

Lecture  
**Music Processing**  
**Introduction**

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

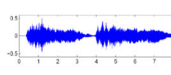


**Music Information Retrieval (MIR)**

Sheet Music (Image)



CD / MP3 (Audio)



MusicXML (Text)

```
<note  
  pitch="440" />  
<note  
  pitch="460" />  
<note  
  pitch="480" />  
<note  
  pitch="500" />  
<note  
  pitch="520" />  
<note  
  pitch="540" />  
<note  
  pitch="560" />  
<note  
  pitch="580" />  
<note  
  pitch="600" />  
</note>
```

Dance / Motion (Mocap)



**Music**

MIDI



Singing / Voice (Audio)



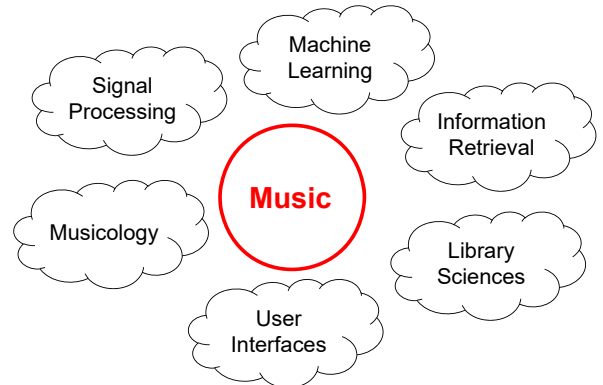
Music Film (Video)



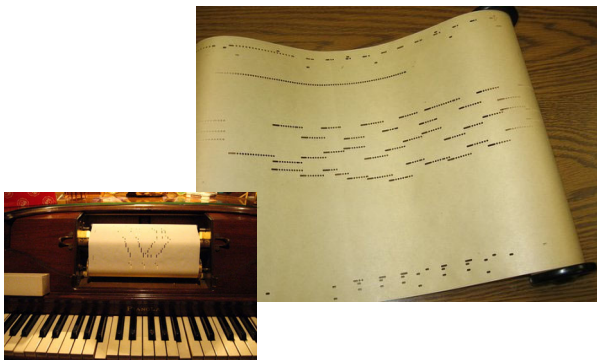
Music Literature (Text)



**Music Information Retrieval (MIR)**

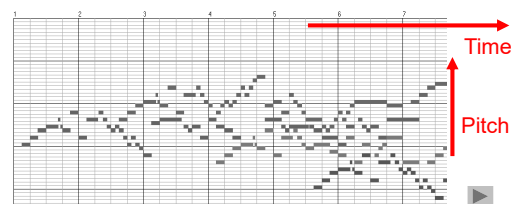
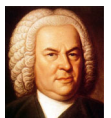


**Piano Roll Representation (1900)**



**Piano Roll Representation**

J.S. Bach, C-Major Fuge  
(Well Tempered Piano, BWV 846)

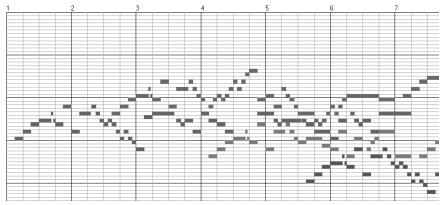


## Piano Roll Representation

Query: 



Goal: Find all occurrences of the query



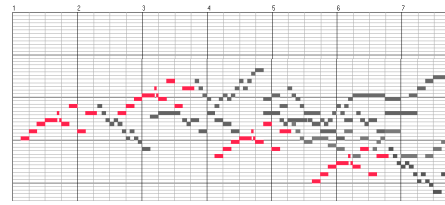
## Piano Roll Representation

Query: 



Goal: Find all occurrences of the query

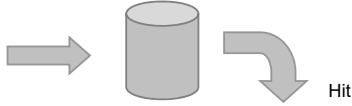
Matches:



## Music Retrieval



Database



Audio ID

Bernstein (1962)  
Beethoven, Symphony No. 5

Version ID

Beethoven, Symphony No. 5:  
▪ Bernstein (1962)  
▪ Karajan (1982)  
▪ Gould (1992)

Category ID

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

## Music Synchronization: Audio-Audio

Beethoven's Fifth

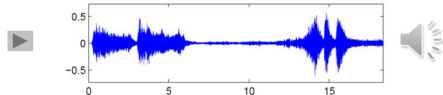


## Music Synchronization: Audio-Audio

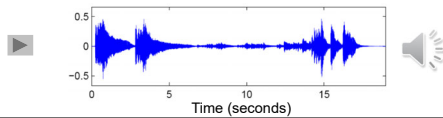
Beethoven's Fifth



Orchester  
(Karajan)



Piano  
(Scherbakov)



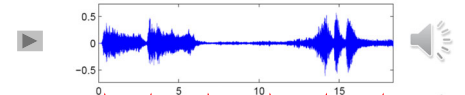
Time (seconds)

## Music Synchronization: Audio-Audio

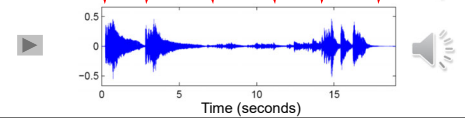
Beethoven's Fifth



Orchester  
(Karajan)



Piano  
(Scherbakov)

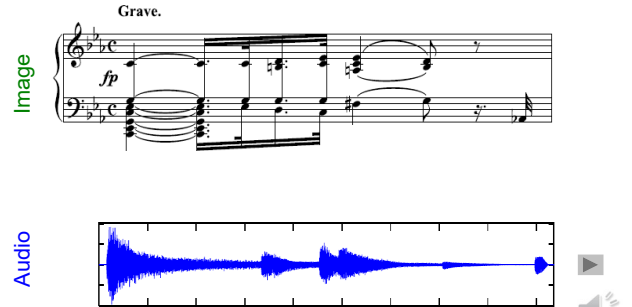


Time (seconds)

## Application: Interpretation Switcher

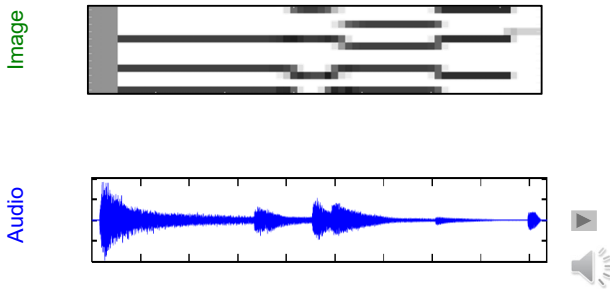


## Music Synchronization: Image-Audio



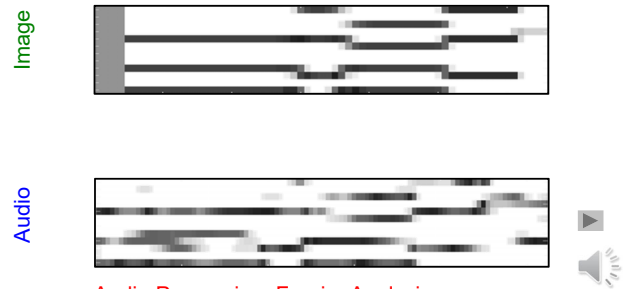
## Music Synchronization: Image-Audio

### Image Processing: Optical Music Recognition



## Music Synchronization: Image-Audio

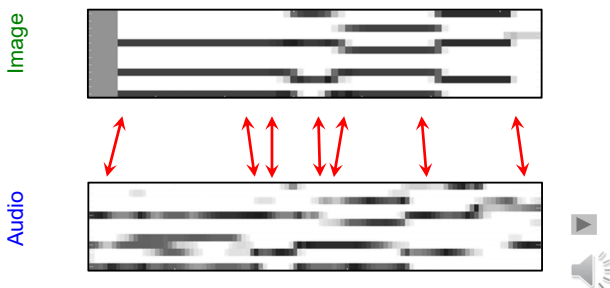
### Image Processing: Optical Music Recognition



### Audio Processing: Fourier Analysis

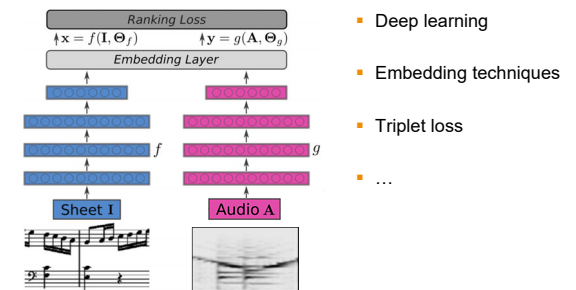
## Music Synchronization: Image-Audio

### Image Processing: Optical Music Recognition

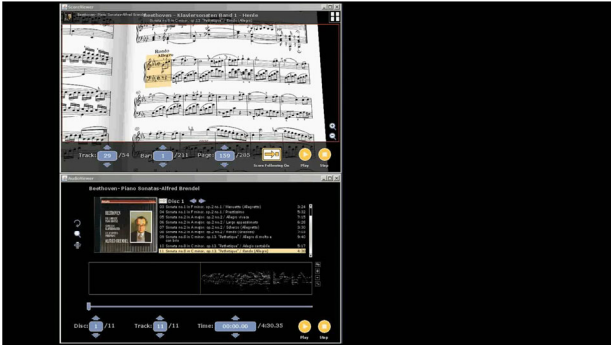


### Audio Processing: Fourier Analysis

## Music Synchronization: Image-Audio

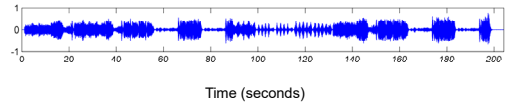


## Application: Score Viewer



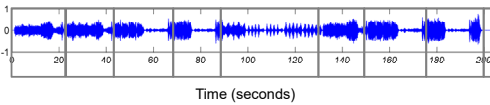
## Music Structure Analysis

Example: Brahms Hungarian Dance No. 5 (Ormandy)



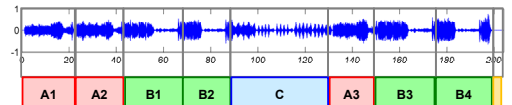
## Music Structure Analysis

Example: Brahms Hungarian Dance No. 5 (Ormandy)



## Music Structure Analysis

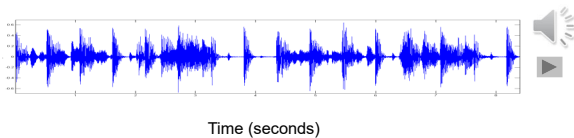
Example: Brahms Hungarian Dance No. 5 (Ormandy)



## Tempo Estimation and Beat Tracking

Basic task: "Tapping the foot when listening to music"

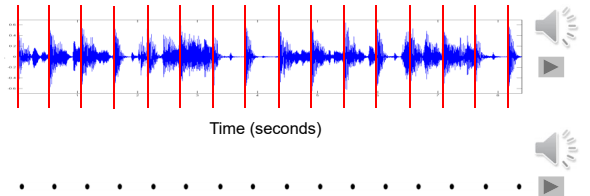
Example: Queen – Another One Bites The Dust



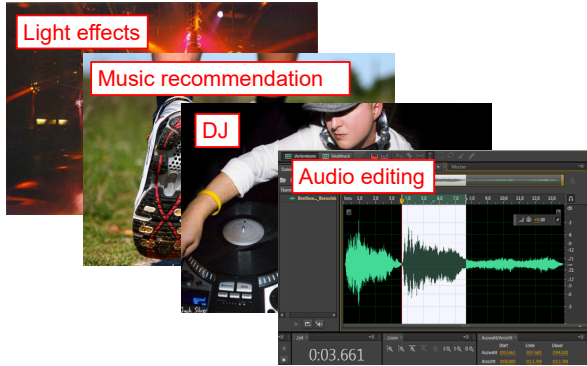
## Tempo Estimation and Beat Tracking

Basic task: "Tapping the foot when listening to music"



Example: Queen – Another One Bites The Dust



## Tempo Estimation and Beat Tracking



## Why is Music Processing Challenging?

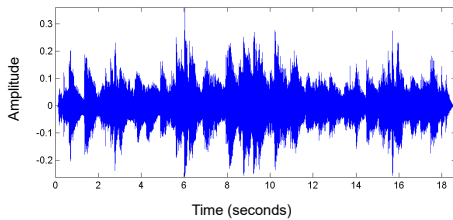
Example: Chopin, Mazurka Op. 63 No. 3  



## Why is Music Processing Challenging?

Example: Chopin, Mazurka Op. 63 No. 3

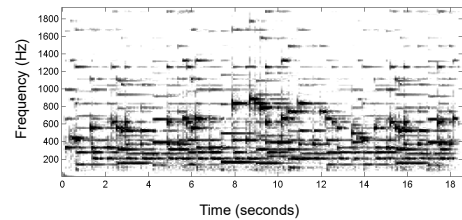
- Waveform





## 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

- 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"

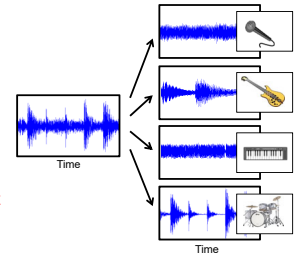


## Source Separation

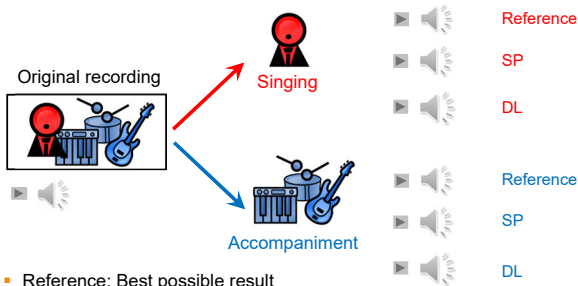
- Decomposition of audio stream into different sound sources
- Central task in digital signal processing
- "Cocktail party effect"
- Several input signals
- Sources are assumed to be statistically independent

## Source Separation (Music)

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



## Source Separation (Music)



- Reference: Best possible result
- SP: Using traditional signal processing
- DL: Using data-driven approach based on deep learning

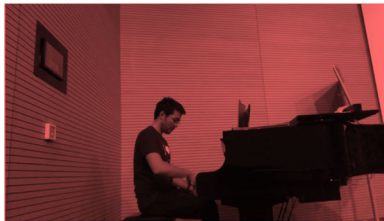
## Source Separation (Music)

- Yigitcan Özer
- PhD student in engineering
- Pianist



## Source Separation (Music)

- Yigitcan Özer
- PhD student in engineering
- Pianist



Only Piano!



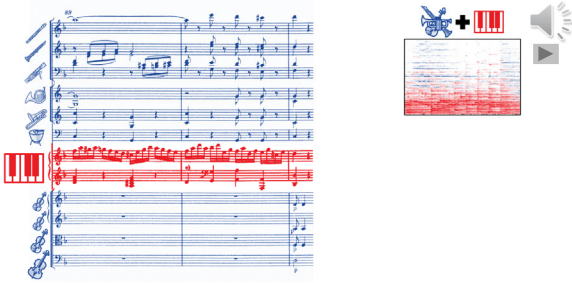
Where is the orchestra?



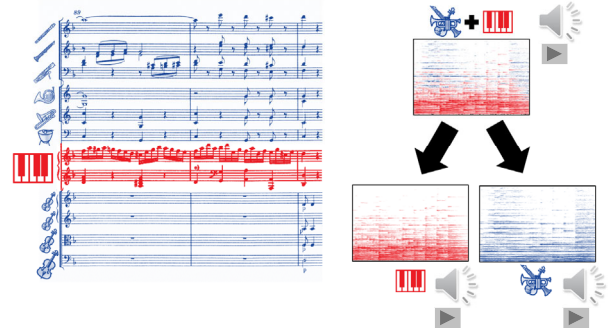
## Source Separation (Music)



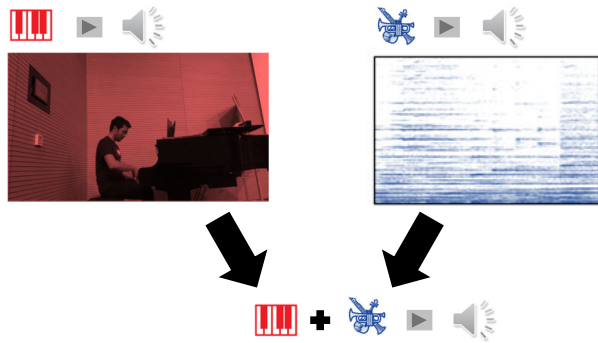
## Source Separation (Music)



## Source Separation (Music)



## Source Separation (Music)



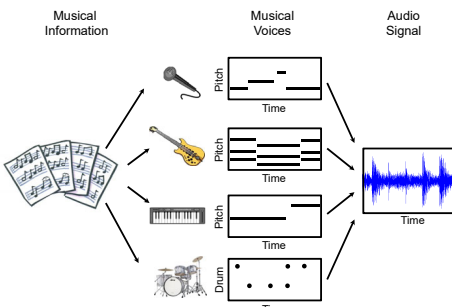
## Score-Informed Source Separation

Exploit musical score to support decomposition process



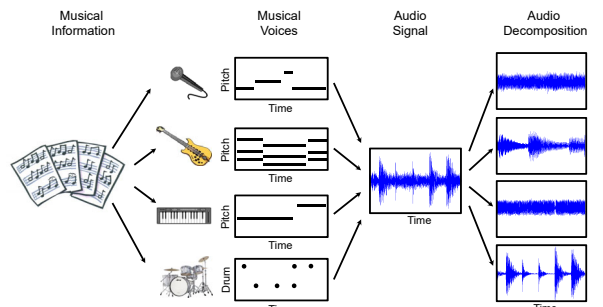
## Score-Informed Source Separation

Exploit musical score to support decomposition process

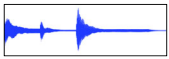


## Score-Informed Source Separation

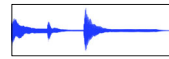
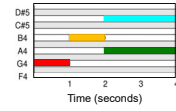
Exploit musical score to support decomposition process



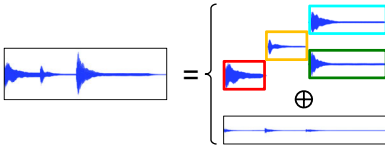
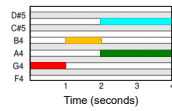
## Score-Informed Audio Decomposition



## Score-Informed Audio Decomposition



## Score-Informed Audio Decomposition

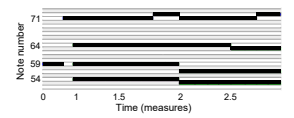


## Score-Informed Audio Decomposition

Sheet music



Piano roll



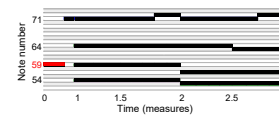
## Score-Informed Audio Decomposition

Sheet music



$p = 59$

Piano roll



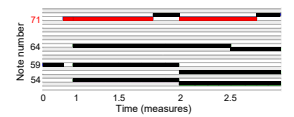
## Score-Informed Audio Decomposition

Sheet music



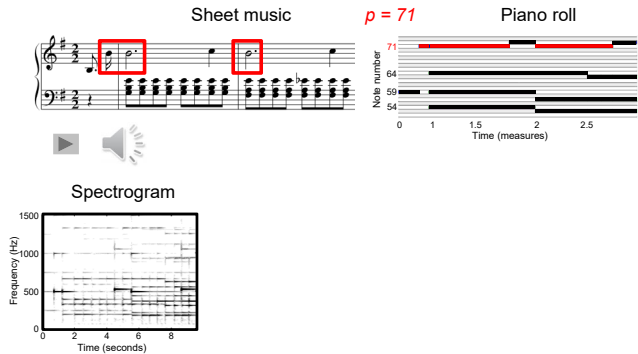
$p = 71$

Piano roll

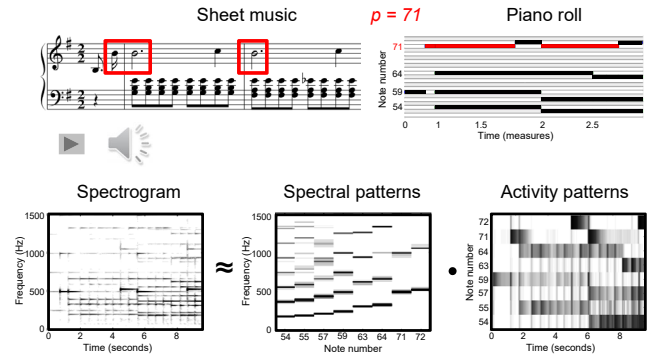




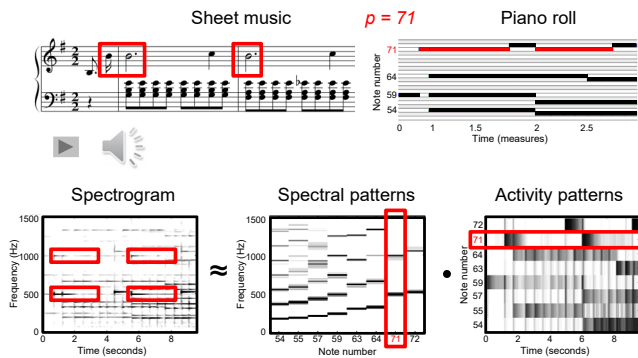
## Score-Informed Audio Decomposition



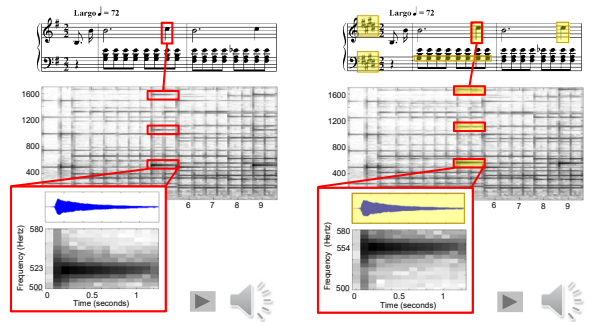
## Score-Informed Audio Decomposition



## Score-Informed Audio Decomposition

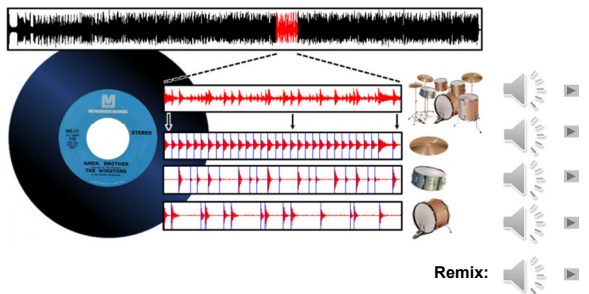


## Score-Informed Audio Decomposition



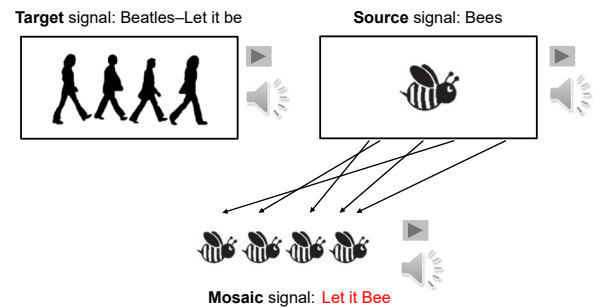
## Score-Informed Audio Decomposition

### Informed Drum-Sound Decomposition



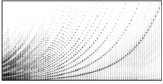
## Score-Informed Audio Decomposition

### Audio mosaicing (style transfer)

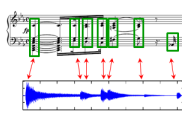


## Music Processing

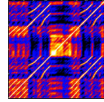
### Fourier Transform Audio Features



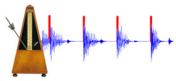
### Music Synchronization



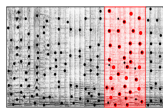
### Structure Analysis



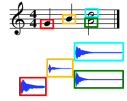
### Tempo and Beat Tracking



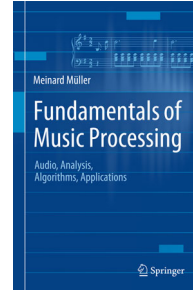
### Audio Identification



### Audio Decomposition



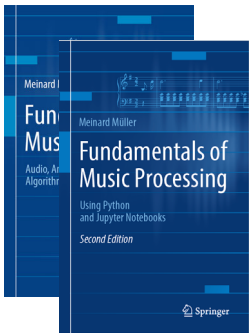
## Fundamentals of Music Processing (FMP)



Meinard Müller  
Fundamentals of Music Processing  
Audio, Analysis, Algorithms, Applications  
Springer, 2015

Accompanying website:  
[www.music-processing.de](http://www.music-processing.de)

## Fundamentals of Music Processing (FMP)



Meinard Müller  
Fundamentals of Music Processing  
Audio, Analysis, Algorithms, Applications  
Springer, 2015

Accompanying website:  
[www.music-processing.de](http://www.music-processing.de)

2nd edition  
Meinard Müller  
Fundamentals of Music Processing  
Using Python and Jupyter Notebooks  
Springer, 2021

## Fundamentals of Music Processing (FMP)

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  
Springer, 2015

Accompanying website:  
[www.music-processing.de](http://www.music-processing.de)

2nd edition  
Meinard Müller  
Fundamentals of Music Processing  
Using Python and Jupyter Notebooks  
Springer, 2021

## FMP Notebooks: Education & Research

**FMP Notebooks**  
Python Notebooks for Fundamentals of Music Processing

The FMP notebooks offer a collection of educational material closely following the textbook [Fundamentals of Music Processing \(FMP\)](https://www.audiolabs-erlangen.de/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>

## References (FMP Notebooks)

- Meinard Müller: Fundamentals of Music Processing – Using Python and Jupyter Notebooks. 2nd Edition, Springer, 2021.  
<https://www.springer.com/gp/book/9783030698072>
- Meinard Müller and Frank Zalkow: libfmp: A Python Package for Fundamentals of Music Processing. Journal of Open Source Software (JOSS), 6(63): 1–5, 2021.  
<https://joss.theoj.org/papers/10.21105/joss.03326>
- Meinard Müller: An Educational Guide Through the FMP Notebooks for Teaching and Learning Fundamentals of Music Processing. Signals, 2(2): 245–285, 2021.  
<https://www.mdpi.com/2624-6120/2/2/18>
- Meinard Müller and Frank Zalkow: FMP Notebooks: Educational Material for Teaching and Learning Fundamentals of Music Processing. Proc. International Society for Music Information Retrieval Conference (ISMIR): 573–580, 2019.  
<https://zenodo.org/record/3527872#.Y0hEQqzaUk>
- Meinard Müller, Brian McFee, and Katherine Kinnaird: Interactive Learning of Signal Processing Through Music: Making Fourier Analysis Concrete for Students. IEEE Signal Processing Magazine, 38(3): 73–84, 2021.  
<https://ieeexplore.ieee.org/document/9418542>

---

## Resources (Group Meinard Müller)

- FMP Notebooks:

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

- libfmp:

<https://github.com/meinardmueller/libfmp>

- synctoolbox:

<https://github.com/meinardmueller/synctoolbox>

- libtsm:

<https://github.com/meinardmueller/libtsm>

- Preparation Course Python (PCP) Notebooks:

<https://www.audiolabs-erlangen.de/resources/MIR/PCP/PCP.html>

<https://github.com/meinardmueller/PCP>