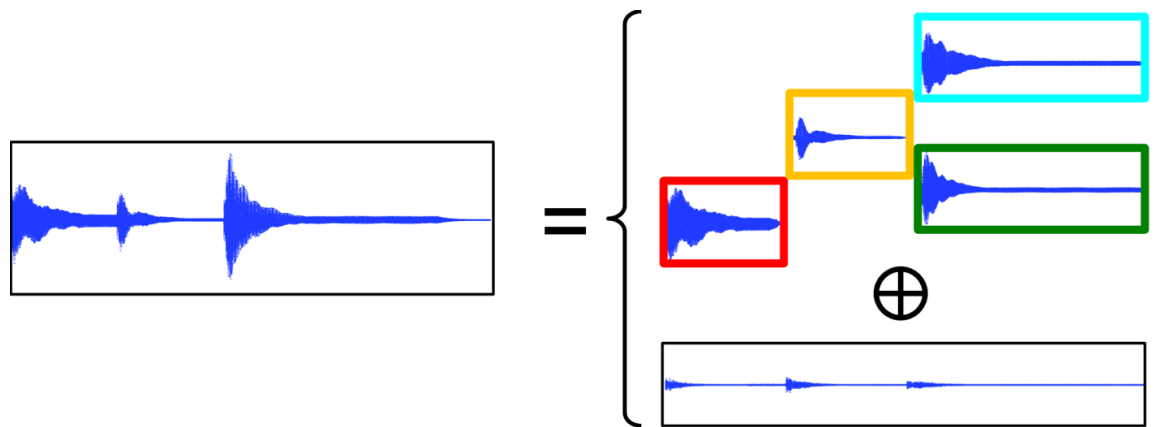
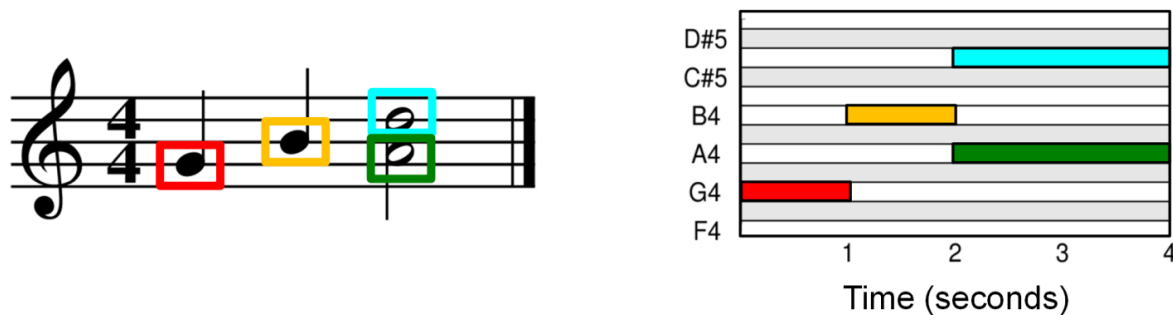
Deep learning approach:



Stöter, Uhlich Luitkus,
Mitsufuji: Open-Unmix – A
Reference Implementation
for Music Source
Separation, JOSS 2019.

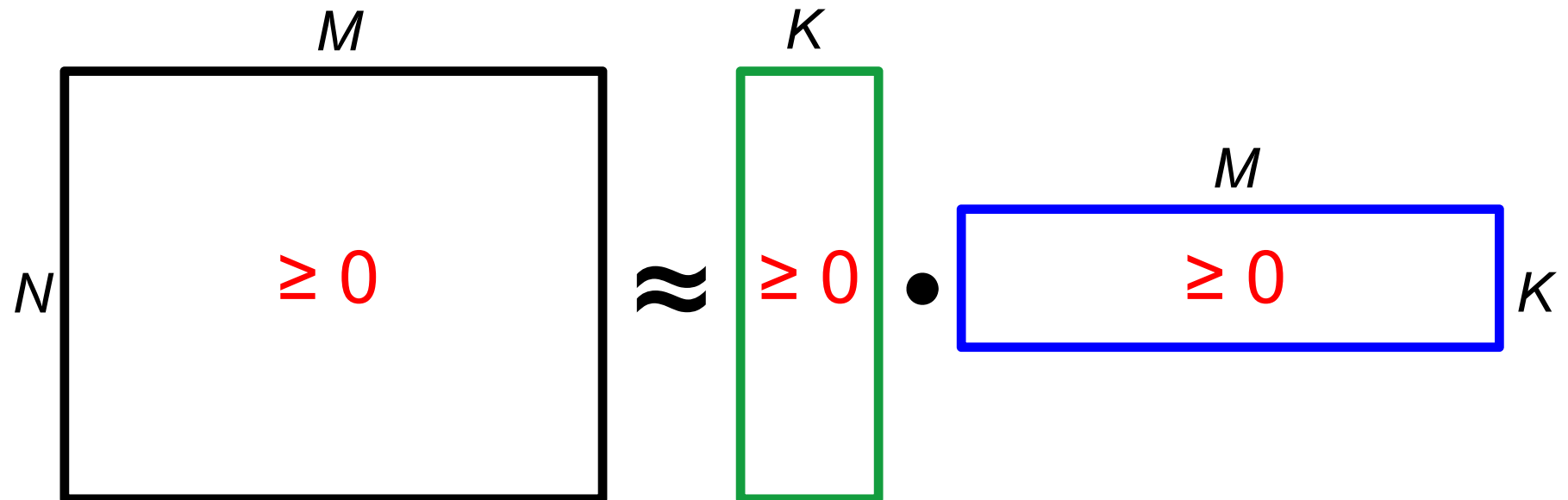
Score-Informed Audio Decomposition

Exploit musical score to support decomposition process

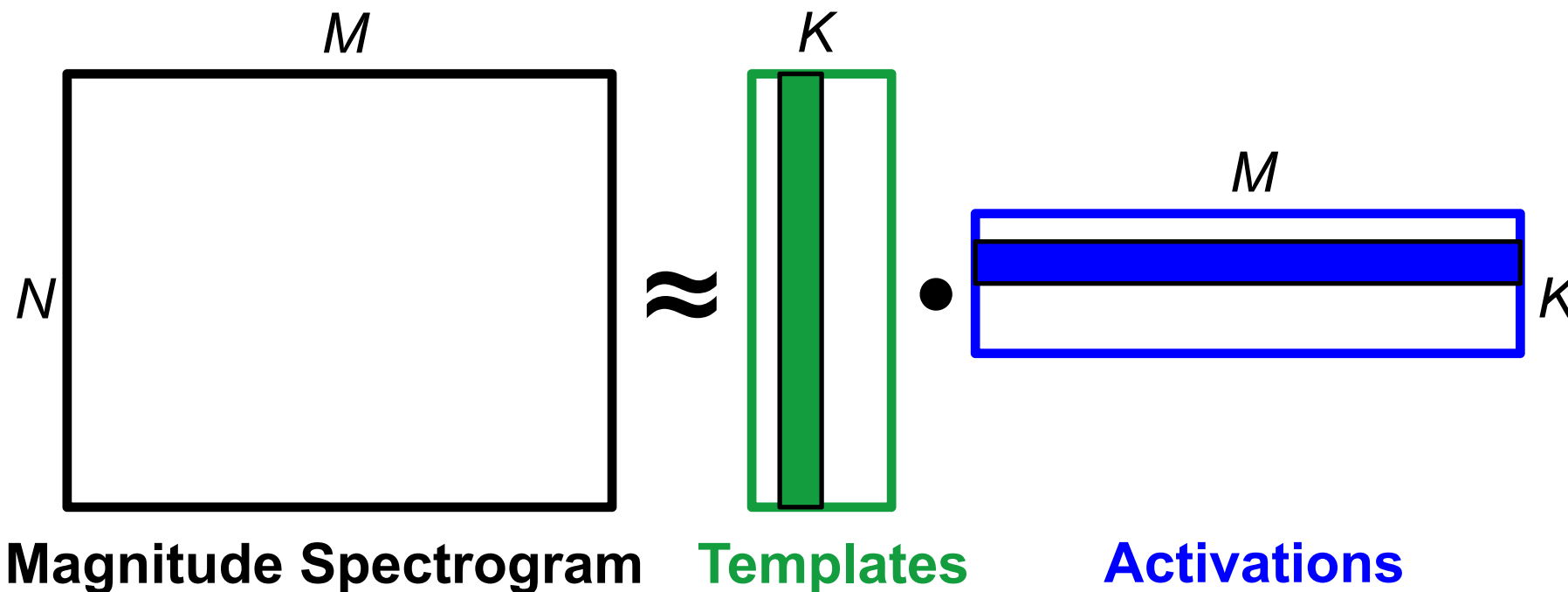


Ewert, Pardo, Müller, Plumbley: Score-Informed Source Separation for Musical Audio Recordings, IEEE SPM, 2014.

NMF (Nonnegative Matrix Factorization)



NMF (Nonnegative Matrix Factorization)



Templates: Pitch + Timbre

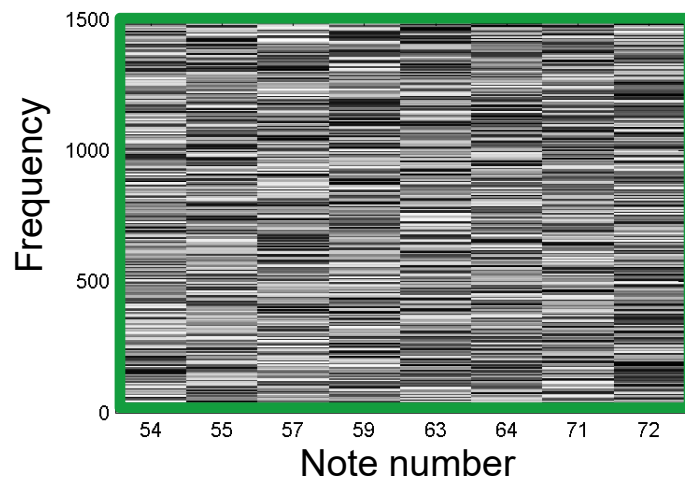
Activations: Onset time + Duration

“How does it sound”

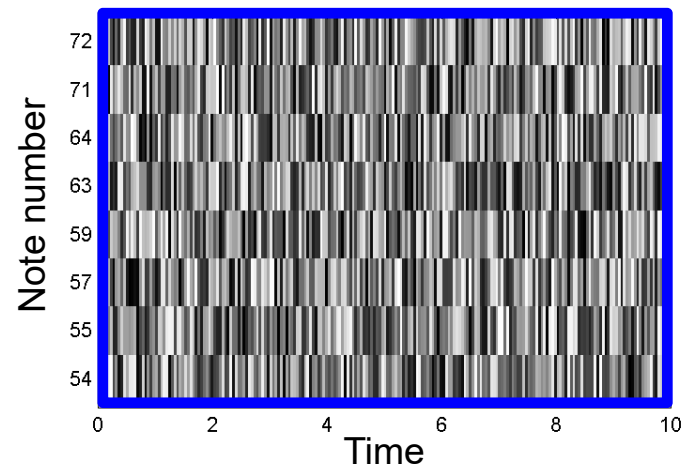
“When does it sound”

NMF-Decomposition

Initialized template



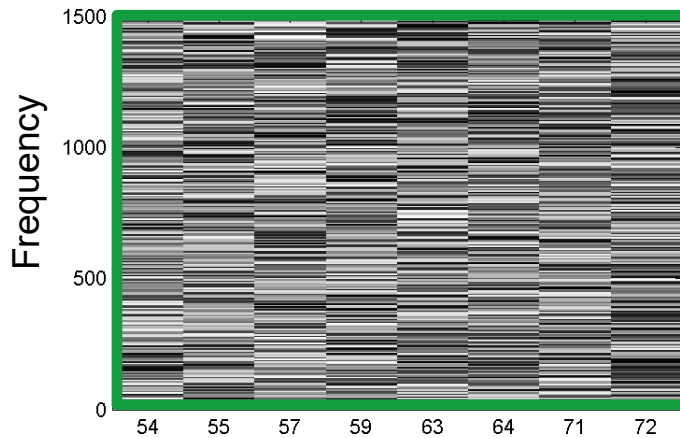
Initialized activations



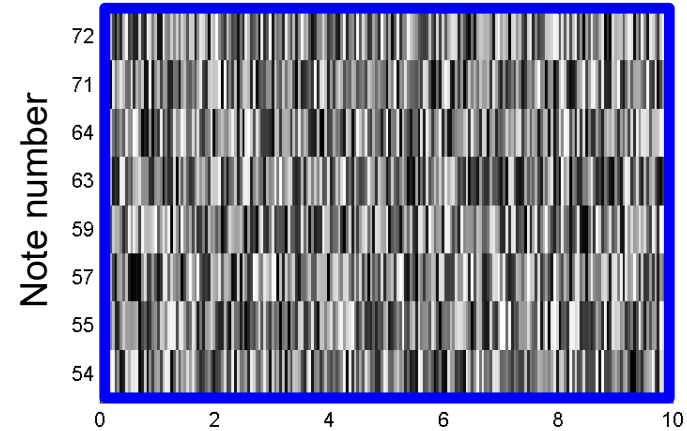
Random initialization

NMF-Decomposition

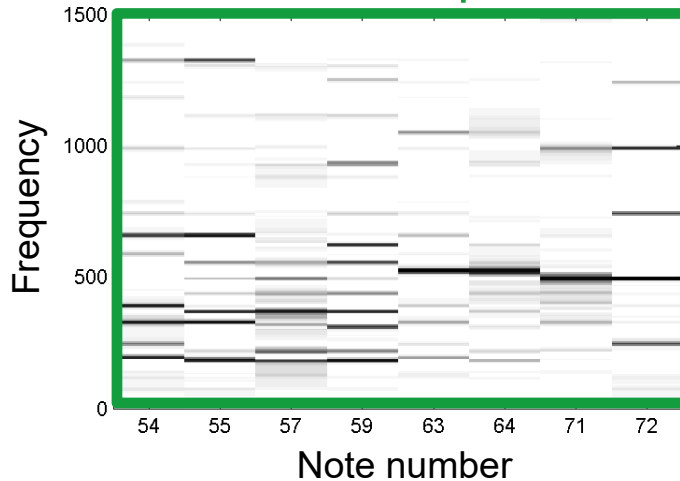
Initialized template



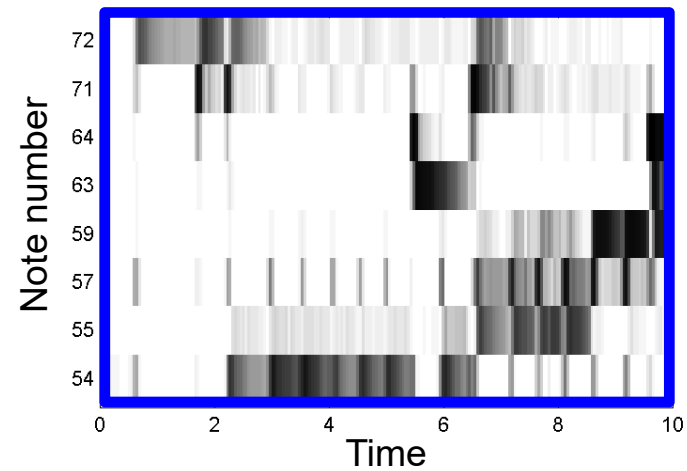
Initialized activations



Learnt templates



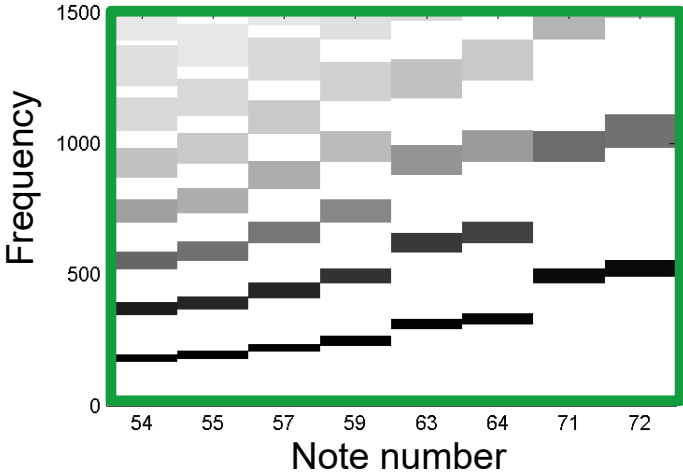
Learnt activations



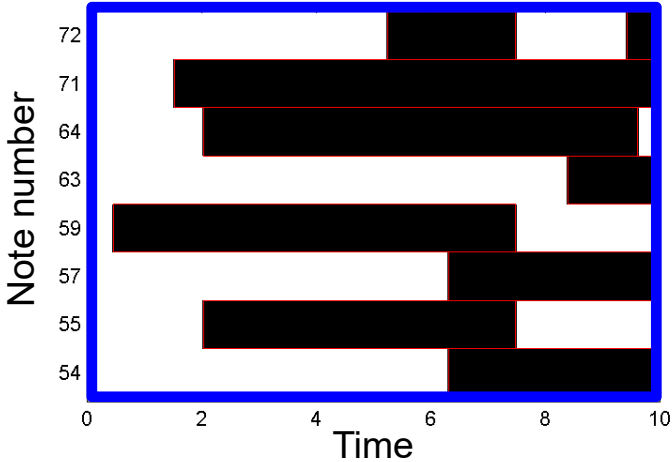
Random initialization → No semantic meaning

NMF-Decomposition

Initialized template

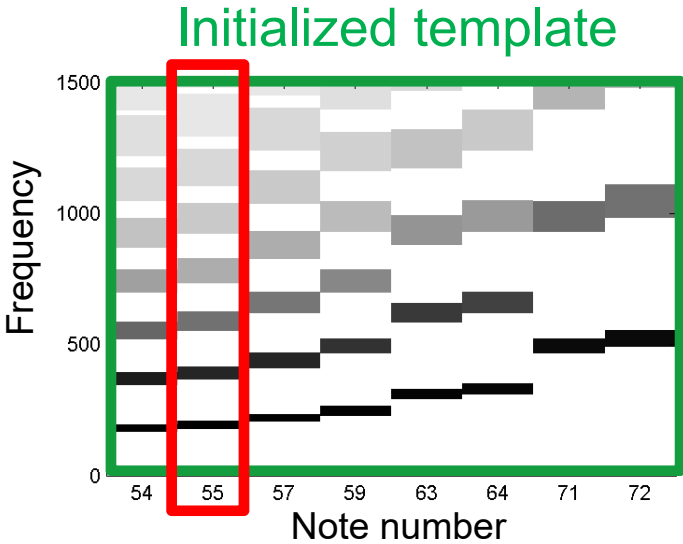


Initialized activations

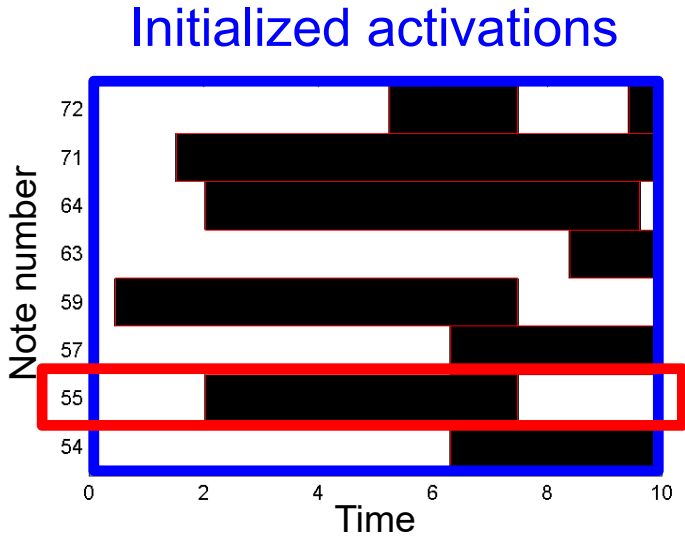


Constrained initialization

NMF-Decomposition



Template constraint for $p=55$

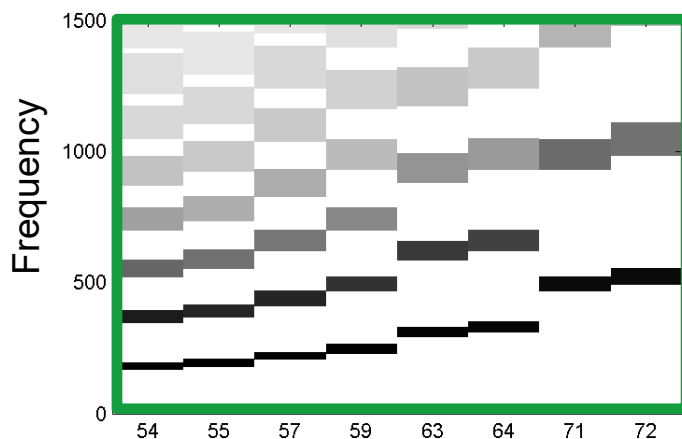


Activation constraints for $p=55$

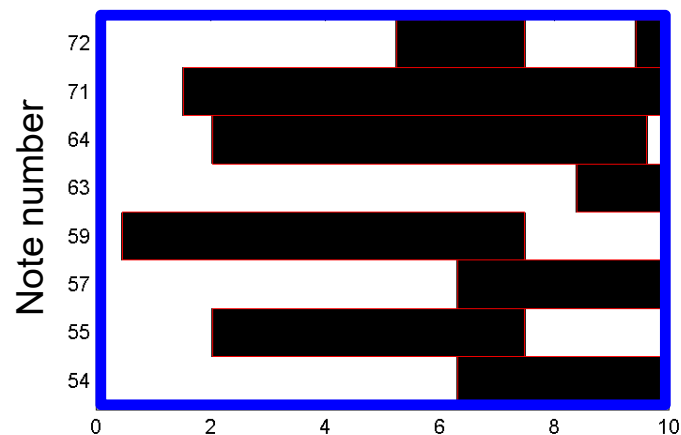
Constrained initialization

NMF-Decomposition

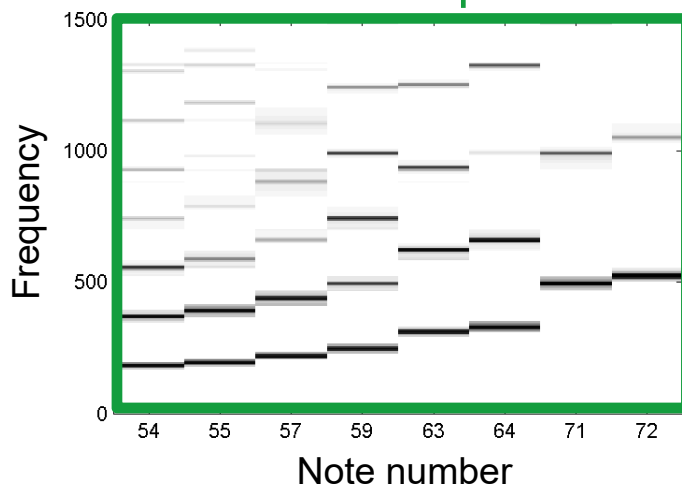
Initialized template



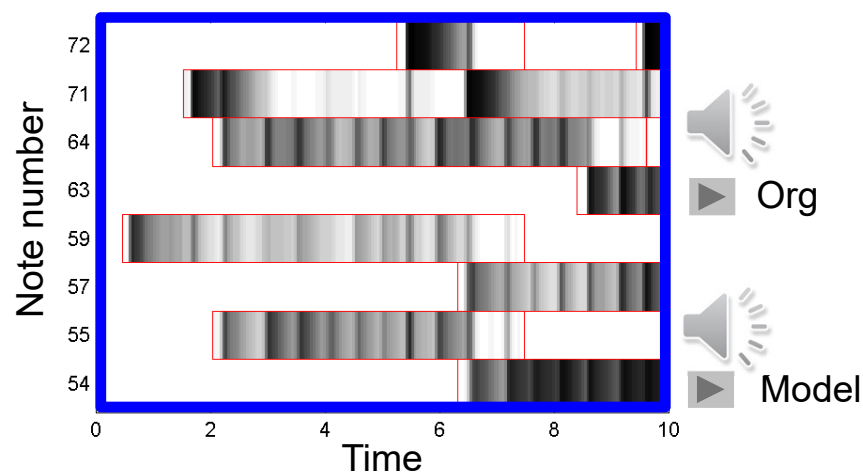
Initialized activations



Learnt templates

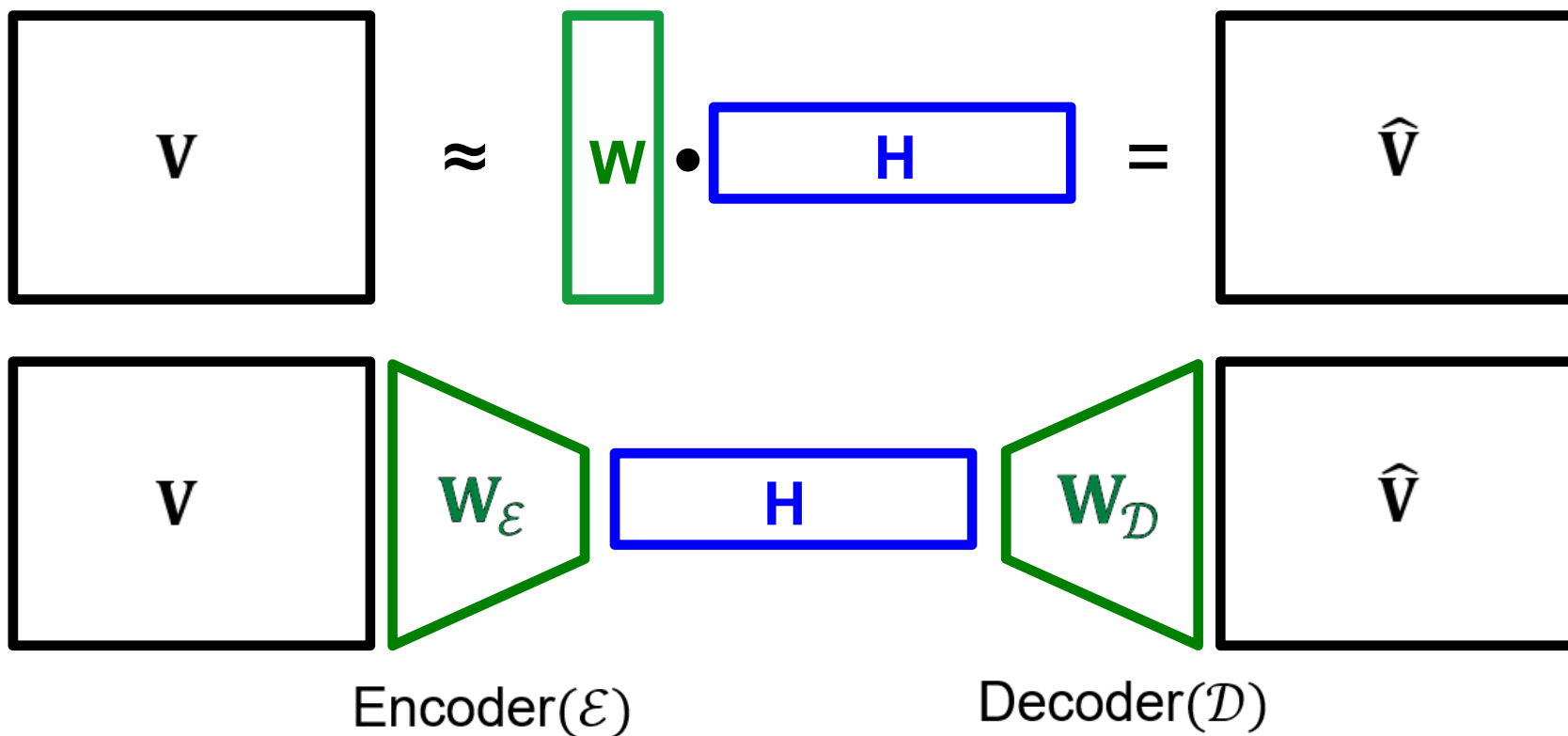


Learnt activations



Constrained initialization → NMF as refinement

NMF-Decomposition

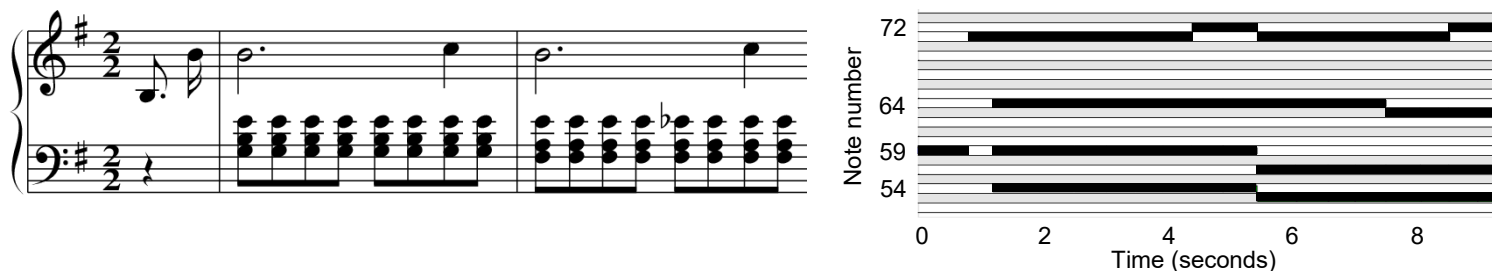


Smaragdis, Venkataramani: A Neural Network Alternative to Non-Negative Audio Models, ICASSP 2017.

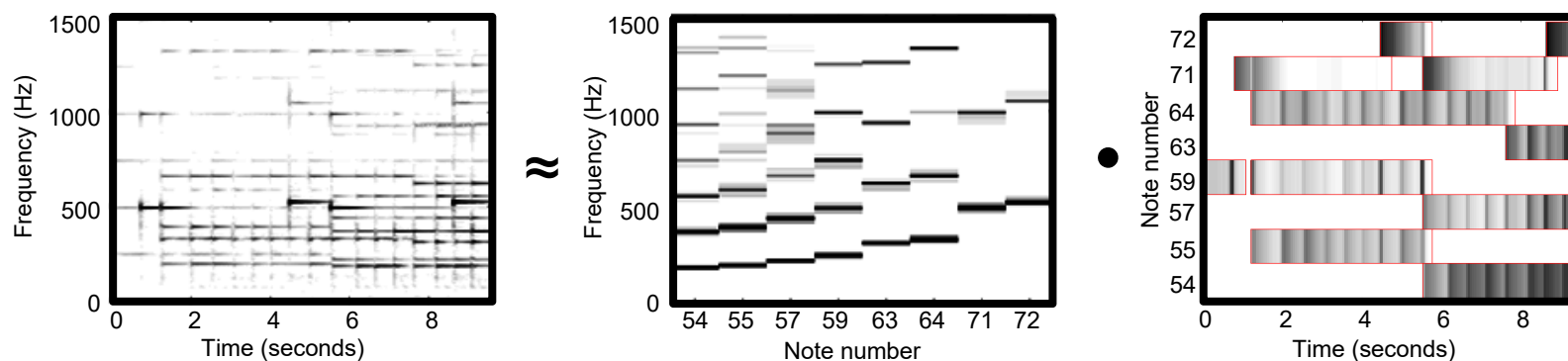
Lecture 6: Nonnegative Autoencoders with Applications to Music Audio Decomposing

Score-Informed Audio Decomposition

Exploit musical score to support decomposition process

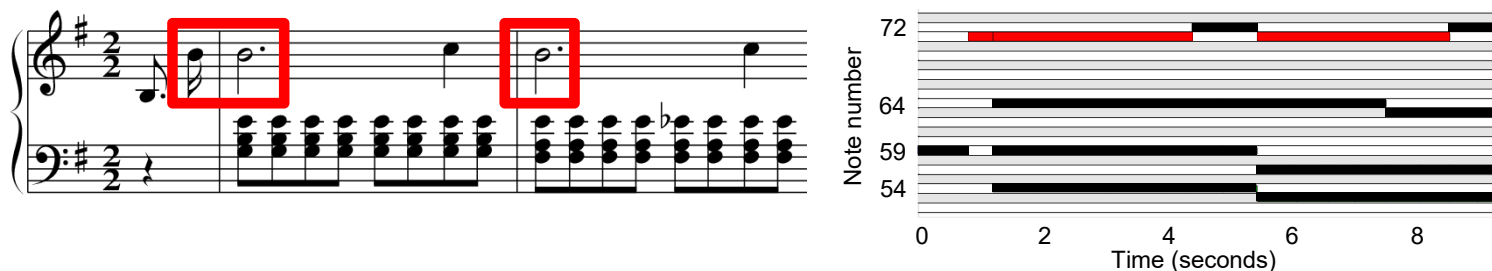


NMF-based spectrogram decomposition

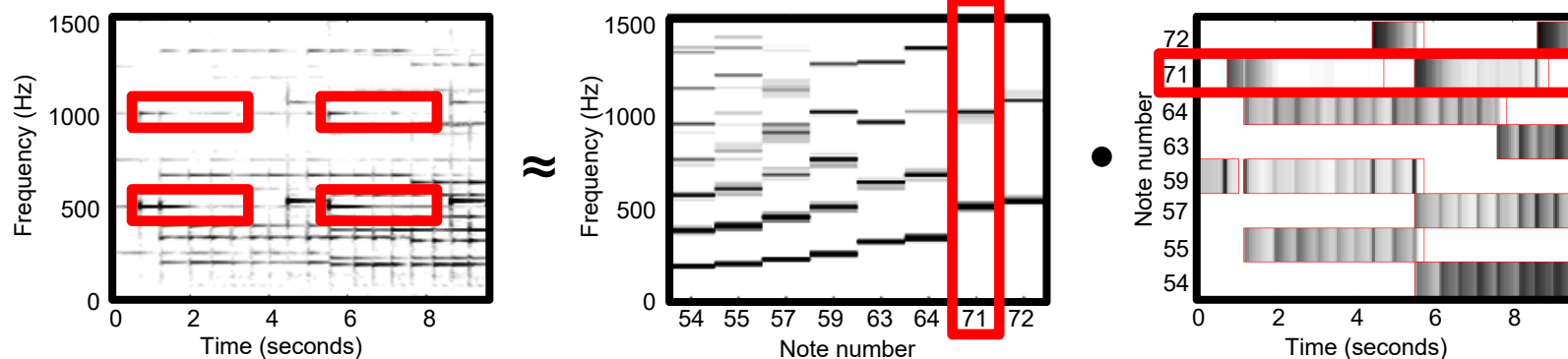


Score-Informed Audio Decomposition

Exploit musical score to support decomposition process

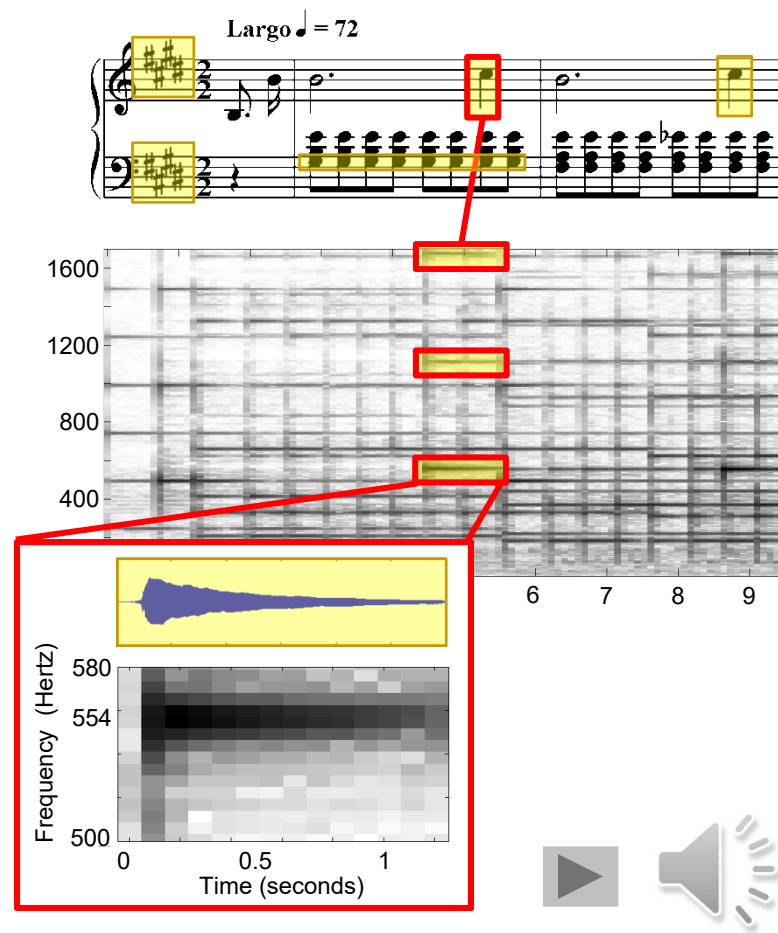
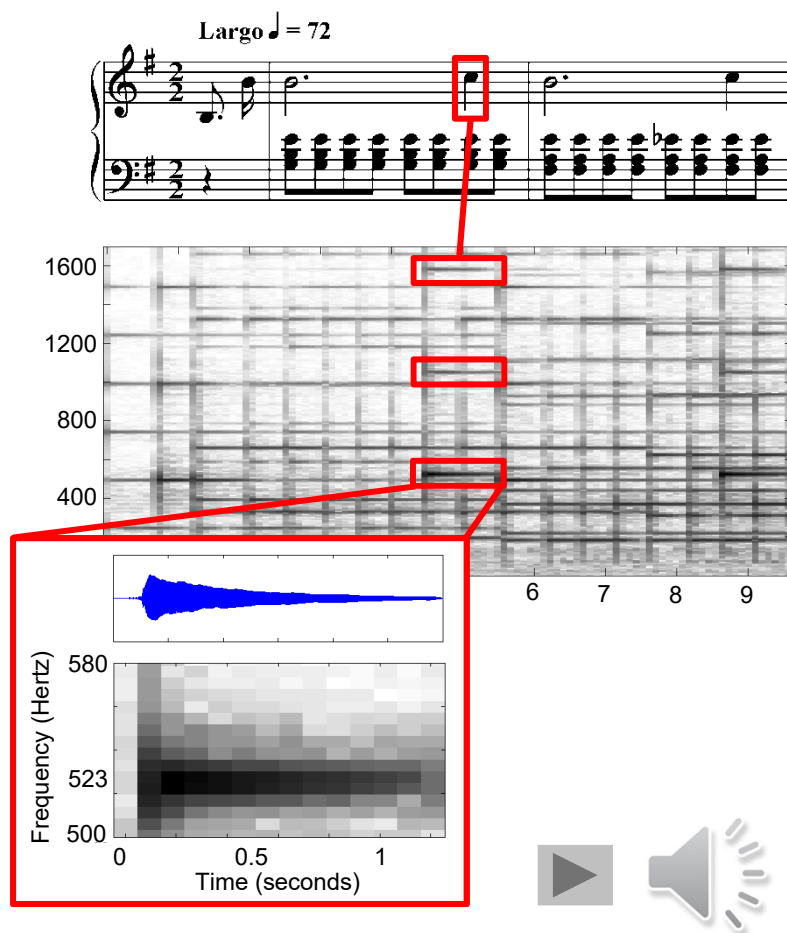


NMF-based spectrogram decomposition

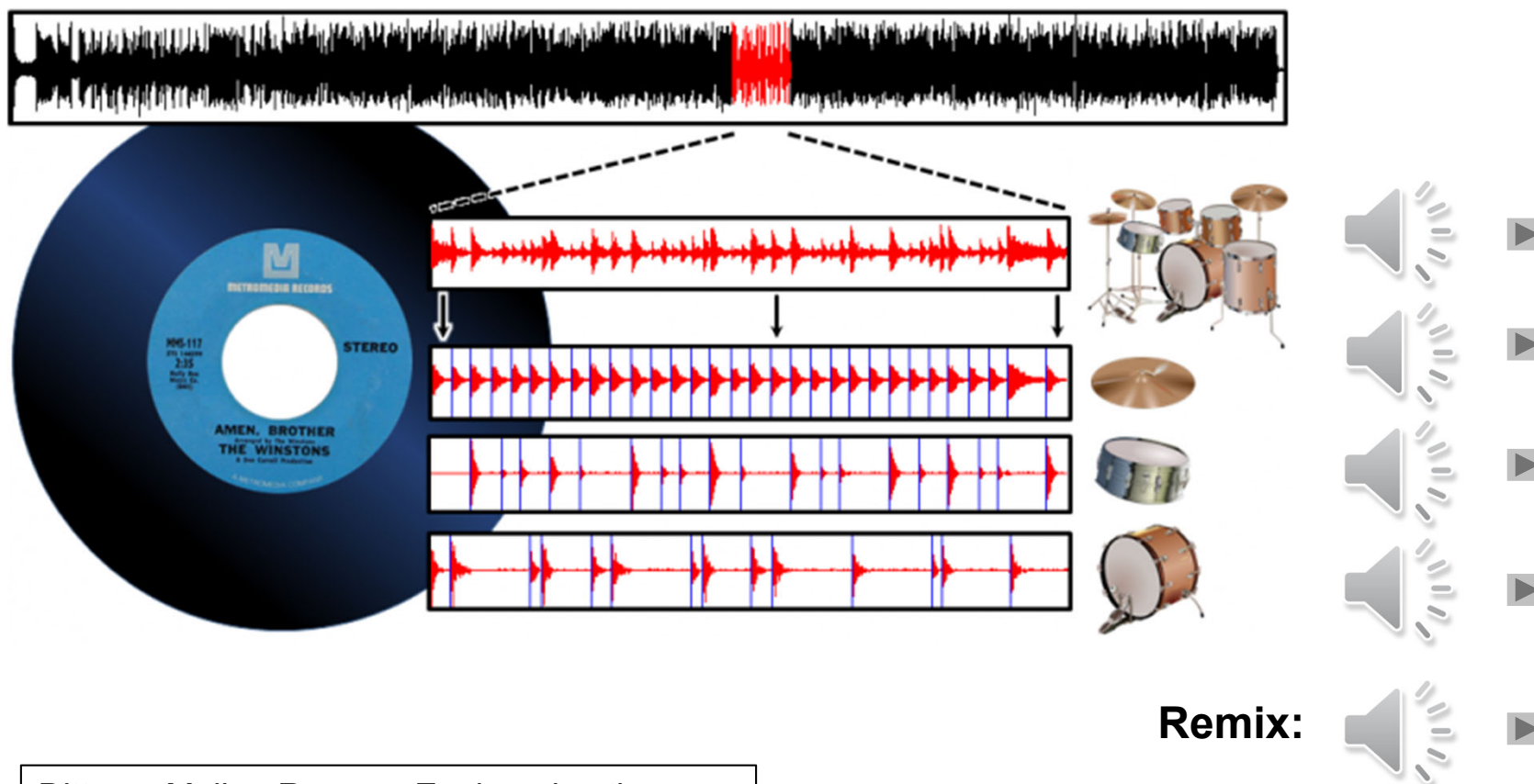


Score-Informed Audio Decomposition

Application: Audio editing



Informed Drum-Sound Decomposition



Dittmar, Müller: Reverse Engineering the Amen Break – Score-Informed Separation and Restoration Applied to Drum Recordings, IEEE/ACM TASLP, 2016.

Informed Drum-Sound Decomposition



Major challenge: Reconstructed sound events often have artifacts

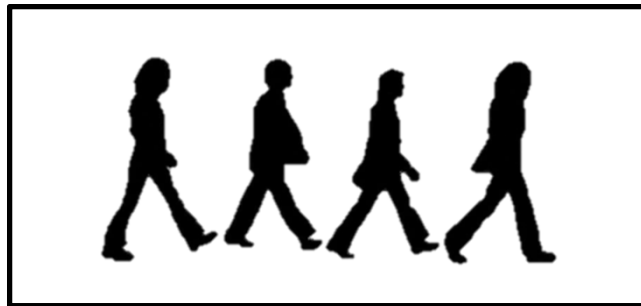
Approaches:

- Resynthesize certain sound components
- Differentiable Digital Signal Processing (DDSP) combines classical DSP and deep learning
- Generative adversarial networks may help to reduce the artifacts

Lecture 8: Recurrent and Generative Adversarial Network Architectures for Text-to-Speech

Audio Mosaicing

Target signal: Beatles–Let it be



Source signal: Bees



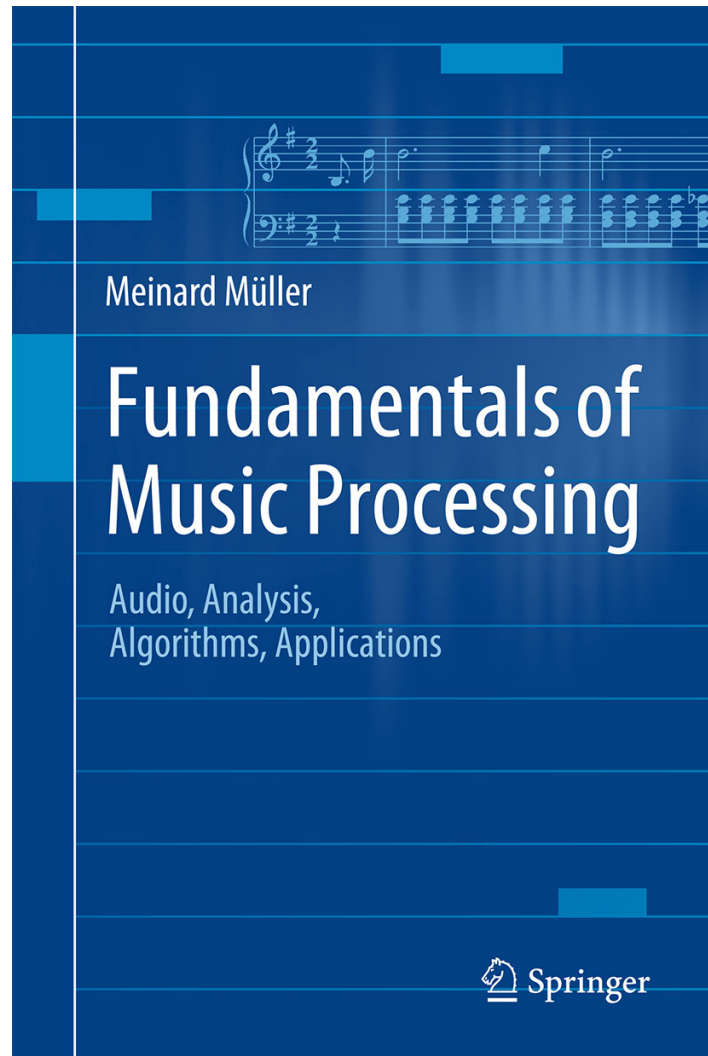
Mosaic signal: **Let it Bee**

Driedger, Prätzlich, Müller: Let It Bee – Towards NMF-Inspired Audio Mosaicing, ISMIR 2015..

Selected Topics in Deep Learning for Audio, Speech, and Music Processing

1. Introduction to Audio and Speech Processing
2. Introduction to Music Processing
3. Permutation Invariant Training Techniques for Speech Separation
4. Deep Clustering for Single-Channel Ego-Noise Suppression
5. Music Source Separation
6. Nonnegative Autoencoders with Applications to Music Audio Decomposing
7. Attention in Sound Source Localization and Speaker Extraction
8. Recurrent and Generative Adversarial Network Architectures for Text-to-Speech
9. Connectionist Temporal Classification (CTC) Loss with Applications to Theme-Based Music Retrieval
10. From Theory to Practise

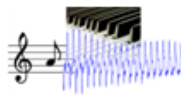

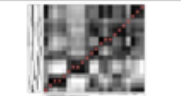


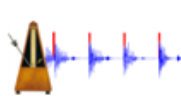
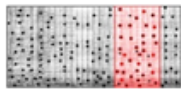
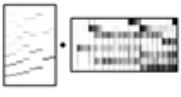
Book: Fundamentals of Music Processing



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

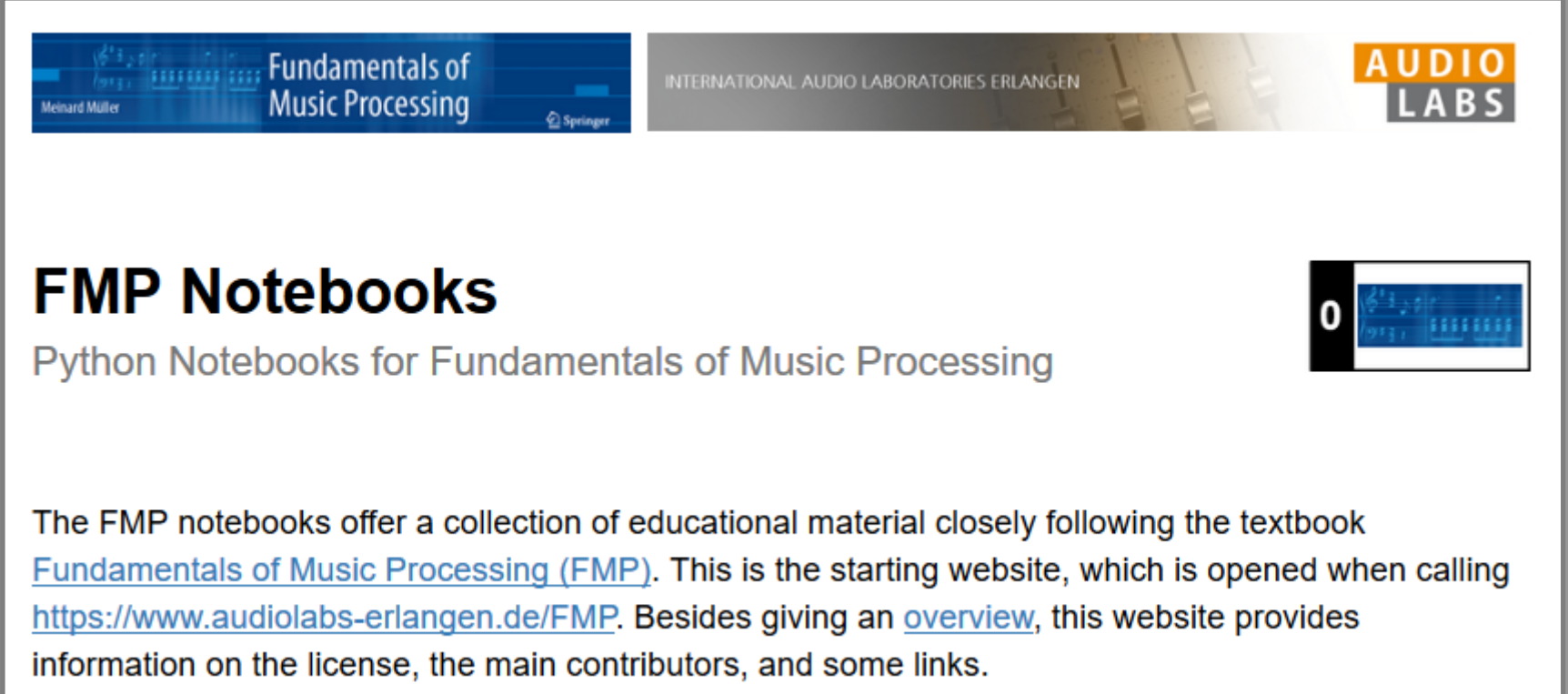
Book: Fundamentals of Music Processing

Chapter		Music Processing Scenario
1		Music Representations
2		Fourier Analysis of Signals
3		Music Synchronization
4		Music Structure Analysis
5		Chord Recognition
6		Tempo and Beat Tracking
7		Content-Based Audio Retrieval
8		Musically Informed Audio Decomposition

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

Software & Audio: FMP Notebooks



The screenshot shows the header of the FMP Notebooks website. On the left is the cover of the book 'Fundamentals of Music Processing' by Meinard Müller, published by Springer. In the center is the text 'INTERNATIONAL AUDIO LABORATORIES ERLANGEN'. On the right is the 'AUDIO LABS' logo. Below the header, the main heading is 'FMP Notebooks' in a large, bold, black font. Underneath it is the subtitle 'Python Notebooks for Fundamentals of Music Processing'. To the right of the subtitle is a small thumbnail image of a notebook page with a blue header and musical notation. Below the subtitle is a paragraph of text: 'The FMP notebooks offer a collection of educational material closely following the textbook [Fundamentals of Music Processing \(FMP\)](#). This is the starting website, which is opened when calling <https://www.audiolabs-erlangen.de/FMP>. Besides giving an [overview](#), this website provides information on the license, the main contributors, and some links.'

<https://www.audiolabs-erlangen.de/FMP>