INTERNATIONAL AUDIO LABORATORIES ERLANGEN
A joint institution of Fraunhofer IIS and Universität Erlangen-Nürnberg

Tutorial T3, EUROGRAPHICS
Saarbrücken, May 8, 2023

# Learning with Music Signals: Technology Meets Education

## Music Retrieval

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

Friedrich-Alexander-Universität Erlangen-Nürnberg
Fraunhofer IIS

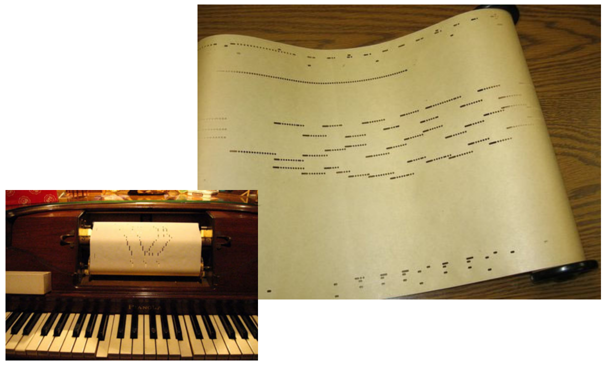
## Music Representations




## Music Representations

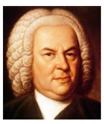



## Piano Roll Representation (1900)



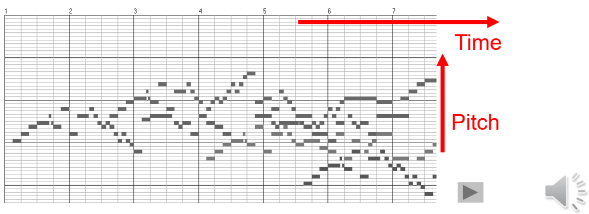


## Piano Roll Representation



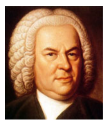
J.S. Bach, C-Major Fuge

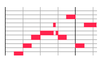
(Well Tempered Piano, BWV 846)



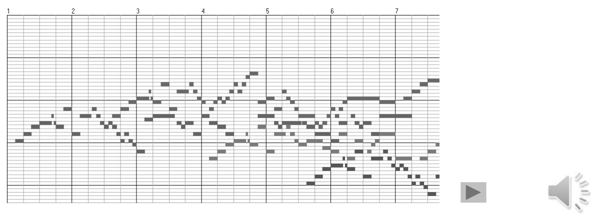



## Piano Roll Representation

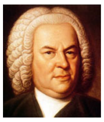


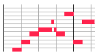
Query: 

Goal: Find all occurrences of the query



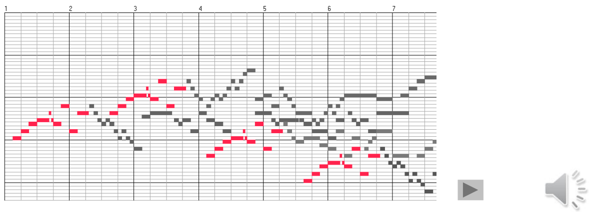
## Piano Roll Representation



Query: 

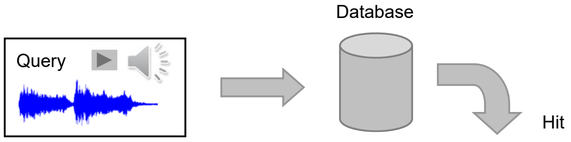
Goal: Find all occurrences of the query

Matches:





## Music Retrieval




Audio ID

Bernstein (1962)
Beethoven, Symphony No. 5

Version ID

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

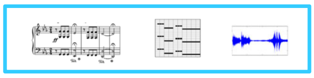
Category ID

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



## Music Retrieval

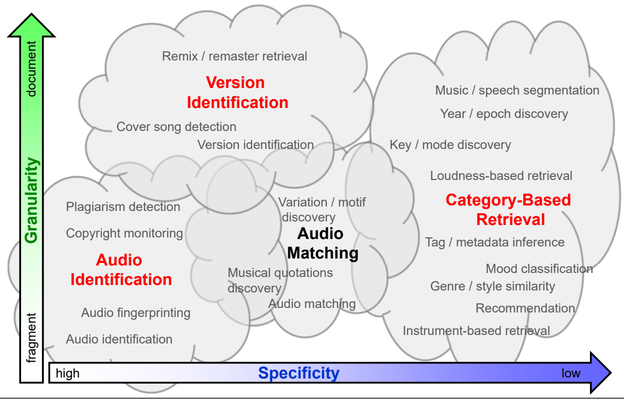
**Modalities**



| **Retrieval tasks:** | **Specificity** | **Granularity** |
|---|---|---|
| Audio ID | High specificity | Fragment-based retrieval |
| Version ID | ↕ | ↕ |
| Category ID | Low specificity | Document-based retrieval |



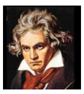
## Music Retrieval






## Music Synchronization: Audio-Audio

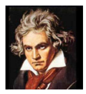
Beethoven's Fifth

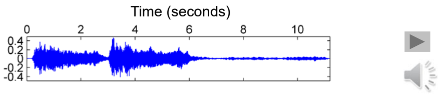


## Music Synchronization: Audio-Audio

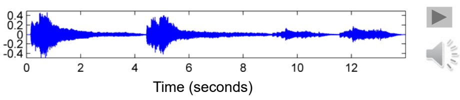
Beethoven's Fifth

Karajan (Orchester)

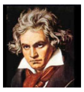



Gould (Piano)



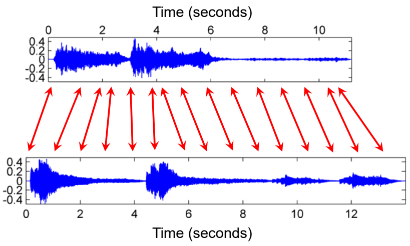
## Music Synchronization: Audio-Audio

Beethoven's Fifth

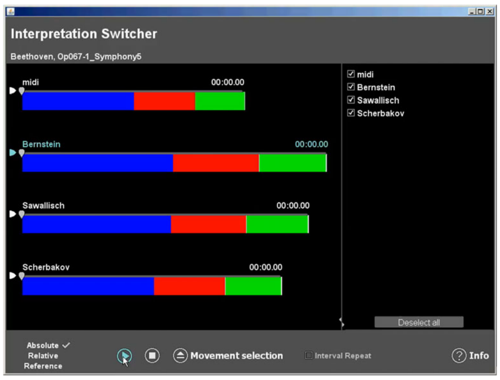
Karajan (Orchester)

Gould (Piano)






## Application: Interpretation Switcher






## 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)**

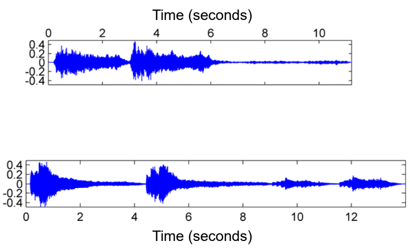


## Music Synchronization: Audio-Audio

Beethoven's Fifth

Karajan (Orchester)

Gould (Piano)



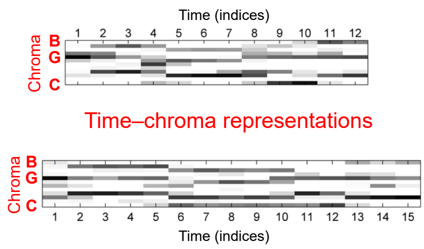



## Music Synchronization: Audio-Audio

Beethoven's Fifth

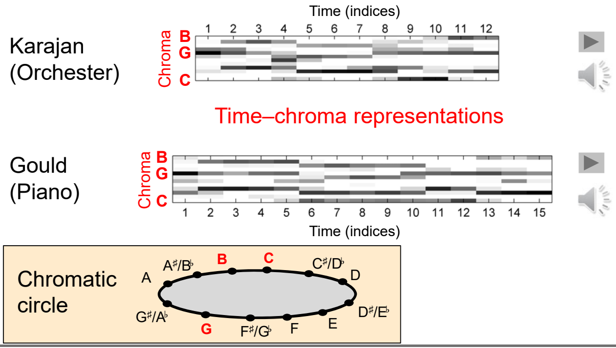
Karajan (Orchester)

Gould (Piano)



## Music Synchronization: Audio-Audio

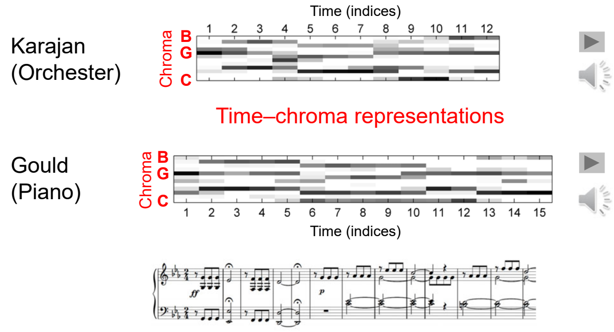
Beethoven's Fifth






## Music Synchronization: Audio-Audio

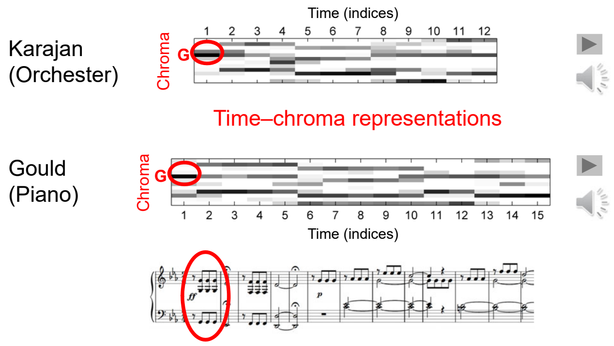
Beethoven's Fifth






## Music Synchronization: Audio-Audio

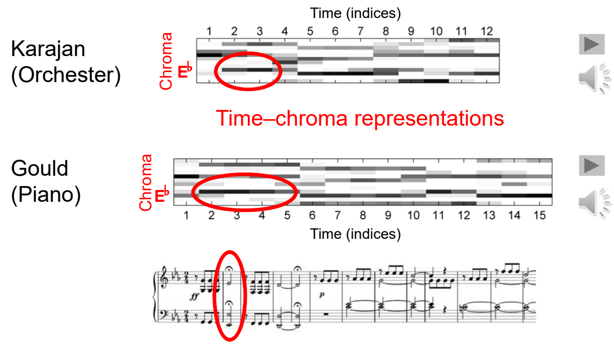
Beethoven's Fifth





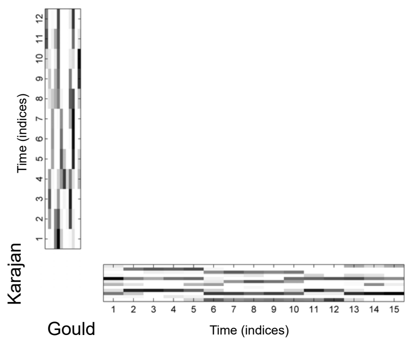

## Music Synchronization: Audio-Audio

Beethoven's Fifth





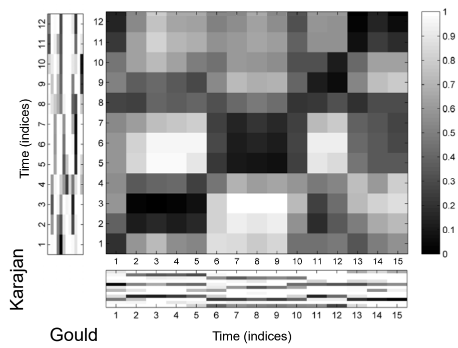

## Music Synchronization: Audio-Audio





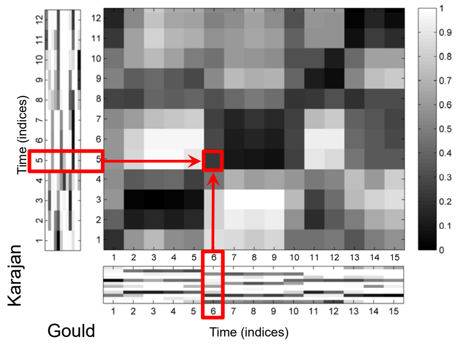

## Music Synchronization: Audio-Audio

Cost matrix



## Music Synchronization: Audio-Audio

### Cost matrix

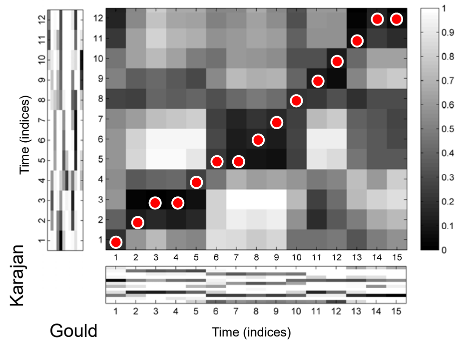






## Music Synchronization: Audio-Audio

### Cost-minimizing warping path

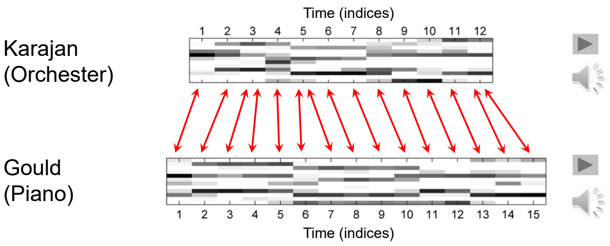






## Music Synchronization: Audio-Audio

### Cost-minimizing warping path = Optimal alignment







## Music Synchronization: Audio-Audio

### Deep Learning Approaches

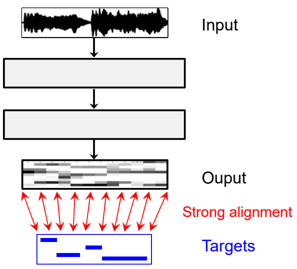
- Learn audio features from data
  - Should be robust to performance variations
  - Should yield high alignment accuracy
  - Should have musical relevance
- Alignment problem
  - Pre-aligned data for training
  - Part of loss function → differentiability?

**CTC-Loss**
Graves et al.: Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. ICML, 2006

**Soft-DTW**
Cuturi, Blondel: Soft-DTW: A Differentiable Loss Function for Time-Series. ICML, 2017



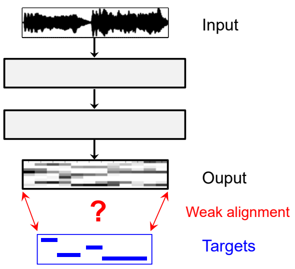

## Feature Learning




- Task: Learn audio features using a neural network
- Loss: Binary cross-entropy
  - framewise loss
  - requires strongly aligned targets
  - hard to obtain




## Feature Learning




- Task: Learn audio features using a neural network
- Loss: Binary cross-entropy
  - framewise loss
  - requires strongly aligned targets
  - hard to obtain
- Alignment as part of loss function
  - requires only weakly aligned targets
  - needs to be differentiable
- Problem: DTW is not differentiable
  → Soft DTW

## Dynamic Time Warping (DTW)

$X := (x_1, x_2, \ldots, x_N)$

$Y := (y_1, y_2, \ldots, y_M)$

$x_n, y_m \in \mathcal{F},\ n \in [1:N],\ m \in [1:M]$

$\mathcal{F}$ = Feature space

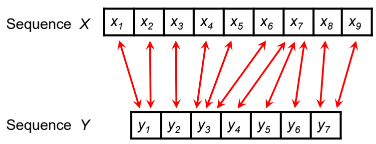
**Alignment matrix**

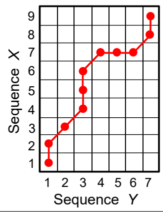
$A \in \{0,1\}^{N \times M}$

Set of all possible alignment matrices

$\mathcal{A}_{N,M} \subset \{0,1\}^{N \times M}$

Alignment






## Dynamic Time Warping (DTW)

$X := (x_1, x_2, \ldots, x_N)$

$Y := (y_1, y_2, \ldots, y_M)$

$x_n, y_m \in \mathcal{F},\ n \in [1:N],\ m \in [1:M]$

$\mathcal{F}$ = Feature space

**Alignment matrix**

$A \in \{0,1\}^{N \times M}$

Set of all possible alignment matrices

$\mathcal{A}_{N,M} \subset \{0,1\}^{N \times M}$

Cost measure: $c : \mathcal{F} \times \mathcal{F} \to \mathbb{R}_{\geq 0}$

Cost matrix: $C \in \mathbb{R}^{N \times M}$ with $C(n,m) := c(x_n, y_m)$

Cost of alignment: $\langle A, C \rangle$

DTW cost: $\mathrm{DTW}(C) = \min\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$

Optimal alignment: $A^* = \mathrm{argmin}\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$



## Dynamic Time Warping (DTW)

DTW cost: $\mathrm{DTW}(C) = \min\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$

- Efficient computation via Bellman's recursion in O(*NM*)

$$D(n,m) = \min\{D(n-1,m), D(n,m-1), D(n,m)\} + C(n,m)$$

for *n>1* and *m>1* and suitable initialization.

$$\mathrm{DTW}(C) = D(N,M)$$

- Problem: DTW(*C*) is not differentiable with regard to *C*
- Idea: Replace min-function by a smooth version

$$\min{}^{\gamma}(\mathcal{S}) = -\gamma \log \sum_{s \in \mathcal{S}} \exp(-s/\gamma)$$

for set $\mathcal{S} \subset \mathbb{R}$ and temperature parameter $\gamma \in \mathbb{R}$



## Soft Dynamic Time Warping (SDTW)

SDTW cost: $\mathrm{SDTW}^{\gamma}(C) = \min{}^{\gamma}\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$

- Efficient computation via Bellman's recursion in O(*NM*) still works:

$$D^{\gamma}(n,m) = \min{}^{\gamma}\{D^{\gamma}(n-1,m), D^{\gamma}(n,m-1), D^{\gamma}(n,m)\} + C(n,m)$$

for *n>1* and *m>1* and suitable initialization.

$$\mathrm{SDTW}^{\gamma}(C) = D^{\gamma}(N,M)$$

- Limit case: $\mathrm{SDTW}^{\gamma}(C) \xrightarrow{\gamma \to 0} \mathrm{DTW}(C)$
- SDTW(*C*) is differentiable with regard to *C*
- Questions:
  - How does the gradient look like?
  - Can it be computed efficiently?
  - How does SDTW generalize the alignment concept?



## Soft Dynamic Time Warping (SDTW)

SDTW cost: $\mathrm{SDTW}^{\gamma}(C) = \min{}^{\gamma}\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$

- Define $p^{\gamma}(C)$ as the following "probability" distribution over $\mathcal{A}_{N,M}$:

$$p^{\gamma}(C)_A = \frac{\exp(-\langle A, C\rangle/\gamma)}{\sum_{A' \in \mathcal{A}_{N,M}} \exp(-\langle A', C\rangle/\gamma)} \quad \text{for } A \in \mathcal{A}_{N,M}$$

- The expected alignment with respect to $p^{\gamma}(C)$ is given by:

$$E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$$

- The gradient is given by:

$$\nabla_C \mathrm{SDTW}^{\gamma}(C) = E^{\gamma}(C)$$

- The gradient can be computed efficiently in O(*NM*) via a recursive algorithm.

**Soft-DTW**
Cuturi, Blondel: Soft-DTW: A Differentiable Loss Function for Time-Series. ICML, 2017

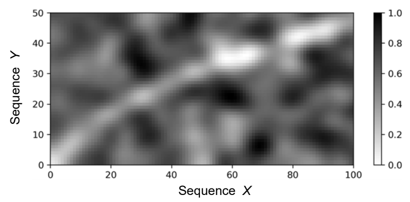


## Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter $\gamma$

Cost matrix C

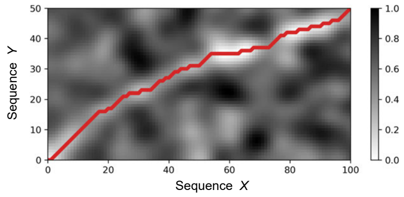


## Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^\gamma(C) = \sum_{A \in \mathcal{A}_{N,M}} p^\gamma(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter $\gamma$

Cost matrix C   Optimal alignment $A^*$



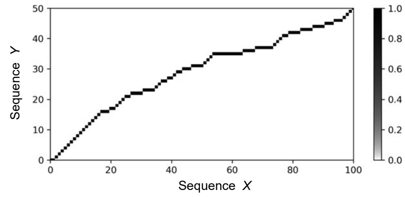



## Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^\gamma(C) = \sum_{A \in \mathcal{A}_{N,M}} p^\gamma(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter $\gamma$

$E^\gamma(C)$ with $\gamma = 0$  $( = A^*)$



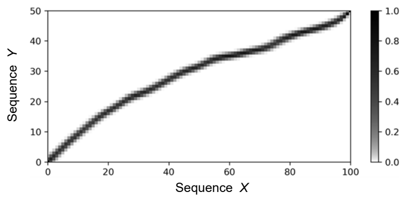



## Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^\gamma(C) = \sum_{A \in \mathcal{A}_{N,M}} p^\gamma(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter $\gamma$

$E^\gamma(C)$ with $\gamma = 0.1$



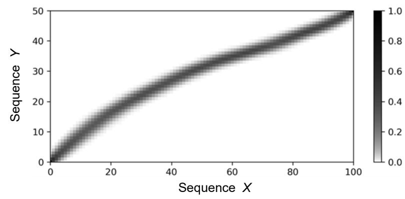



## Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^\gamma(C) = \sum_{A \in \mathcal{A}_{N,M}} p^\gamma(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter $\gamma$

$E^\gamma(C)$ with $\gamma = 1$



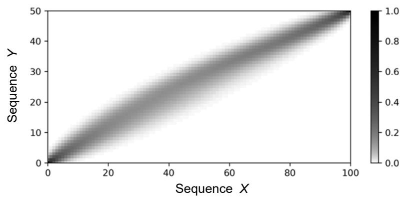



## Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^\gamma(C) = \sum_{A \in \mathcal{A}_{N,M}} p^\gamma(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter $\gamma$

$E^\gamma(C)$ with $\gamma = 10$



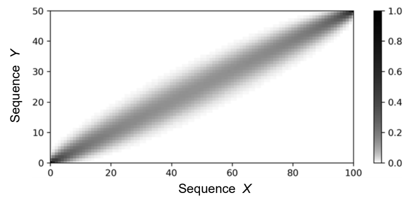



## Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^\gamma(C) = \sum_{A \in \mathcal{A}_{N,M}} p^\gamma(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter $\gamma$

$E^\gamma(C)$ with $\gamma = 100$



# Soft Dynamic Time Warping (SDTW)

## Conclusions

- Direct generalization of DTW (replacing min by smooth variant)
- Gradient is given by expected alignment
- Fast forward algorithm: O($NM$)
- Fast gradient computation: O($NM$)
- SDTW yields a (typically) poor lower bound for DTW
- Can be used as loss function to learn from weakly aligned sequences



# Soft Dynamic Time Warping (SDTW)

## References

- Marco Cuturi, Mathieu Blondel: Soft-DTW: A Differentiable Loss Function for Time-Series. ICML, pages 894–903, 2017.
- Mathieu Blondel, Arthur Mensch, Jean-Philippe Vert: Differentiable Divergences Between Time Series. AISTATS, pages 3853 – 3861, 2021.
- Michael Krause, Christof Weiß, Meinard Müller: Soft Dynamic Time Warping for Multi-Pitch Estimation and Beyond. IEEE ICASSP, 2023.

Thanks:
Michale Krause (Ph.D. 2023)
Johannes Zeitler (Ph.D.)

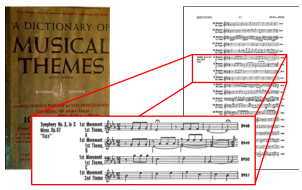


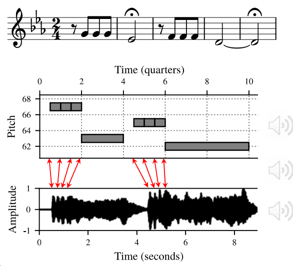
# Theme-Based Audio Retrieval



# Theme-Based Audio Retrieval

## Barlow & Morgenstern (1949): A Dictionary of Musical Themes



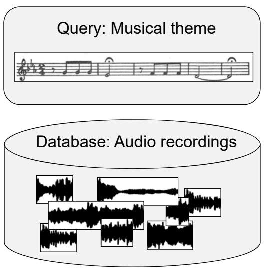




- 2067 themes by 54 different composers
- Recordings (1126 recordings, ~ 120 hours)
- Theme occurences (~ 5 hours)



# Theme-Based Audio Retrieval

## Barlow & Morgenstern (1949): A Dictionary of Musical Themes



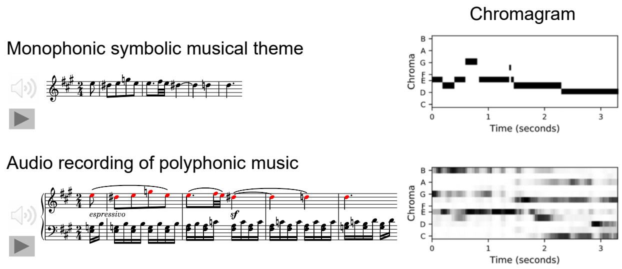

### Challenges

- **Cross-modality**
  Symbolic vs. audio data
- **Tuning**
  Deviations from standard tuning
- **Transposition**
  Played key vs. written key
- **Tempo**
  Local & global tempo deviations
- **Polyphony**
  Monophonic query vs. polyphonic audio



# Theme-Based Audio Retrieval

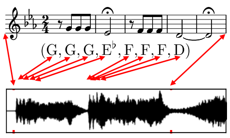
## Monophony–Polyphony Challenge




Goal: Compute "enhanced" chromagram from polyphonic audio recording that better matches the symbolic monophonic theme

## Theme-Based Audio Retrieval

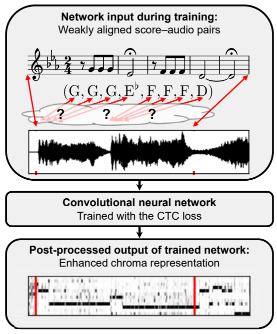
### Strongly Aligned Training Data





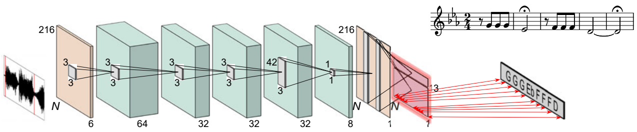

## Theme-Based Audio Retrieval

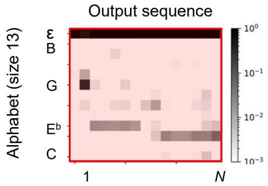
### Weakly Aligned Training Data






## Theme-Based Audio Retrieval

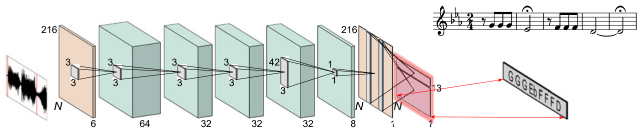



**Salience Computation**
Bittner, McFee, Salamon, Li, Bello: Deep salience representations for F0 tracking in polyphonic music. ISMIR, 2017.



## Theme-Based Audio Retrieval



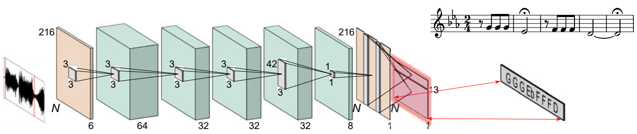

- Idea of CTC loss similar to SDTW
- Theme is given as label sequence over finite alphabet (size 13 including blank symbol)
- Expand label sequence to match audio feature sequence → valid alignment
- CTC loss considers probability over **all** valid alignments → differentiable

**CTC Loss**
Graves, Fernández, Gomez, Schmidhuber: Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. ICML, 2006.

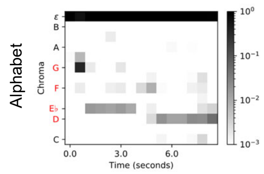


## Theme-Based Audio Retrieval

### CTC-Based Training



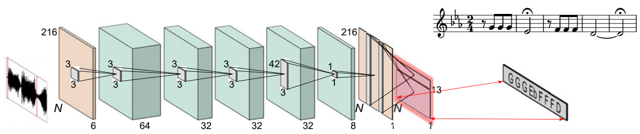

Label sequence $Y$
G G G E♭ F F F D



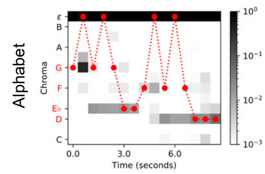



## Theme-Based Audio Retrieval

### CTC-Based Training



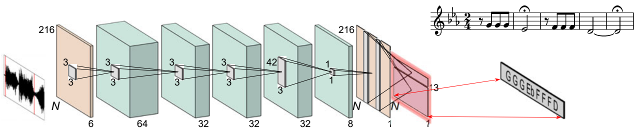

Label sequence $Y$
G G G E♭ F F F D




Valid alignment
G ε G ε G E♭ E♭ F ε F ε F D D D
→ matches sequence $X$

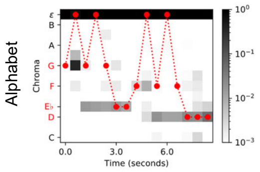
# Theme-Based Audio Retrieval

## CTC-Based Training



Label sequence $Y$
G G G E♭ F F F D

Output sequence $X$

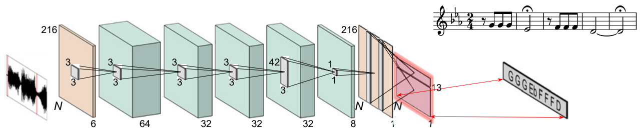



- Set of all valid alignments
  $$\mathbb{K}_{X,Y} = \{A \in (\mathbb{A}')^N : \kappa(A) = Y\}$$
- Probability of label sequence
  $$P(Y \mid X) = \sum_{A \in \mathbb{K}_{X,Y}} P(A \mid X)$$
- CTC loss
  $$L_\theta(X,Y) = -\log P(Y \mid X)$$



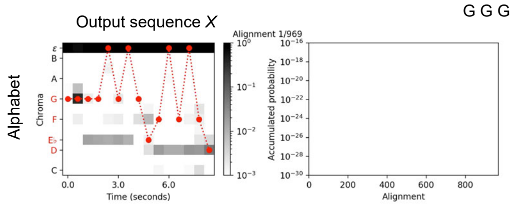
# Theme-Based Audio Retrieval

## CTC-Based Training



Label sequence $Y$
G G G E♭ F F F D

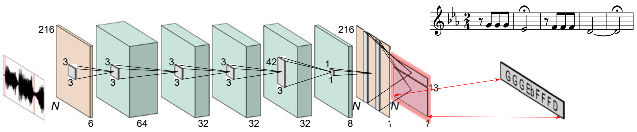
Output sequence $X$





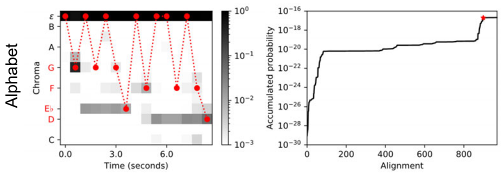

# Theme-Based Audio Retrieval

## CTC-Based Training



Label sequence $Y$
G G G E♭ F F F D

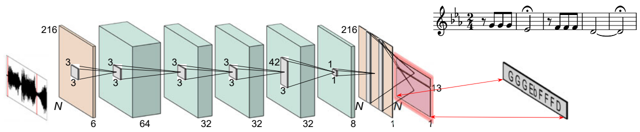
Output sequence $X$



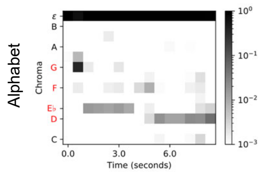



# Theme-Based Audio Retrieval

## CTC-Based Training



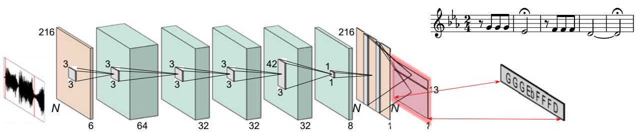
Output sequence $X$



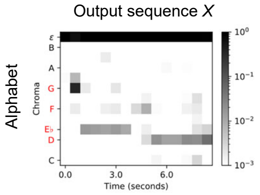



# Theme-Based Audio Retrieval

## CTC-Based Training

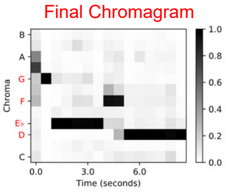


Output sequence $X$




Post processing

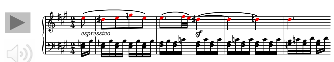
Final Chromagram





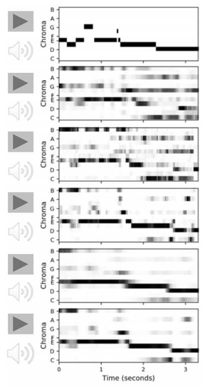

# Theme-Based Audio Retrieval

## Evaluation Results




(E, D♯, E, G, E, E, F♯, E, D♯, D, D)

| Chroma Variant | Top-1 | Top-10 |
|---|---|---|
| Standard chromagram | 0.561 | 0.723 |
| Enhanced chromagram (baseline) | 0.824 | 0.861 |
| DNN-based chromagram (CTC) | 0.867 | 0.942 |
| DNN-based chromagram (linear scaling) | 0.829 | 0.914 |
| DNN-based chromagram (strong alignment) | 0.882 | 0.939 |



# Theme-Based Audio Retrieval

## References


- R. Bittner, B. McFee, J. Salamon, P. Li, and J. Bello: Deep salience representations for F0 tracking in polyphonic music. Proc. ISMIR, pages 63–70, 2017.
- A. Graves, S. Fernández, F. J. Gomez, and J. Schmidhuber: Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. ICML, 2006.
- F. Zalkow, S. Balke, V. Arifi-Müller, and M. Müller. MTD: A multimodal dataset of musical themes for MIR research. TISMIR, 3(1), 2020.
- F. Zalkow, S. Balke, and M. Müller. Evaluating salience representations for cross-modal retrieval of Western classical music recordings. Proc. ICASSP, 2019.
- F. Zalkow and M. Müller. CTC-based learning of deep chroma features for score-audio music retrieval. 2021. IEEE/ACM Trans. on Audio, Speech, and Language Processing, 29, pages 2957–2971, 2021.



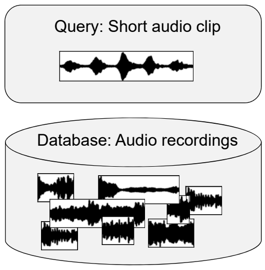
Thanks:
Frank Zalkow (Ph.D. 2021)
Stefan Balke (Ph.D. 2018)




# Audio Matching

## Task

Given a short query audio clip, find corresponding audio clips of similar musical content.




## Challenges

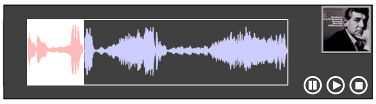
- Similarity measure
  - Different performances
  - Instrumentation may change
  - Similar harmonic progression
- Local comparison
  - Query is short
  - Database recordings are long
- Efficiency
  - Database may be huge



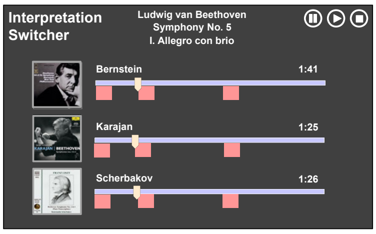
# Audio Matching

## Task

Query:



Database: Matches



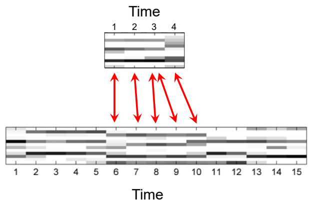



# Audio Matching

## Task

Query: Sequence $X$

Database: Sequence $Y$

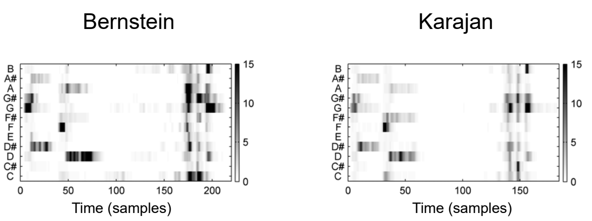



Subsequence matching



# Audio Features

Example: Beethoven's Fifth



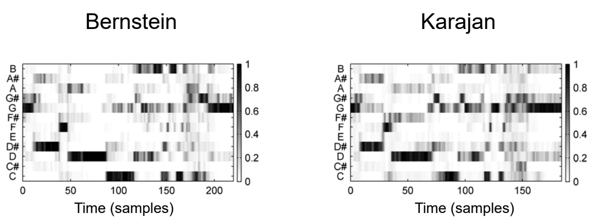

Chroma representation (10 Hz)

**Chroma Features**
Müller, Kurth, Clausen: Audio Matching via Chroma-Based Statistical Features. ISMIR, 2005



# Audio Features

Example: Beethoven's Fifth

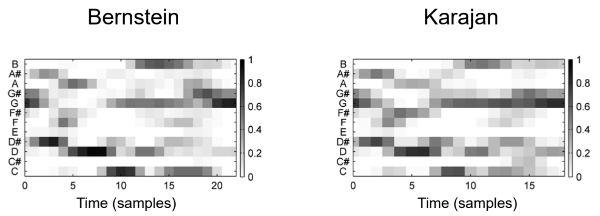



Chroma representation (10 Hz)
- Normalization

**Chroma Features**
Müller, Kurth, Clausen: Audio Matching via Chroma-Based Statistical Features. ISMIR, 2005

## Audio Features

Example: Beethoven's Fifth

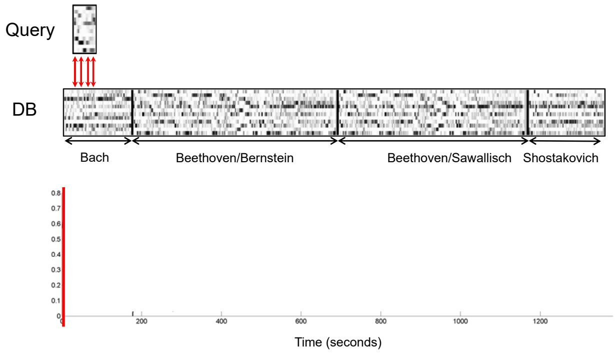



Chroma representation (1 Hz)
- Normalization
- Smoothing & downsampling

**Chroma Features**
Müller, Kurth, Clausen: Audio Matching via Chroma-Based Statistical Features. ISMIR, 2005

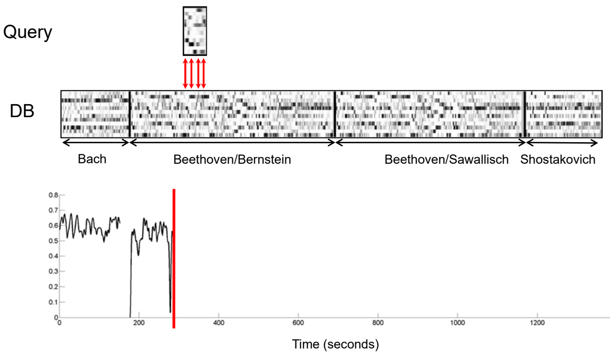


## Matching Procedure



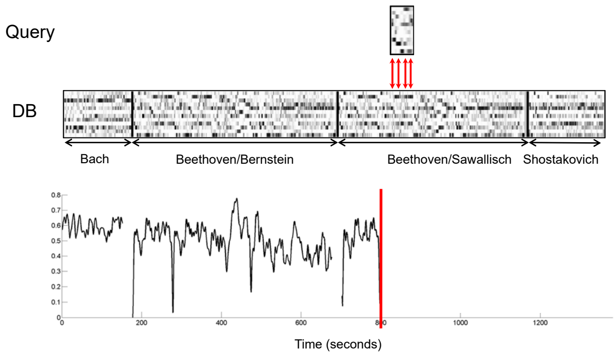



## Matching Procedure



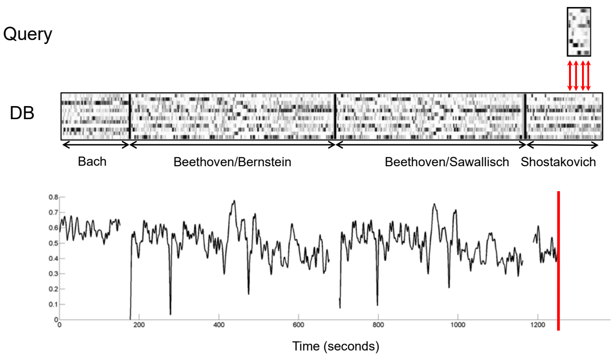



## Matching Procedure






## Matching Procedure

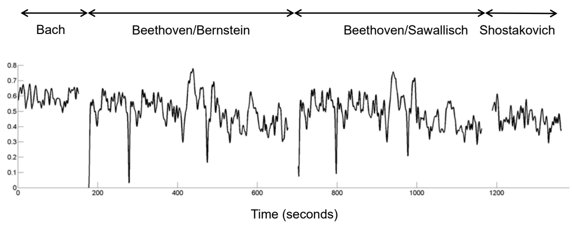





## Matching Procedure

**Matching curve**

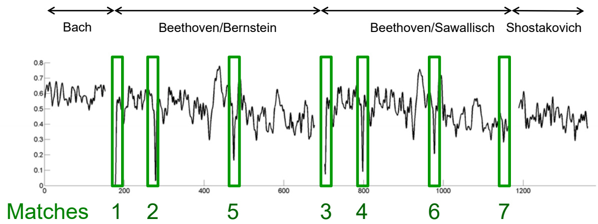
Query: Beethoven's Fifth / Bernstein (first 20 seconds)



## Matching Procedure

**Matching curve**

Query: Beethoven's Fifth / Bernstein (first 20 seconds)

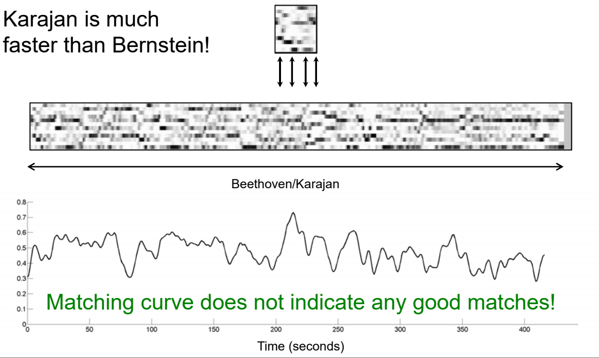





## Matching Procedure

Problem: How to deal with tempo differences?

Karajan is much
faster than Bernstein!



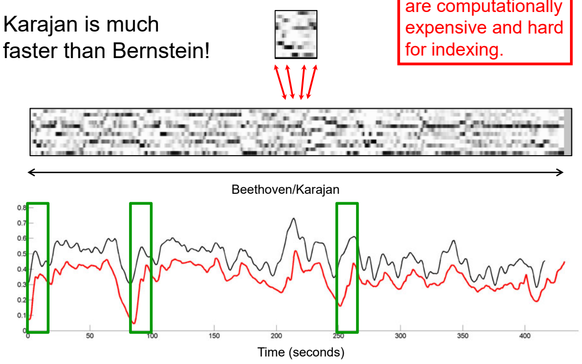



## Matching Procedure

1. Strategy: Usage of local warping

Karajan is much
faster than Bernstein!

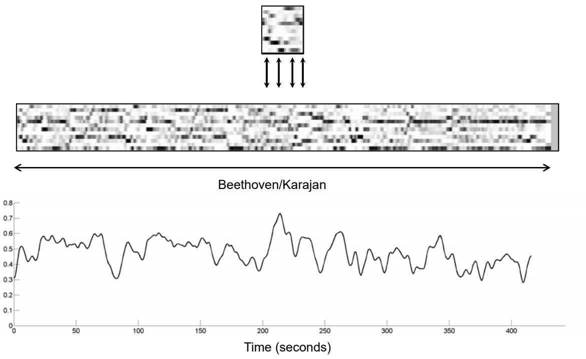
Warping strategies are computationally expensive and hard for indexing.






## Matching Procedure

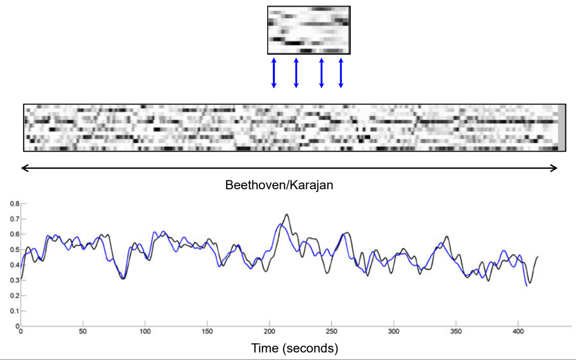
2. Strategy: Usage of multiple scaling






## Matching Procedure

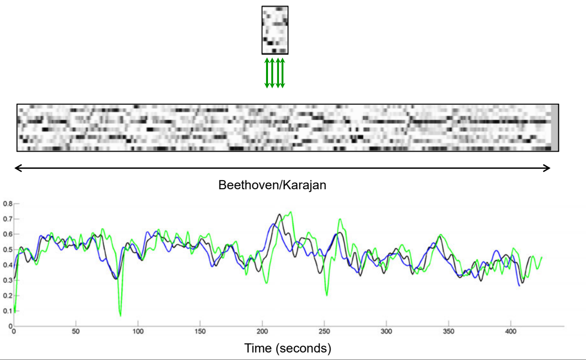
2. Strategy: Usage of multiple scaling






## Matching Procedure

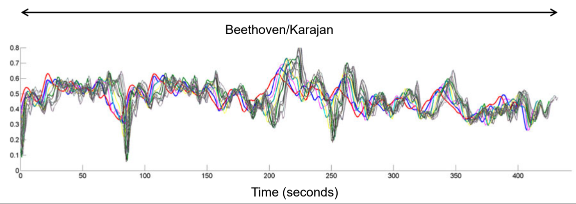
2. Strategy: Usage of multiple scaling



## Matching Procedure

### 2. Strategy: Usage of multiple scaling

- Query resampling simulates tempo changes

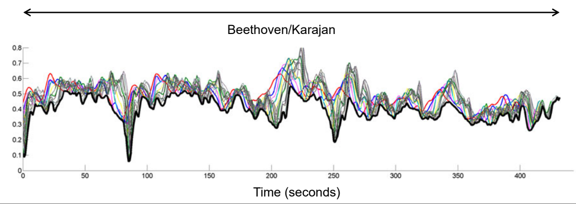





## Matching Procedure

### 2. Strategy: Usage of multiple scaling

- Query resampling simulates tempo changes
- Minimize over all curves

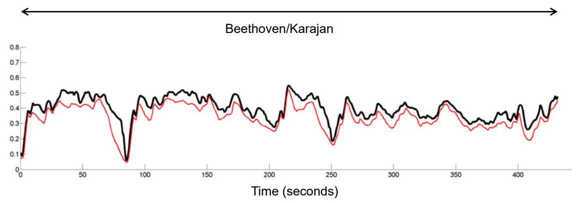





## Matching Procedure

### 2. Strategy: Usage of multiple scaling

- Query resampling simulates tempo changes
- Minimize over all curves
- Resulting curve is similar to warping curve






## Audio Matching

### Query: Beethoven's Fifth / Bernstein (first 20 seconds)

| Rank | Piece | Position |
|---|---|---|
| 1 | Beethoven's Fifth/Bernstein | 0 - 21 |
| 2 | Beethoven's Fifth/Bernstein | 101- 122 |
| 3 | Beethoven's Fifth/Karajan | 86 - 103 |
| ⋮ | ⋮ | ⋮ |
| ⋮ | ⋮ | ⋮ |
| 10 | Beethoven's Fifth/Karajan | 252 - 271 |
| 11 | Beethoven's Fifth/Scherbakov | 0 - 19 |
| 12 | Beethoven's Fifth/Sawallisch | 275 - 296 |
| 13 | Beethoven's Fifth/Scherbakov | 86 - 103 |
| 14 | Schumann Op. 97,1/Levine | 28 - 43 |



## Audio Matching

### Strategy: Handle variations at various levels

- Chroma → invariance to timbre
- Normalization → invariance to dynamics
- Smoothing → invariance to local time deviations
- Multiple queries → invariance to global tempo

**Audio Matching**
Müller, Kurth, Clausen: Audio Matching via Chroma-Based Statistical Features. ISMIR, 2005

Notes:

- There is no "standard" chroma feature.
  → Variants can make a huge difference!
- Learn invariance from examples
  → "Deep Chroma"
- Temporal warping makes problem hard
- Efficiency

**Deep Chroma**
Korzeniowski, Widmer: Feature Learning for Chord Recognition: The Deep Chroma Extractor. ISMIR, 2016



## Shingle-Based Retrieval

### Idea

- Query and database are split up into small overlapping shingles that consist of short feature subsequences.
- Shingles can be matched using efficient nearest neighbor retrieval.
- Trade-off:
  - Large shingles have high musical relevance
  - High shingle dimensionality makes indexing difficult

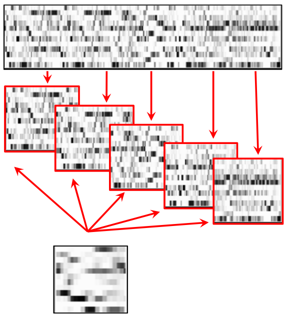
# Shingle-Based Retrieval

Database
Chroma sequence

Chroma shingles

Retrieval
(index-based)

Query
Chroma sequence
(ca. 10 to 30 seconds)

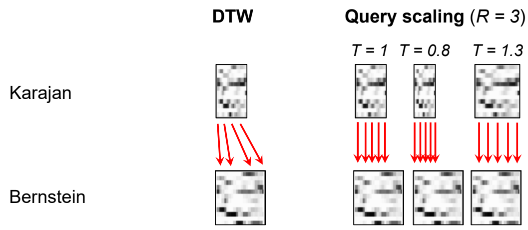




# Shingle-Based Retrieval

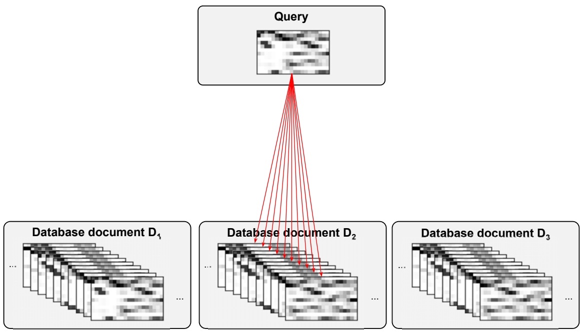
## Tempo-invariant matching

Avoiding expensive temporal warping, tempo differences are handled by creating *R* scaled variants of the query, each simulating a global change in tempo of up to ± 50 %.



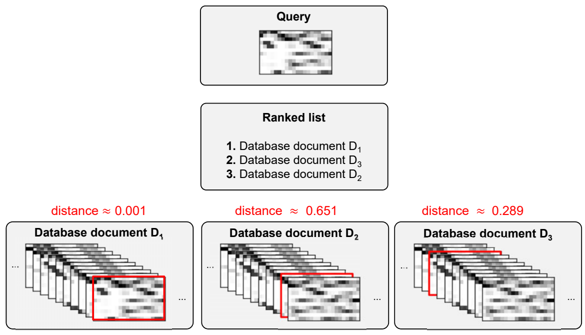



# Shingle-Based Retrieval






# Shingle-Based Retrieval





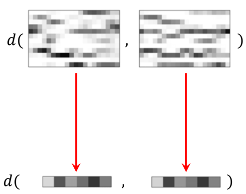

# Shingle-Based Retrieval

## Dimensionality Reduction

Retrieval based on distance computation between shingles

Expensive for high shingle dimensions

Strategy: dimensionality reduction



**1.** Using classical PCA

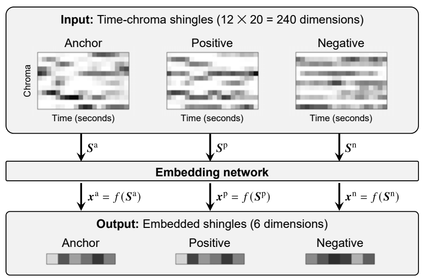
**2.** Using a neural network trained with triplet loss

**Triplet Loss**
F. Schroff, D. Kalenichenko, J. Philbin: FaceNet: A unified embedding for face recognition and clustering. CVPR, 2015.



# Shingle-Based Retrieval

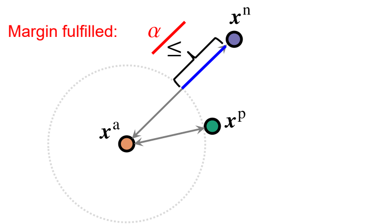
## Triplet-Based Embedding



## Shingle-Based Retrieval

### Triplet Loss

$$\mathcal{L}(X) = \max\left(0, d(\boldsymbol{x}^{\mathrm{a}}, \boldsymbol{x}^{\mathrm{p}}) - d(\boldsymbol{x}^{\mathrm{a}}, \boldsymbol{x}^{\mathrm{n}}) + \alpha\right)$$

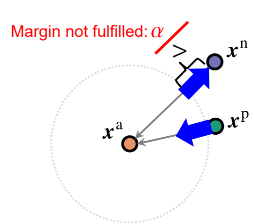





## Shingle-Based Retrieval

### Triplet Loss

$$\mathcal{L}(X) = \max\left(0, d(\boldsymbol{x}^{\mathrm{a}}, \boldsymbol{x}^{\mathrm{p}}) - d(\boldsymbol{x}^{\mathrm{a}}, \boldsymbol{x}^{\mathrm{n}}) + \alpha\right)$$




Loss tries to

- **push** $\boldsymbol{x}^{\mathrm{n}}$ from anchor $\boldsymbol{x}^{\mathrm{a}}$
- **pull** $\boldsymbol{x}^{\mathrm{p}}$ towards anchor $\boldsymbol{x}^{\mathrm{a}}$

until margin $\alpha$ is fulfilled

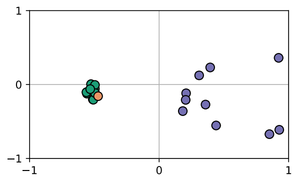


## Shingle-Based Retrieval

### Triplet Loss

$$\mathcal{L}(X) = \max\left(0, d(\boldsymbol{x}^{\mathrm{a}}, \boldsymbol{x}^{\mathrm{p}}) - d(\boldsymbol{x}^{\mathrm{a}}, \boldsymbol{x}^{\mathrm{n}}) + \alpha\right)$$

Embeddings after training






## Shingle-Based Retrieval

### Experiment

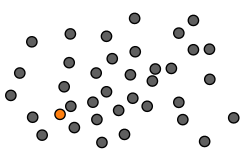
- Training set: 357 recordings of different pieces by Beethoven, Chopin, and Vivaldi (~ 19 hours)
- Test set: 330 different recordings of different pieces by the same composers (~ 16 hours)

| Shingle Reduction | Dimensionality | Retrieval Quality P@1 | Retrieval Quality MAP | Retrieval Time (seconds) |
|---|---|---|---|---|
| No reduction | 240 | 0.996 | 0.972 | 23.0 |
| DNN | 30 | 0.981 | 0.959 | 3.4 |
| DNN | 12 | 0.964 | 0.928 | 1.8 |
| DNN | 6 | 0.890 | 0.856 | 1.2 |



## Shingle-Based Retrieval

### Nearest Neighbor Search



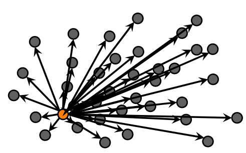


## Shingle-Based Retrieval

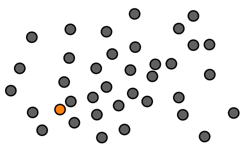
### Nearest Neighbor Search

### Strategies

- Brute force



## Shingle-Based Retrieval
### Nearest Neighbor Search



### Strategies

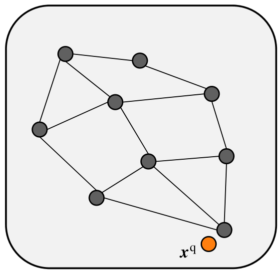
- Brute force
- K-D trees
- HNSW graphs

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### Graph-Based Nearest Neighbor Search

Initial situation



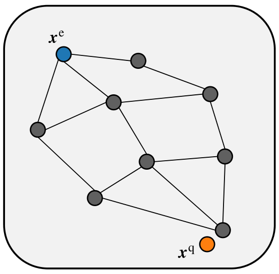

- Given: query node $x^{\mathrm{q}}$

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### Graph-Based Nearest Neighbor Search

Step 1



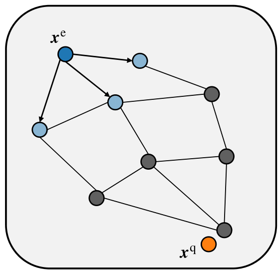

- Given: query node $x^{\mathrm{q}}$
- Start with (random) entry node $x^{\mathrm{e}}$

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### Graph-Based Nearest Neighbor Search

Step 1



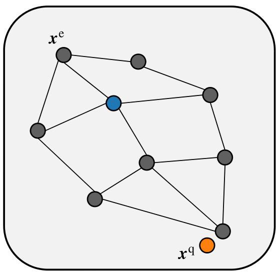

- Given: query node $x^{\mathrm{q}}$
- Start with (random) entry node $x^{\mathrm{e}}$
- Traverse graph along edges and compare nodes with $x^{\mathrm{q}}$

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### Graph-Based Nearest Neighbor Search

Step 2



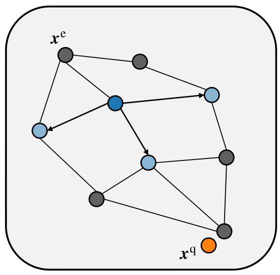

- Given: query node $x^{\mathrm{q}}$
- Start with (random) entry node $x^{\mathrm{e}}$
- Traverse graph along edges and compare nodes with $x^{\mathrm{q}}$
- Continue with closest node

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### Graph-Based Nearest Neighbor Search

Step 2

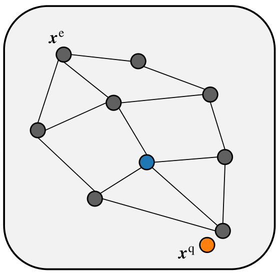



- Given: query node $x^{\mathrm{q}}$
- Start with (random) entry node $x^{\mathrm{e}}$
- Traverse graph along edges and compare nodes with $x^{\mathrm{q}}$
- Continue with closest node

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.

## Shingle-Based Retrieval
### Graph-Based Nearest Neighbor Search

Step 3



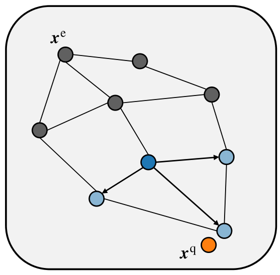

- Given: query node $x^{\mathrm{q}}$
- Start with (random) entry node $x^{\mathrm{e}}$
- Traverse graph along edges and compare nodes with $x^{\mathrm{q}}$
- Continue with closest node

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### Graph-Based Nearest Neighbor Search

Step 3



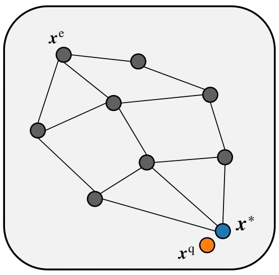

- Given: query node $x^{\mathrm{q}}$
- Start with (random) entry node $x^{\mathrm{e}}$
- Traverse graph along edges and compare nodes with $x^{\mathrm{q}}$
- Continue with closest node

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### Graph-Based Nearest Neighbor Search

Step 4

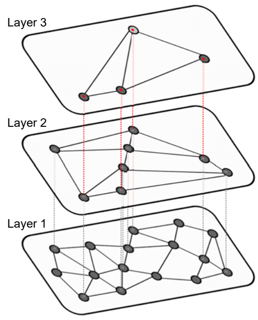



- Given: query node $x^{\mathrm{q}}$
- Start with (random) entry node $x^{\mathrm{e}}$
- Traverse graph along edges and compare nodes with $x^{\mathrm{q}}$
- Continue with closest node
- Stop when distances increase

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



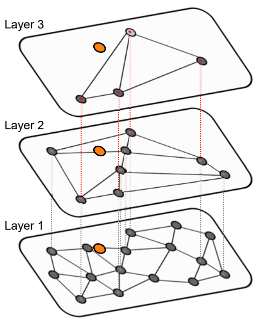
## Shingle-Based Retrieval
### HNSW Graphs




**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



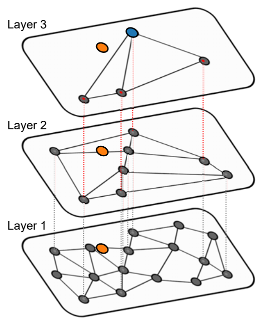
## Shingle-Based Retrieval
### HNSW Graphs




**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.

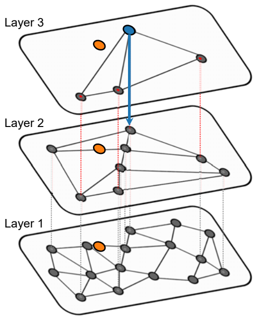


## Shingle-Based Retrieval
### HNSW Graphs




**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.

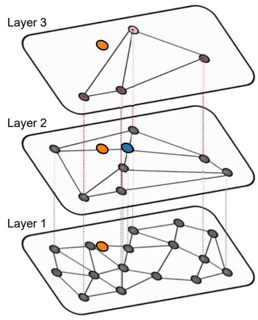
## Shingle-Based Retrieval
### HNSW Graphs




**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



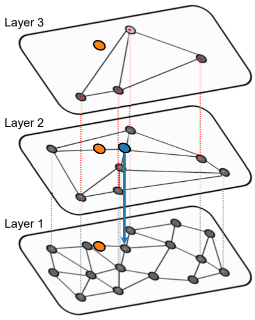
## Shingle-Based Retrieval
### HNSW Graphs




**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



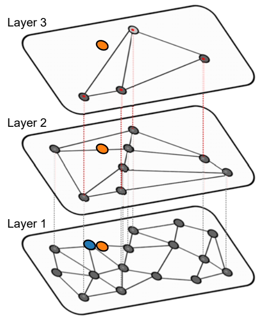
## Shingle-Based Retrieval
### HNSW Graphs




**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### HNSW Graphs




### Properties

- Approximate nearest neighbor search
- Search runtime logarithmic in dataset size
- Works well with high dimensional data
- Efficient algorithm to build graph structure

**HNSW Graphs**
Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.



## Shingle-Based Retrieval
### Experiment

- Approximate search yields nearly same results as exact search
- Dataset: Entire audio catalogue by Carus publisher (7115 recordings, ~ 390 hours, > 1,25 million shingles)
- Runtime for brute force approach: ~ 100 ms to 300 ms per query

| Search | Shingle Reduction | Dimensionality | Time (ms) |
|---|---|---|---|
| KD | No reduction | 240 | 772.95 |
| KD | DNN | 30 | 117.54 |
| KD | DNN | 12 | 7.24 |
| KD | DNN | 6 | 0.66 |
| HNSW | No reduction | 240 | 0.20 |
| HNSW | DNN | 30 | 0.08 |
| HNSW | DNN | 12 | 0.06 |
| HNSW | DNN | 6 | 0.06 |



## Shingle-Based Retrieval
### References

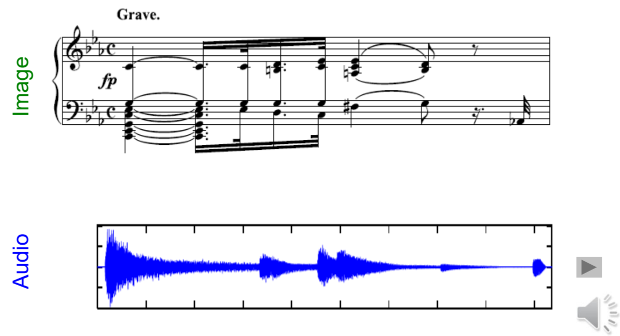
- P. Grosche, M. Müller: Toward characteristic audio shingles for efficient cross-version music retrieval. IEEE ICASSP, pages 473-476, 2012
- Y. Malkov and D. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Transactions on PAMI, 2020.
- F. Schroff, D. Kalenichenko, J. Philbin: FaceNet: A unified embedding for face recognition and clustering. CVPR, 2015.
- F. Zalkow and M. Müller: Learning low-dimensional embeddings of audio shingles for cross-version retrieval of classical music. Applied Sciences, 10(1), 2020.
- F. Zalkow, J. Brandner, and M. Müller: Efficient retrieval of music recordings using graph-based index structures. Signals, 2(2), 2021.

Thanks:
Frank Zalkow (Ph.D. 2021)

## Music Synchronization: Image-Audio

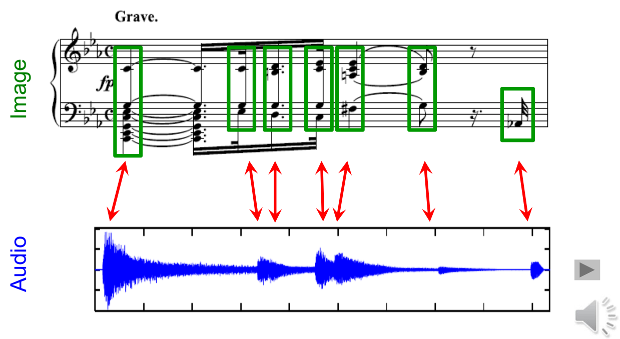


## Music Synchronization: Image-Audio



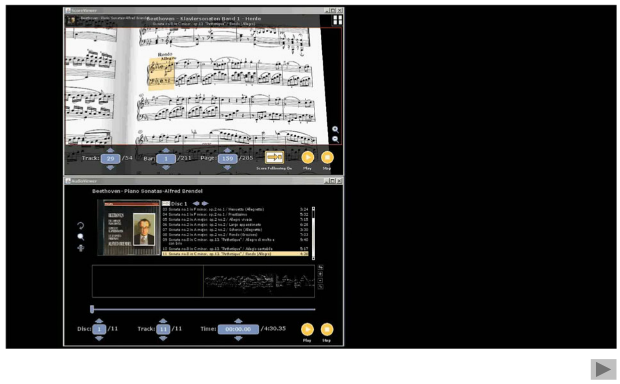



## Music Synchronization: Image-Audio



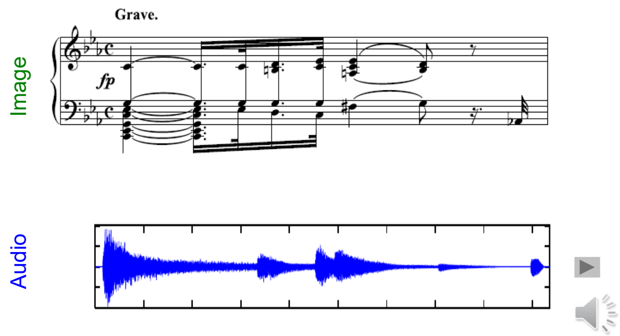



## Application: Score Viewer





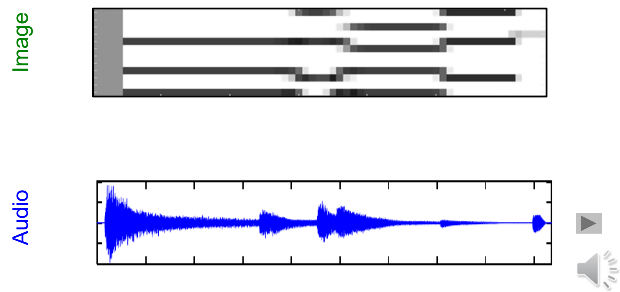
## Music Synchronization: Image-Audio





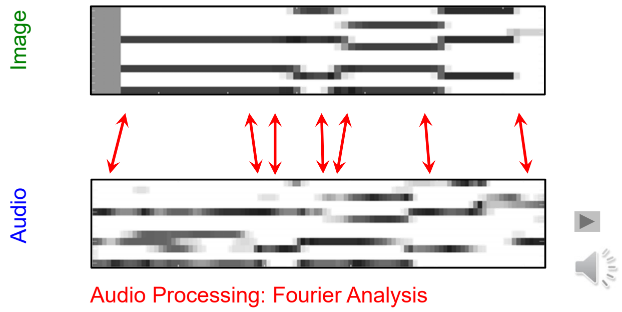

## Music Synchronization: Image-Audio

Image Processing: Optical Music Recognition



# Music Synchronization: Image-Audio

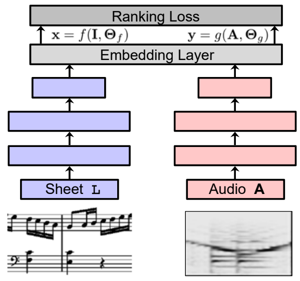
Image Processing: Optical Music Recognition




Audio Processing: Fourier Analysis



# Music Synchronization: Image-Audio




- Representation learning
- Embedding techniques
- Weak annotations
- Loss functions
- …

**Cross-Modal Retrieval**
Dorfer et al.: End-to-End Cross-Modality Retrieval with CCA Projections and Pairwise Ranking Loss. International Journal of Multimedia Information Retrieval, 2018.



# Music Retrieval