Lecture

Information Retrieval for Music and Motion

Meinard Müller

Summer Term 2008

Audio Matching

UNIVERSITÄT DES SAARLANDES

M2CI CLUSTER OF EXCELLENCE

mpi max planck institut informatik

Audio Matching

Given: Large music database containing several
- recordings of the same piece of music
- interpretations by various musicians
- arrangements in different instrumentations

Goal: Given a short query audio clip, identify all corresponding audio clips of similar musical content
- irrespective of the specific interpretation and instrumentation
- automatically and efficiently

Query-by-Example paradigm

Audio Matching

- Müller/Kurth/Clausen (ISMIR 2005)
- Kurth/Müller (IEEE T-ASLP 2008)

Related problems

Audio identification

- Allamanche et al. (AES 2001)
- Cano et al. (*IEEE MMSP* 2002)
- Kurth/Clausen/Ribbrock (AES 2002)
- Wang (ISMIR 2003)
- Shrestha/Kalker (ISMIR 2004)

Audio synchronization

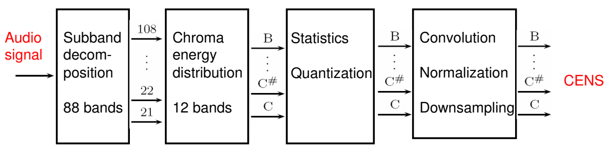
Audio structure analysis

Audio Matching

General strategy

- Normalized and smoothed chroma features
  - correlates to harmonic progression
  - robust to variations in dynamics, timbre, articulation, local tempo
- Robust matching procedure
  - efficient
  - robust to global tempo variations
  - scalable using index structure

Feature Design



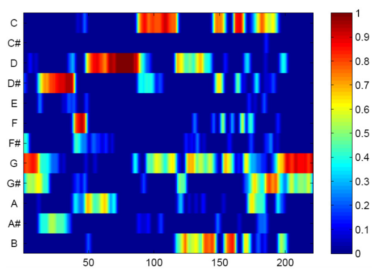
Two stages:

Stage 1: Local chroma energy distribution features
Stage 2: Normalized short-time statistics

$\rightsquigarrow$ CENS = Chroma Energy Normalized Statistics

Feature Design

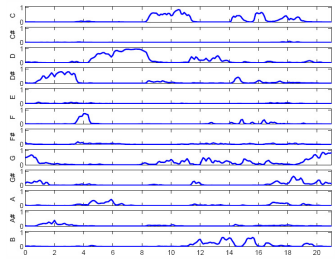
Beethoven's Fifth: Bernstein



Resolution: 10 features/second
Feature window size: 200 milliseconds

Feature Design

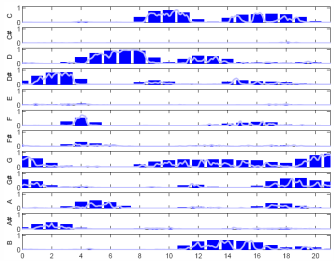
Beethoven's Fifth: Bernstein



Resolution: 10 features/second
Feature window size: 200 milliseconds

Feature Design

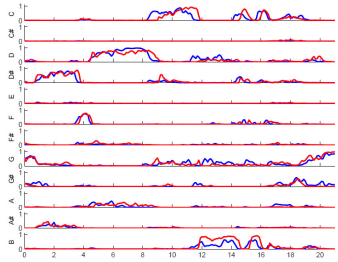
Beethoven's Fifth: Bernstein



Resolution: 1 features/second
Feature window size: 4000 milliseconds

Feature Design

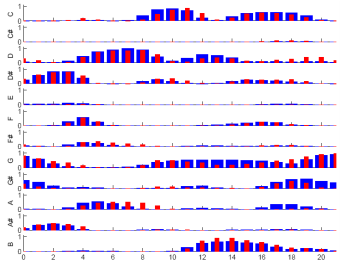
Beethoven's Fifth: Bernstein vs. Sawallisch



Resolution: 10 features/second
Feature window size: 200 milliseconds

Feature Design

Beethoven's Fifth: Bernstein vs. Sawallisch

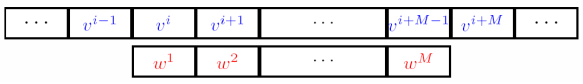


Resolution: 1 features/second
Feature window size: 4000 milliseconds

Matching Procedure

Compute CENS feature sequences

- Database $D \rightsquigarrow F[D] = (v^1, v^2, \ldots, v^N)$
- Query $Q \rightsquigarrow F[Q] = (w^1, w^2, \ldots, w^M)$
- $N \approx 500000$, $M \approx 20$

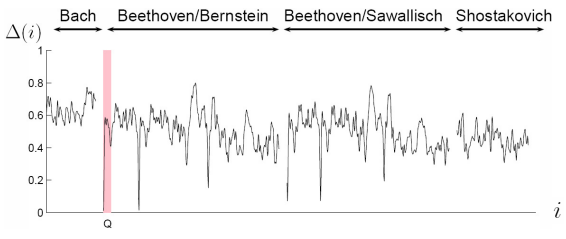


$\Delta(i) :=$ local distance$((v^i, v^{i+1} \ldots, v^{i+M-1}),(w^1, w^2, \ldots, w^M))$

$\rightsquigarrow$ Global distance function $\Delta : [1:N] \rightarrow [0,1]$

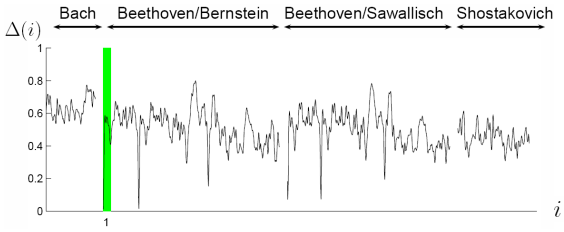
Matching Procedure

Query: Beethoven's Fifth / Bernstein, first 20 seconds



Matching Procedure

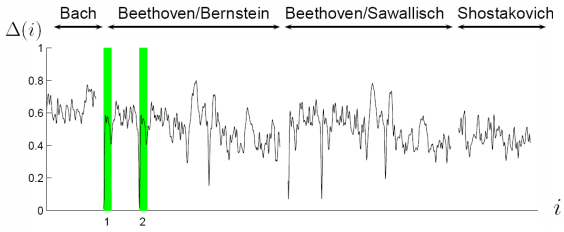
Query: Beethoven‘s Fifth / Bernstein, first 20 seconds



Best audio matches: 1

Matching Procedure

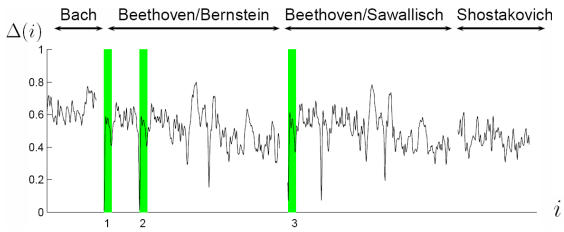
Query: Beethoven‘s Fifth / Bernstein, first 20 seconds



Best audio matches: 2

Matching Procedure

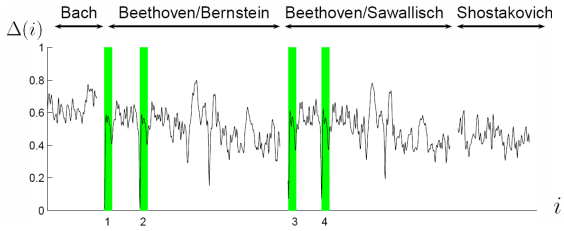
Query: Beethoven‘s Fifth / Bernstein, first 20 seconds



Best audio matches: 3

Matching Procedure

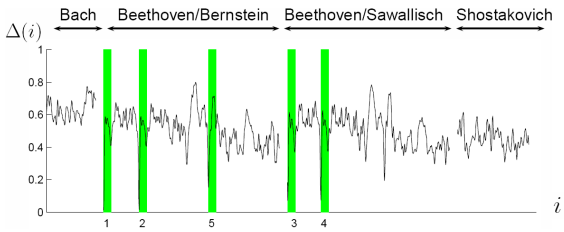
Query: Beethoven‘s Fifth / Bernstein, first 20 seconds



Best audio matches: 4

Matching Procedure

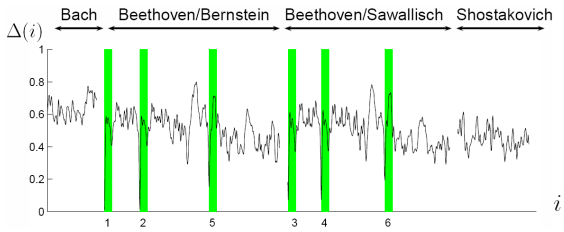
Query: Beethoven‘s Fifth / Bernstein, first 20 seconds



Best audio matches: 5

Matching Procedure

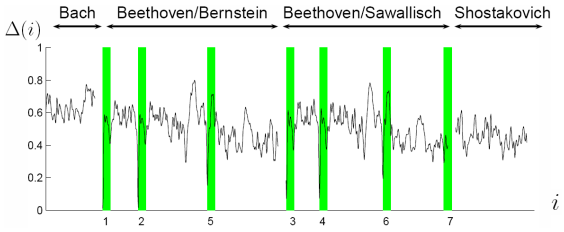
Query: Beethoven‘s Fifth / Bernstein, first 20 seconds



Best audio matches: 6

Matching Procedure

Query: Beethoven's Fifth / Bernstein, first 20 seconds



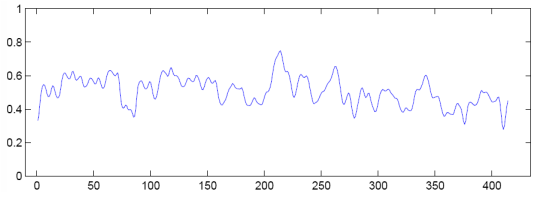
Best audio matches: 7

Global Tempo Variations

Query: Beethoven's Fifth / Bernstein, first 20 seconds

Problem: Karajan is much faster $\rightsquigarrow$ useless Δ

Solution?

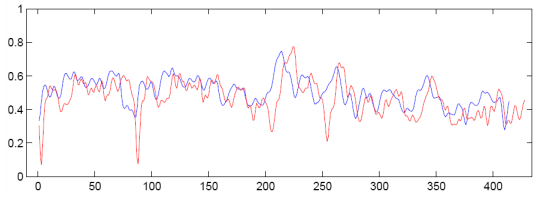


Global Tempo Variations

Query: Beethoven's Fifth / Bernstein, first 20 seconds

Problem: Karajan is much faster $\rightsquigarrow$ useless Δ

Solution: Make Bernstein query faster and comute new Δ

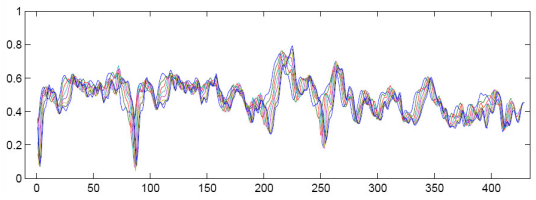


Global Tempo Variations

Query: Beethoven's Fifth / Bernstein, first 20 seconds

Problem: Karajan is much faster $\rightsquigarrow$ useless Δ

Solution: Compute Δ for various tempi

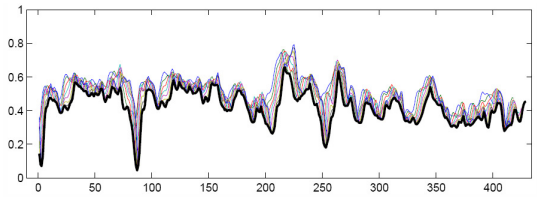


Global Tempo Variations

Query: Beethoven's Fifth / Bernstein, first 20 seconds

Problem: Karajan is much faster $\rightsquigarrow$ useless Δ

Solution: Minimize over all resulting Δ's $\rightsquigarrow$ $\Delta^{\min}$

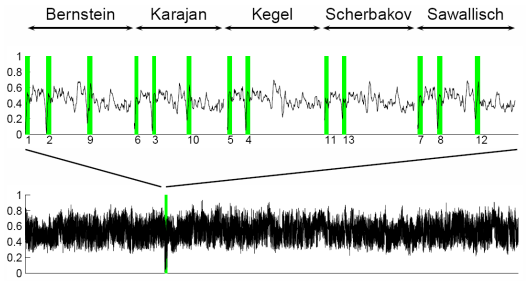


Experiments

- Audio database > 110 hours, 16.5 GB
- Preprocessing $\rightsquigarrow$ CENS features, 40.3 MB
- Query clip $\approx$ 20 seconds
- Query response time < 10 seconds

Experiments

Query: Beethoven‘s Fifth / Bernstein, first 20 seconds



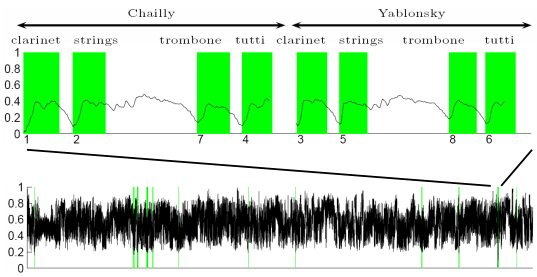
Experiments

Query: Beethoven‘s Fifth / Bernstein, first 20 seconds

Rank	$\Delta^{\min}$	Piece	Position
1	0.0114	Beethoven’s Fifth/Bernstein	0 - 21
2	0.0150	Beethoven’s Fifth/Bernstein	101 - 122
3	0.0438	Beethoven’s Fifth/Karajan	86 - 103
⋮	⋮	⋮	⋮
10	0.1796	Beethoven’s Fifth/Karajan	252 - 271
11	0.1827	Beethoven (Liszt) Fifth/Scherbakov	0 - 19
12	0.1945	Beethoven’s Fifth/Sawallisch	275 - 296
13	0.1970	Beethoven’s Fifth (Liszt)/Scherbakov	86 - 103
14	0.2169	Schumann op 97,1/Levine	28 - 43
⋮	⋮	⋮	⋮

Experiments

Query: Shostakovich, Waltz/Chailly, first 27 seconds



Experiments

Query: Shostakovich, Waltz/Chailly, first 21 seconds

Rank	$\Delta^{\min}$	Piece	Position
1	0.0172	Shostakovich/Chailly	0 - 21
2	0.0505	Shostakovich/Chailly	41 - 60
3	0.0983	Shostakovich/Chailly	180 - 198
4	0.1044	Shostakovich/Yablonsky	1 - 19
5	0.1090	Shostakovich/Yablonsky	36 - 52
6	0.1401	Shostakovich/Yablonsky	156 - 174
7	0.1476	Shostakovich/Chailly	144 - 162
8	0.1626	Bach BWV 582/Chorzempa	358 - 373
9	0.1668	Beethoven op 37,1/Toscanini	12 - 28
10	0.1729	Beethoven op 37,1/Pollini	202 - 218
⋮	⋮	⋮	⋮

Conclusions

Strategy: Absorb variations at feature level

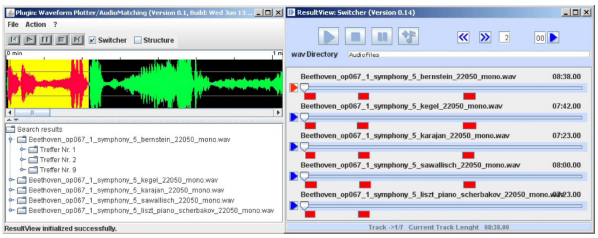
- Chroma $\rightsquigarrow$ invariance to timbre
- Normalization $\rightsquigarrow$ invariance to dynamics
- Smoothing $\rightsquigarrow$ invariance to local time deviations

Conclusions

Global Matching Procedure

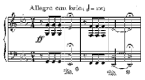
- Strategy: Exact matching and multiple scaled queries
  - simulate tempo variations by feature resampling
  - different queries correspond to different tempi
  - indexing possible
- Strategy: Dynamic Time Warping
  - subsequence variant
  - more flexible (in particular for longer queries)
  - indexing hard

System: SyncPlayer/AudioMatching



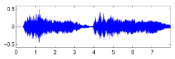
Multimodal Computing and Interaction

Sheet Music (Image)



MIDI

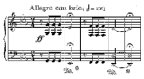
CD / MP3 (Audio)



Music

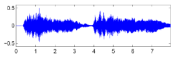
Multimodal Computing and Interaction

Sheet Music (Image)



MIDI

CD / MP3 (Audio)



MusicXML (Text)

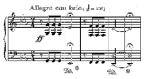
Music

Singing / Voice (Audio)

Music Literature (Text)

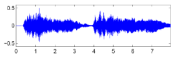
Multimodal Computing and Interaction

Sheet Music (Image)



MIDI

CD / MP3 (Audio)



MusicXML (Text)

Music

Singing / Voice (Audio)

Music Literature (Text)

Music Film (Video)

Dance / Motion (Mocap)