Introduction to **Audio Content Analysis**

module 10.2: audio-to-audio & audio-to-score alignment

alexander lerch

Georgia Tech Center for Music Technology
College of Design

introduction
overview

corresponding textbook section

sections 10.2 & 10.3

- **lecture content**
  - Audio-to-Audio alignment
    - use cases
    - features
    - distance measures
    - typical accuracy
  - Audio-to-Score alignment
- **learning objectives**
  - elaborate on possible use cases for audio-to-audio alignment
  - give examples for features and distance measures for alignment
  - discuss differences between audio-to-audio and audio-to-score alignment

corresponding textbook section

sections 10.2 & 10.3

- **lecture content**
  - Audio-to-Audio alignment
    - use cases
    - features
    - distance measures
    - typical accuracy
  - Audio-to-Score alignment

- **learning objectives**
  - elaborate on possible use cases for audio-to-audio alignment
  - give examples for features and distance measures for alignment
  - discuss differences between audio-to-audio and audio-to-score alignment

audio-to-audio alignment

introduction

- **objective**
  - align two sequences of audio
- **use cases**
  - *quick browsing* for certain parts in recordings
  - *timing adjustment* (backing vocals, loops, ...)
  - *automated dubbing*
  - *musicological analysis* (relative timing of several performances)
- **processing steps**
  - extract suitable features
  - compute distance matrix
  - compute alignment path

audio-to-audio alignment
introduction

- **objective**
  - align two sequences of audio
- **use cases**
  - *quick browsing* for certain parts in recordings
  - *timing adjustment* (backing vocals, loops, . . .)
  - *automated dubbing*
  - *musicological analysis* (relative timing of several performances)
- **processing steps**
  - extract suitable features
  - compute distance matrix
  - compute alignment path

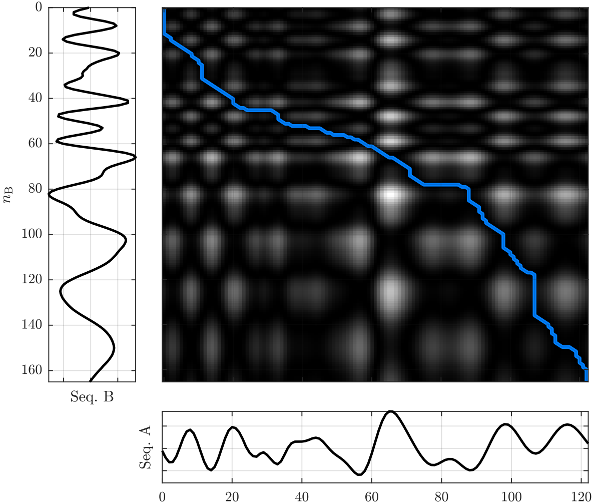
audio-to-audio alignment
introduction

- **objective**
  - align two sequences of audio
- **use cases**
  - *quick browsing* for certain parts in recordings
  - *timing adjustment* (backing vocals, loops, . . .)
  - *automated dubbing*
  - *musicological analysis* (relative timing of several performances)
- **processing steps**
  - extract suitable features
  - compute distance matrix
  - compute alignment path

audio-to-audio alignment

alignment path computation

- prerequisite:
  Module 10.1 — Dynamic Time Warping



matlab source: plotDtwPath.m

audio-to-audio alignment
features

- **use case examples**
  - **quick browsing** — find the same part across files
    ⇒ use *pitch based* features
  - **timing adjustment** — backing vocals to lead vocals
    ⇒ use *intensity based* features
  - **automated dubbing** — same speaker several recordings
    ⇒ use *intensity based* and *timbre based* features
- **feature categories**
  - **intensity**: energy, onset probability, ...
  - **tonal**: pitch chroma, ...
  - **timbral**: MFCCs, spectral shape, ...

plot from[1]

[1] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-audio alignment
features

- **use case examples**
    - **quick browsing** — find the same part across files
      ⇒ use *pitch based* features
    - **timing adjustment** — backing vocals to lead vocals
      ⇒ use *intensity based* features
    - **automated dubbing** — same speaker several recordings
      ⇒ use *intensity based* and *timbre based* features

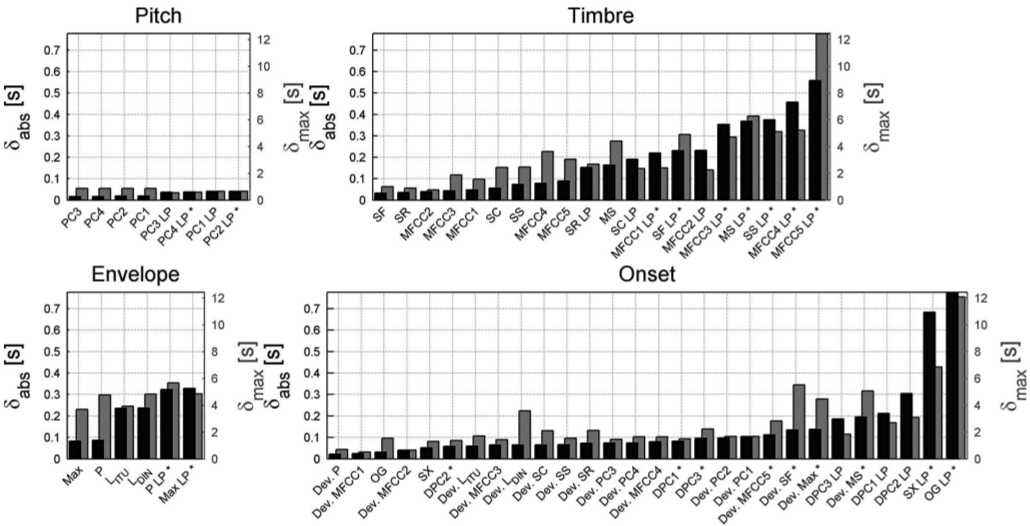
- **feature categories**
    - **intensity**: energy, onset probability, ...
    - **tonal**: pitch chroma, ...
    - **timbral**: MFCCs, spectral shape, ...

plot from[1]

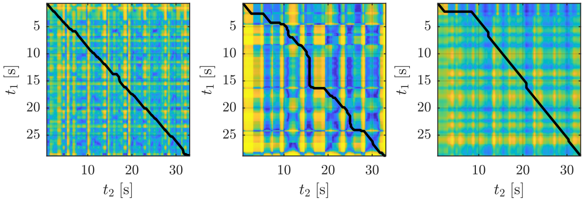
[1] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-audio alignment

features



audio-to-audio alignment
feature-dependency of results



features (left to right): pitch chroma, RMS, MFCCs

matlab source: plotDtwFeatures.m

audio-to-audio alignment

compute distance matrix — distance measures

- typical distance measures
  - *Euclidean distance*: $d_\mathrm{E}(s) = \sqrt{\sum\limits_{j=0}^{11} \left(\nu_\mathrm{e}(j) - \nu_\mathrm{t,s}(j)\right)^2}$
  - *Manhattan distance*: $d_\mathrm{M}(s) = \sum\limits_{j=0}^{11} \left|\nu_\mathrm{e}(j) - \nu_\mathrm{t,s}(j)\right|$
  - *Cosine distance*: $d_\mathrm{C}(s) = 1 - \left(\frac{\sum\limits_{j=0}^{11} \nu_\mathrm{e}(j)\cdot\nu_\mathrm{t,s}(j)}{\sqrt{\sum\limits_{j=0}^{11} \nu_\mathrm{e}(j)^2}\sqrt{\cdot\sum\limits_{j=0}^{11} \nu_\mathrm{t,s}(j)^2}}\right)$
  - *Kullback-Leibler divergence*: $d_\mathrm{KL}(s) = \sum\limits_{j=0}^{11} \nu_\mathrm{e}(j) \cdot \log\left(\frac{\nu_\mathrm{e}(j)}{\nu_\mathrm{t,s}(j)}\right)$
- data-driven approach: train classifier with 2-class problem[1]

[1] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-audio alignment

compute distance matrix — distance measures

- typical distance measures
  - *Euclidean distance*: $d_\mathrm{E}(s) = \sqrt{\sum\limits_{j=0}^{11} \left(\nu_\mathrm{e}(j) - \nu_\mathrm{t,s}(j)\right)^2}$
  - *Manhattan distance*: $d_\mathrm{M}(s) = \sum\limits_{j=0}^{11} \left|\nu_\mathrm{e}(j) - \nu_\mathrm{t,s}(j)\right|$
  - *Cosine distance*: $d_\mathrm{C}(s) = 1 - \left(\frac{\sum\limits_{j=0}^{11} \nu_\mathrm{e}(j)\cdot\nu_\mathrm{t,s}(j)}{\sqrt{\sum\limits_{j=0}^{11} \nu_\mathrm{e}(j)^2}\sqrt{\cdot\sum\limits_{j=0}^{11} \nu_\mathrm{t,s}(j)^2}}\right)$
  - *Kullback-Leibler divergence*: $d_\mathrm{KL}(s) = \sum\limits_{j=0}^{11} \nu_\mathrm{e}(j)\cdot\log\left(\frac{\nu_\mathrm{e}(j)}{\nu_\mathrm{t,s}(j)}\right)$
- data-driven approach: train classifier with 2-class problem[1]

[1] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-audio alignment

compute distance matrix — distance measures

- typical distance measures
  - *Euclidean distance*: $d_\mathrm{E}(s) = \sqrt{\sum\limits_{j=0}^{11}\left(\nu_\mathrm{e}(j) - \nu_\mathrm{t,s}(j)\right)^2}$
  - *Manhattan distance*: $d_\mathrm{M}(s) = \sum\limits_{j=0}^{11}\left|\nu_\mathrm{e}(j) - \nu_\mathrm{t,s}(j)\right|$
  - *Cosine distance*: $d_\mathrm{C}(s) = 1 - \left(\frac{\sum\limits_{j=0}^{11}\nu_\mathrm{e}(j)\cdot\nu_\mathrm{t,s}(j)}{\sqrt{\sum\limits_{j=0}^{11}\nu_\mathrm{e}(j)^2}\sqrt{\cdot\sum\limits_{j=0}^{11}\nu_\mathrm{t,s}(j)^2}}\right)$
  - *Kullback-Leibler divergence*: $d_\mathrm{KL}(s) = \sum\limits_{j=0}^{11}\nu_\mathrm{e}(j)\cdot\log\left(\frac{\nu_\mathrm{e}(j)}{\nu_\mathrm{t,s}(j)}\right)$
- data-driven approach: train classifier with 2-class problem[1]

[1] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-audio alignment

compute distance matrix — distance measures

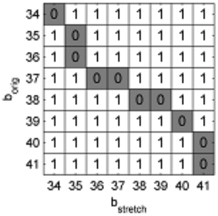
- typical distance measures
    - *Euclidean distance*: $d_\mathrm{E}(s) = \sqrt{\sum\limits_{j=0}^{11}\left(\nu_\mathrm{e}(j) - \nu_\mathrm{t,s}(j)\right)^2}$
    - *Manhattan distance*: $d_\mathrm{M}(s) = \sum\limits_{j=0}^{11}\left|\nu_\mathrm{e}(j) - \nu_\mathrm{t,s}(j)\right|$
    - *Cosine distance*: $d_\mathrm{C}(s) = 1 - \left(\frac{\sum\limits_{j=0}^{11}\nu_\mathrm{e}(j)\cdot\nu_\mathrm{t,s}(j)}{\sqrt{\sum\limits_{j=0}^{11}\nu_\mathrm{e}(j)^2}\sqrt{\cdot\sum\limits_{j=0}^{11}\nu_\mathrm{t,s}(j)^2}}\right)$
    - *Kullback-Leibler divergence*: $d_\mathrm{KL}(s) = \sum\limits_{j=0}^{11}\nu_\mathrm{e}(j)\cdot\log\left(\frac{\nu_\mathrm{e}(j)}{\nu_\mathrm{t,s}(j)}\right)$
- data-driven approach: train classifier with 2-class problem[1]

[1] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-audio alignment

compute distance matrix — distance measures

- typical distance measures
- data-driven approach: train classifier with 2-class problem[1]

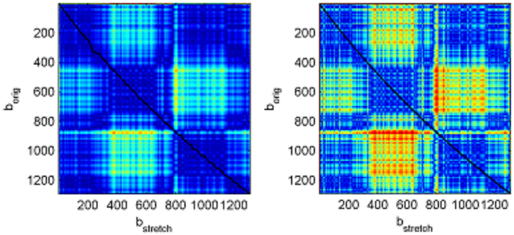


[1] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-audio alignment

compute distance matrix — distance measures

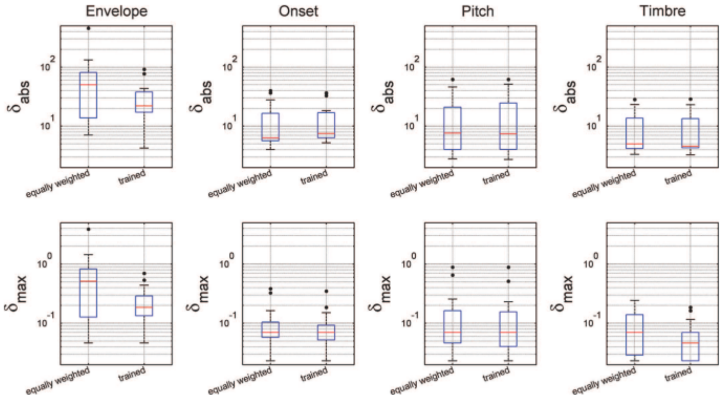
- typical distance measures
- data-driven approach: train classifier with 2-class problem[1]



[1] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-audio alignment

typical results



originals

left: instrumental

right: a capella

synced

[2] H. Kirchhoff and A. Lerch, "Evaluation of Features for Audio-to-Audio Alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011. DOI: 10.1080/09298215.2010.529917.

audio-to-score alignment

overview

- **objective**
  - align an audio sequence with a score sequence
- **use cases**
  - score viewer
  - music education
  - retrieve matching score/audio via cost function
  - musicological analysis
- **processing steps**
  - see audio-to-audio alignment

audio-to-score alignment

overview

- **objective**
  - align an audio sequence with a score sequence
- **use cases**
  - score viewer
  - music education
  - retrieve matching score/audio via cost function
  - musicological analysis
- **processing steps**
  - see audio-to-audio alignment

audio-to-score alignment

overview

- **objective**
  - align an audio sequence with a score sequence
- **use cases**
  - score viewer
  - music education
  - retrieve matching score/audio via cost function
  - musicological analysis
- **processing steps**
  - see audio-to-audio alignment

audio-to-score alignment
challenges

- features from **different domains**
  - score contains no timbre info
  - score cannot be expected to contain no loudness info
  - score has no clear “time axis”

⇒ two prototypical for distance/similarity calculation
  - *approach 1*: convert score into audio-like representation
    - MIDI-to-audio
    - use model synthesize
  - *approach 2*: convert audio into score-like representation
    - audio-to-MIDI
    - pitch chroma
    - event-based segmentation

audio-to-score alignment
challenges

- features from **different domains**
  - score contains no timbre info
  - score cannot be expected to contain no loudness info
  - score has no clear "time axis"

$\Rightarrow$ two prototypical for distance/similarity calculation

- *approach 1*: convert score into audio-like representation
  - MIDI-to-audio
  - use model synthesize
- *approach 2*: convert audio into score-like representation
  - audio-to-MIDI
  - pitch chroma
  - event-based segmentation

alignment
evaluation

- **goal**: compare two sequences of time stamps

- evaluation **challenges**
  - pauses/rests, and long held notes: what is the reference path?
  - noise in the begin and end of the recording
  - data not easily available
    - synthesized
    - piano sensors
    - pseudo-ground truth with time stretching
    - automatic annotation with quality assurance

alignment
evaluation

- **goal**: compare two sequences of time stamps
- evaluation **challenges**
  - pauses/rests, and long held notes: what is the reference path?
  - noise in the begin and end of the recording
  - data not easily available
    - synthesized
    - piano sensors
    - pseudo-ground truth with time stretching
    - automatic annotation with quality assurance

alignment

evaluation metrics

- *audio-to-score*
  - missed note rate
  - misalign rate
  - piece completion
  - average absolute deviation
  - variance of deviation

- *audio-to-audio*
  - mean deviation
  - mean absolute deviation
  - maximum deviation
  - relative number of matching path points

summary
lecture content

- **audio-to-audio alignment**
  1. extract features
  2. create distance matrix with suitable distance measure
  3. use DTW to find alignment path
  4. (use time-stretching to actually align the sequences)

- **audio-to-score alignment**
  1. extract usually pitch-based features
  2. distance measure
  3. use DTW, HMM, etc to extract alignment path