A Sequential Approach to Musical Event Detection

Colin Raffel
Music and Technology Seminar
Carnegie Mellon University

February 17, 2011

Outline

- Introduction and Motivation
- Tempo Estimation
- Beat Tracking
- Downbeat/Measure Detection
- Dynamics Change
- Instrumentation Change
- Melody Change

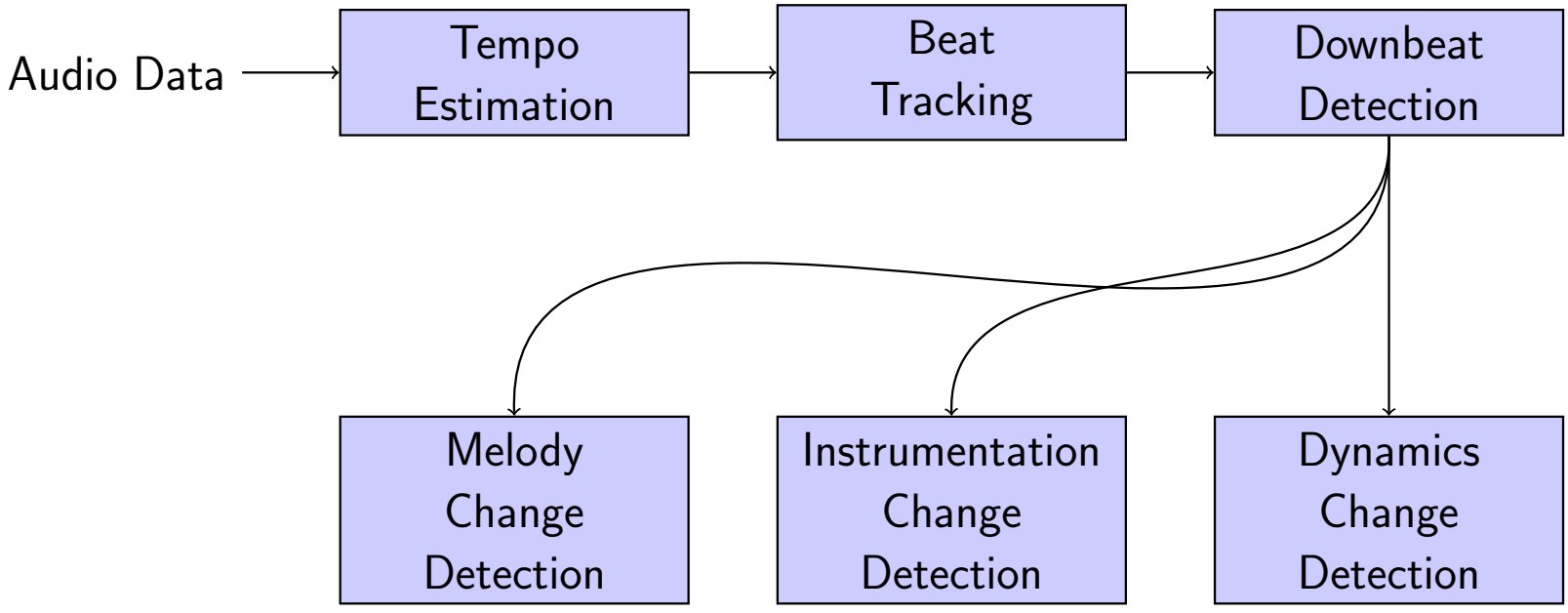
Events

- When something “notable” happens
- For example...
  - Note onsets
  - Rhythmic cues
  - Large-scale changes
- Fixed grid (sometimes)
- Hierarchical (sometimes)
- Agreement across listeners

Motivation

- Uses
  - Synchronization
  - Summarization
  - “Smart” processing
  - Rudimentary classification
- Manual annotation is time-consuming
- Subjectivity makes the problem difficult

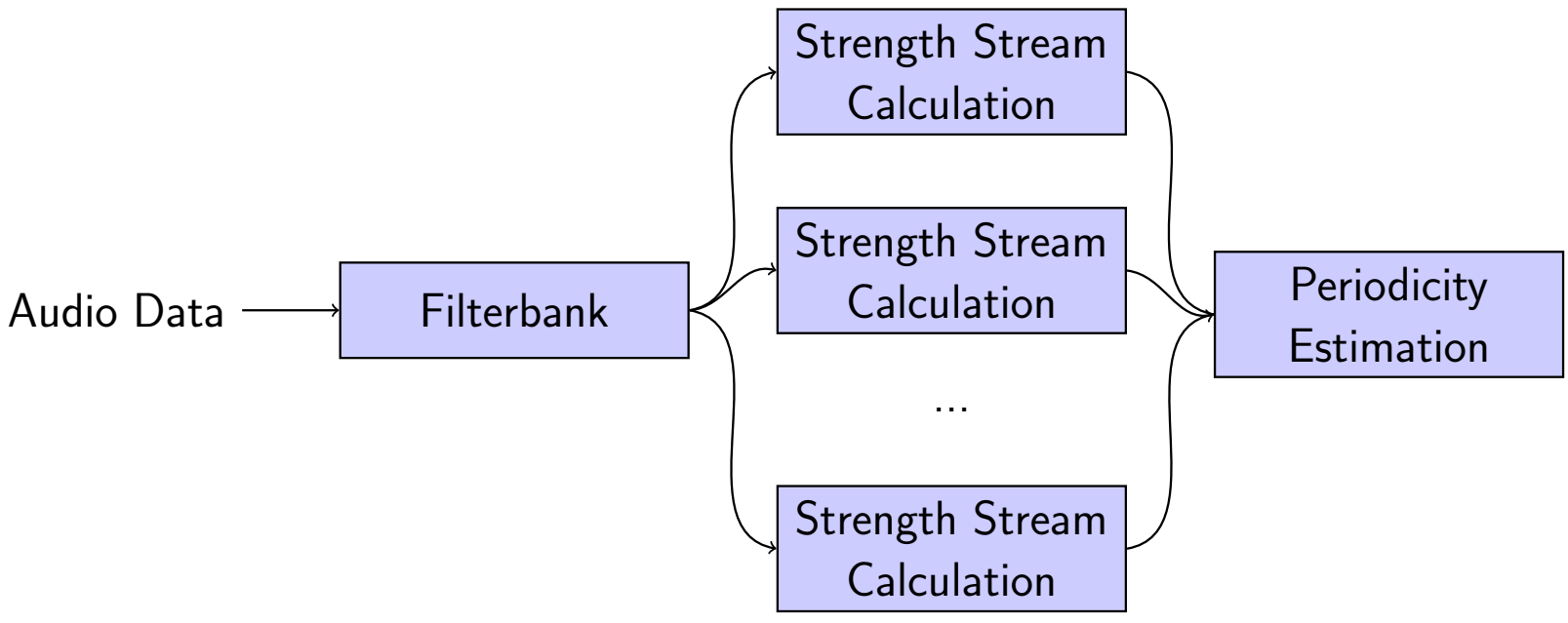
“A Sequential Approach”



Tempo Estimation

- Defines the typical rhythmic interval
- "First pass" at beat detection
- Periodicity of a slow signal
- Difficulty depends on...
  - Beat regularity
  - Rhythmic strength
  - Existence/speed of tempo changes

Typical System



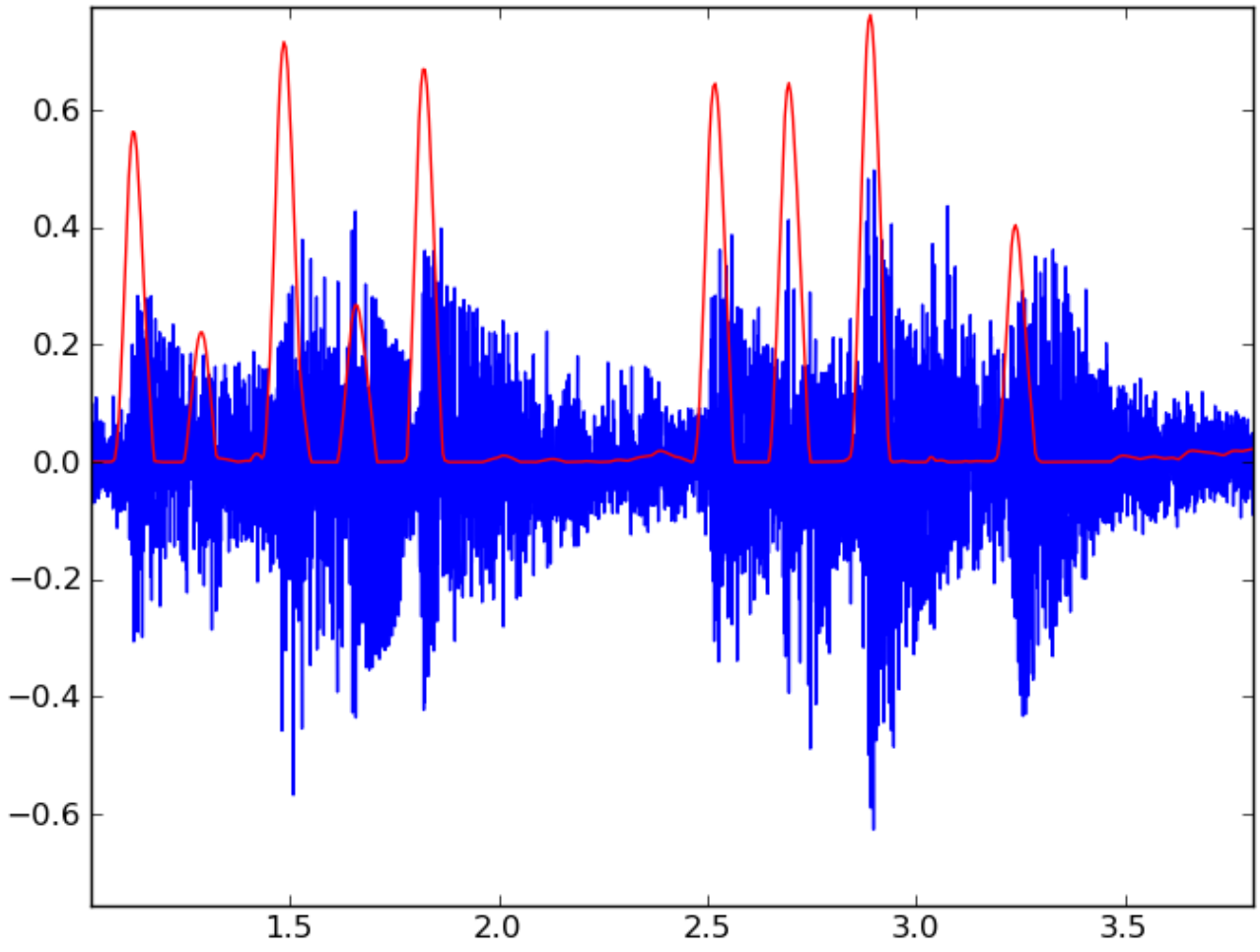
Filterbank

- Look at change in frequency bands
- Number and type of filters varies
  - None
  - STFT
  - Octave-width
  - Mel-scale
  - Instrument-based
- Effectiveness varies

“Strength Stream”

- Derive the “perceptual importance”
- Typical steps:
  - Rectify
  - Smooth
  - Decimate
  - Take logarithm
  - Differentiate
  - Half-wave rectify
- Alternatives:
  - Onset detection
  - Symbolic data

Strength Stream for Piano Notes



Combining Bands

- Unweighted sum
- Perceptual weighting
- Weighting by Spectral Crest Factor
- “Incomplete” combination

Periodicity Measurement

- Pitch detection
  - Autocorrelation techniques
  - YIN Algorithm
  - Harmonic product spectrum (spectral product)
- Inter-onset interval clustering
- Comb filters or phase-locking resonators
- Klapuri argues algorithm used here is not important

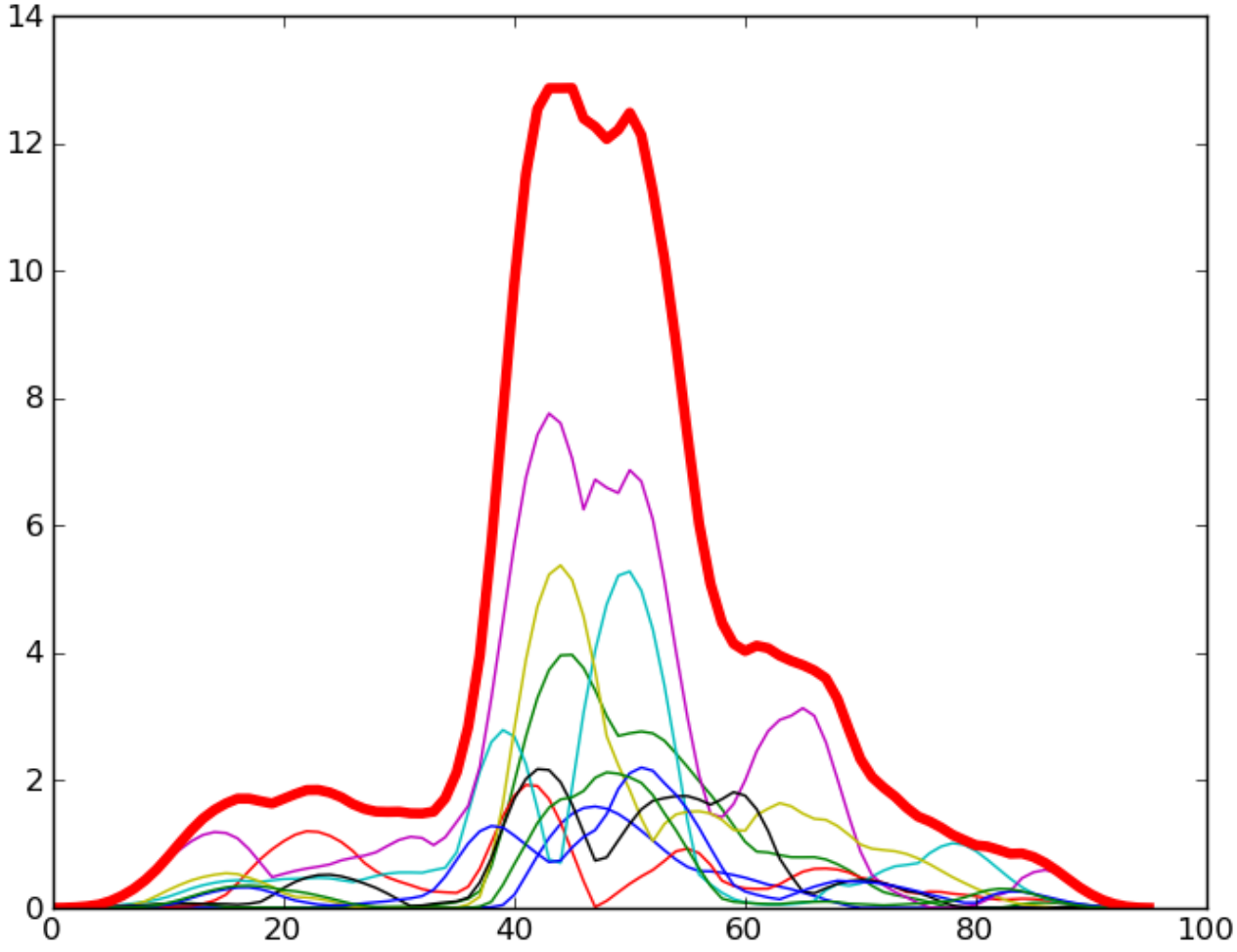
Beat Tracking

- Strength stream $\rightarrow$ semi-regular grid
- Some techniques used:
  - Local search (greedy algorithm)
  - Multi-agent greedy algorithm
  - Dynamic programming
  - Phase-locked resonators
  - Probabilistic (HMMs)
- Accuracy tends to be what you'd expect!

A Greedy Algorithm

- Estimate offset
- Find strength stream maximum near expected location
- Update tempo estimate
- Helpful tweaks:
  - Taper strength stream search window
  - Add expected strength stream
  - Include neighboring expected beats
  - Update tempo according to confidence
  - Update window size according to strength

Extracted beat windows



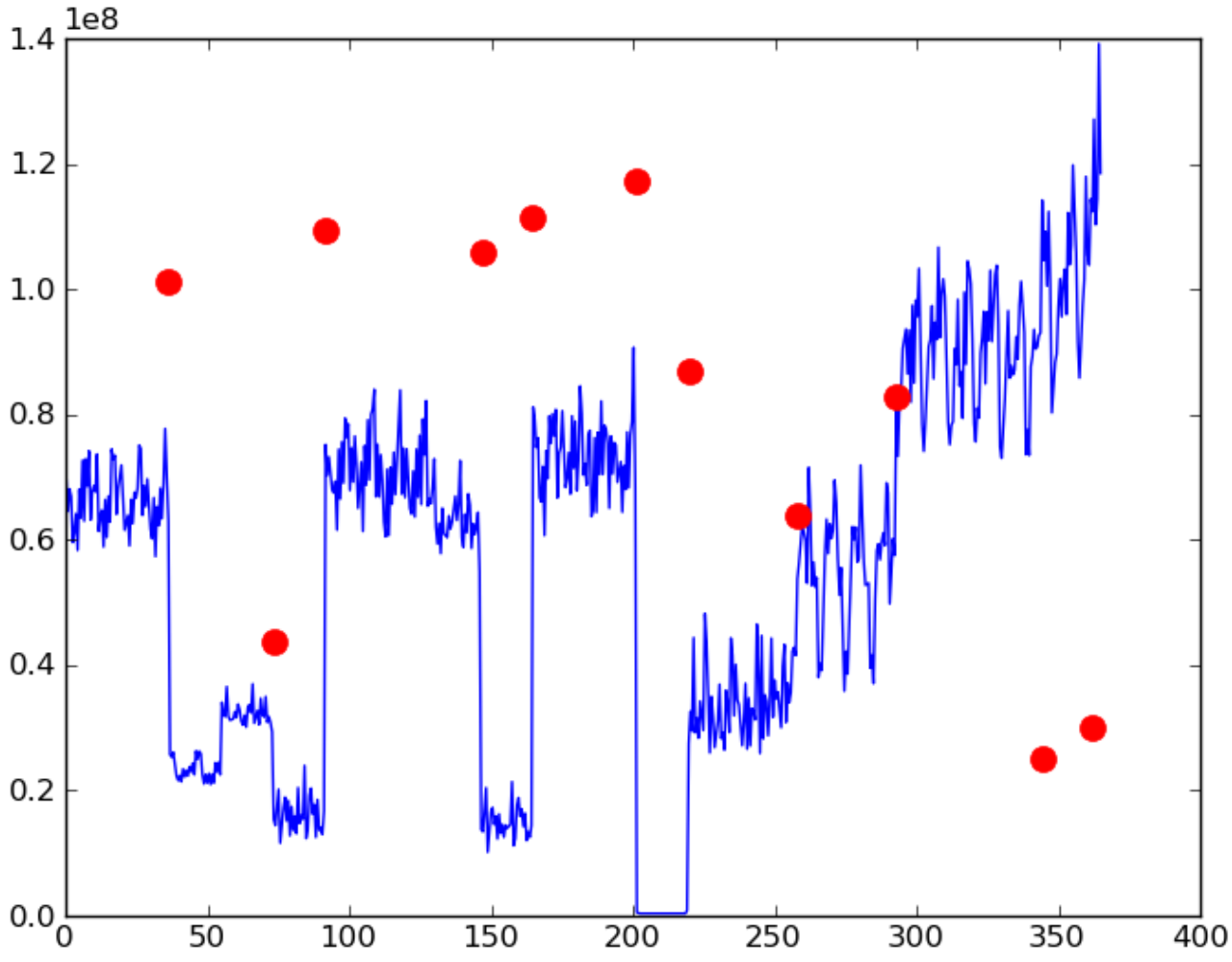
Downbeat/Measure Detection

- Beat grid $\rightarrow$ measure grid
- Naive approach:
  - Assume first onset is first downbeat
  - Assume all beat locations are correct
  - Assume constant meter
- Smarter approaches based on:
  - Perceptual importance
  - Pattern matching
  - Changes in melody/timbre
  - Machine listening/learning

Dynamics Change Detection

- Demarcate where music gets louder or softer
- Loudness measures:
  - RMS
  - Zwicker-Fastl loudness
  - MFC coefficient 0 (FBE)
- Calculate per-beat or per-measure
- Smooth as necessary
- Find large-scale changes in this signal

Loudness signal



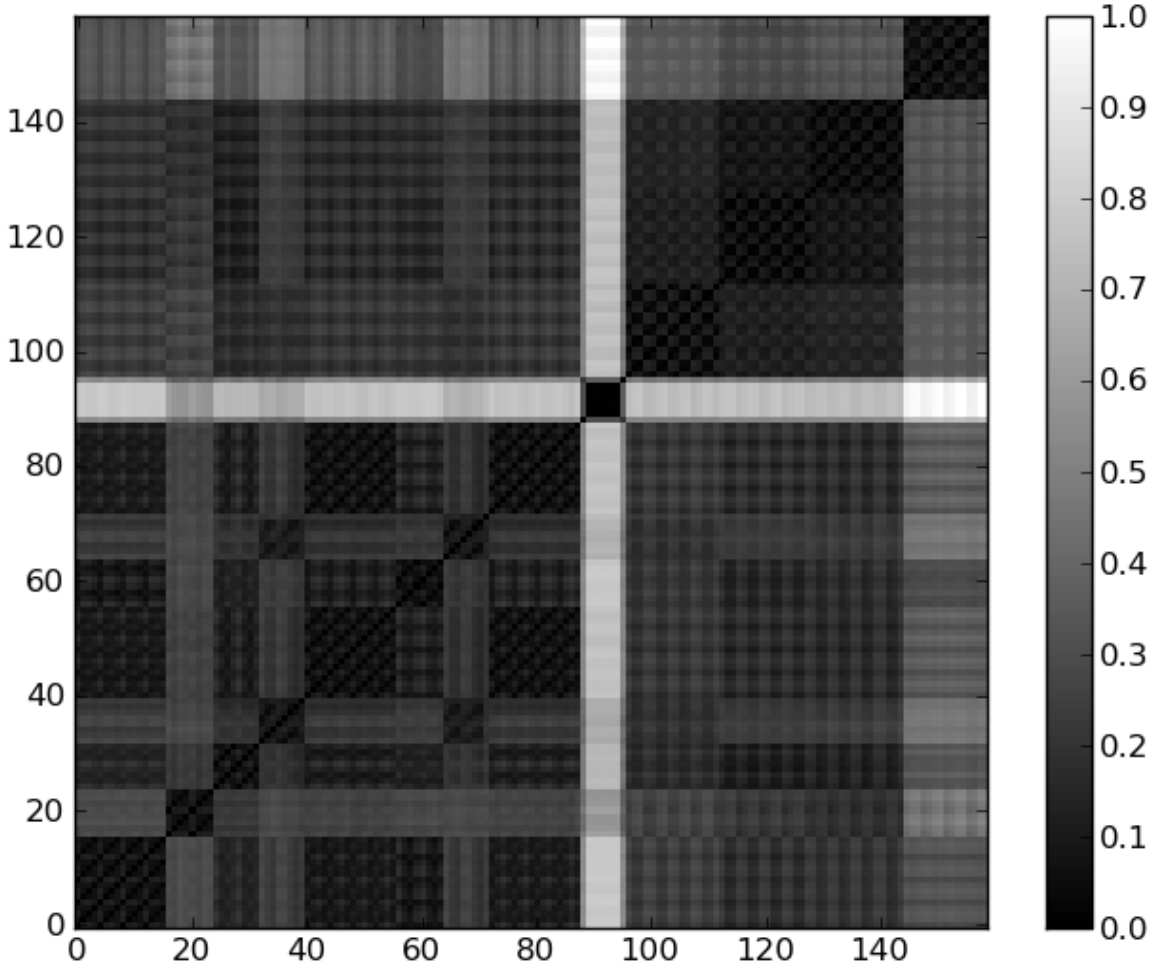
Instrumentation Change Detection

- Entrance or omission of instruments
- Instrument quality change?
- Instrumentation is hard to determine
- Instead, try MFCCs
  - Suggests "timbre change"
  - Also used for segmentation
  - Makes the problem tractable

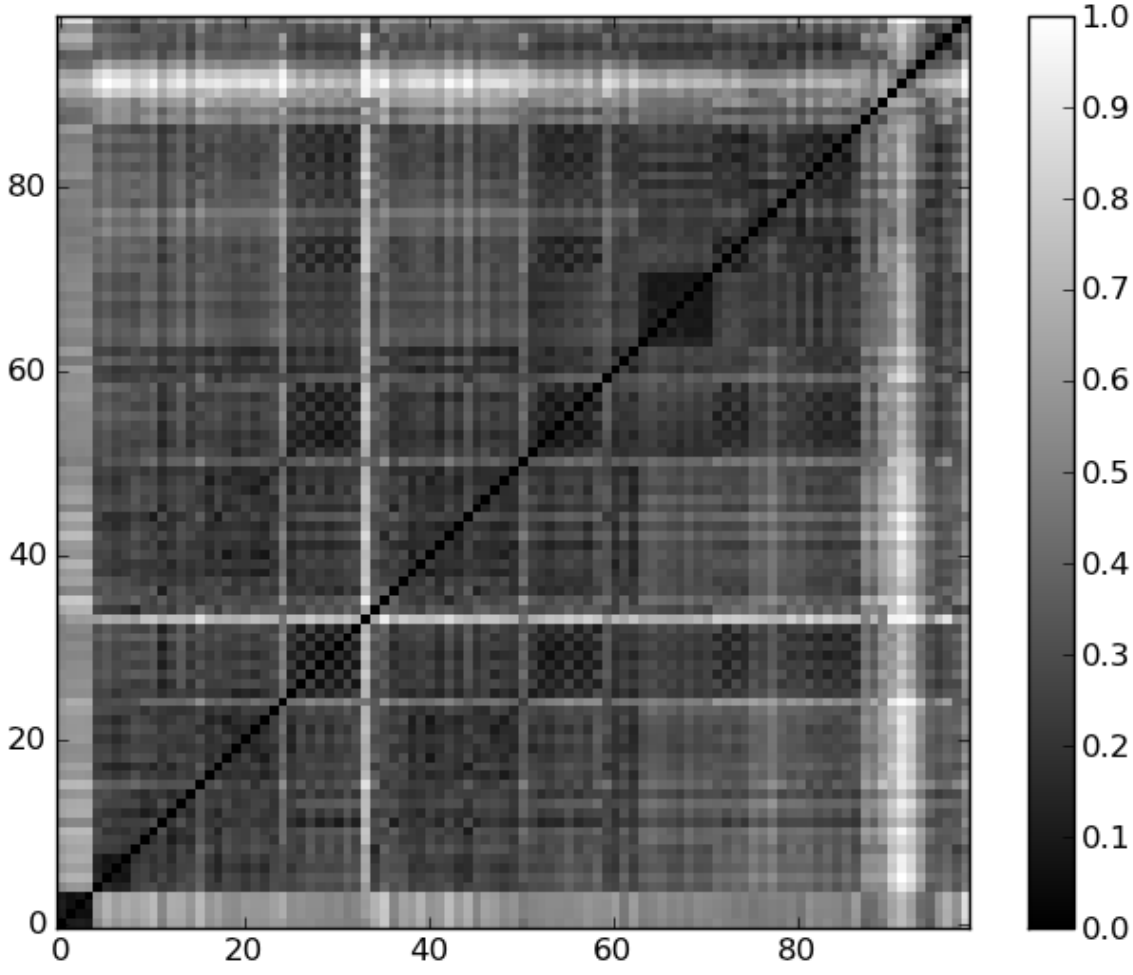
Similarity Matrices

- Represents "similarity" between a set of vectors
- Similarity = distance metric
- Cooper/Foote used for summarization
- Shiu suggests per-measure
- "Change" = dissimilar segments
- Useful for visualization

MFCC Similarity Matrix



MFCC Similarity Matrix



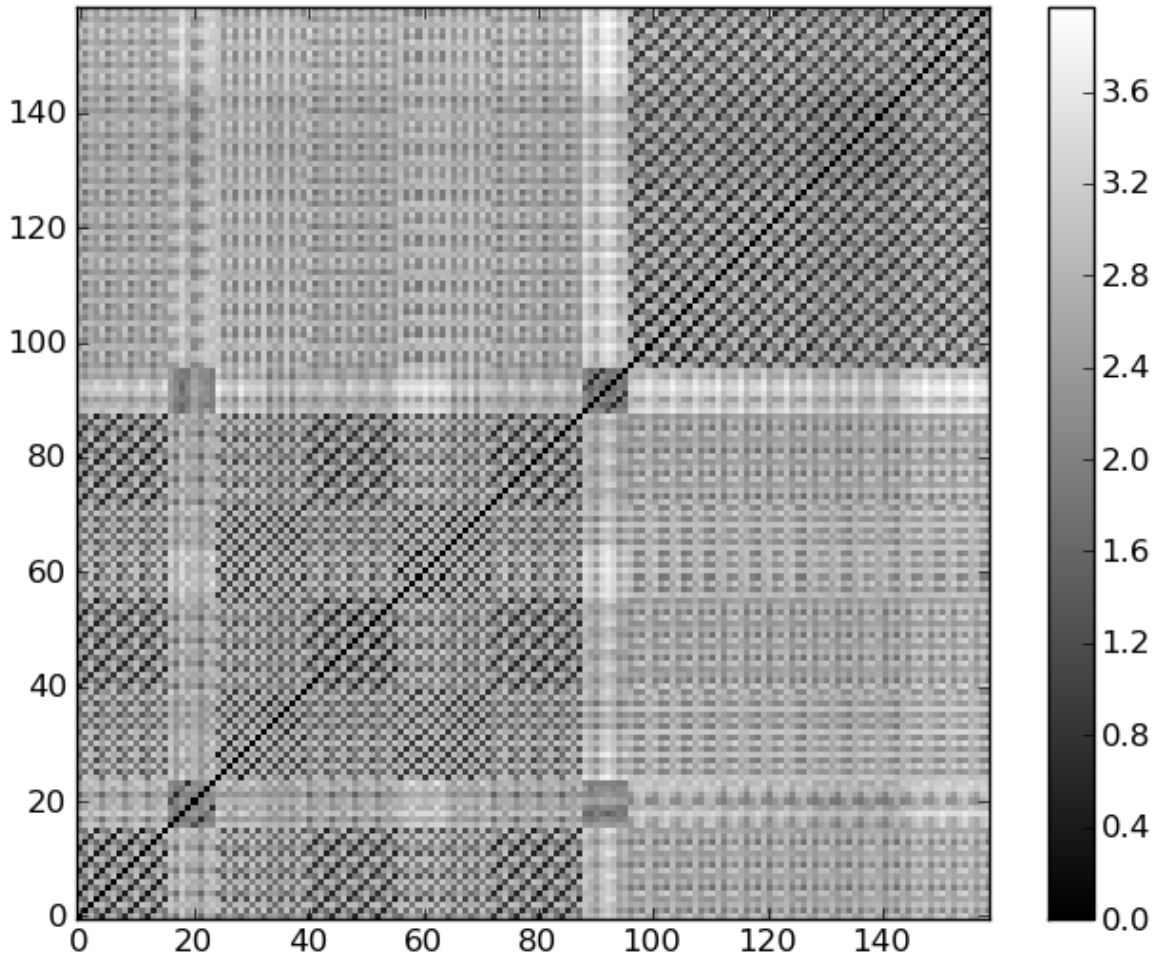
Matrix $\rightarrow$ Changes?

- Visual segmentation is not hard, but is highly subjective
- Possible algorithms:
  - Off-diagonal
  - Cooper/Foote "Novelty Kernel"
  - Square edge change
  - Dynamic programming
- Try combination of approaches

Melody Changes

- Changes in note sequences
- Notes: PCP
- Sequence: Concatenate beat vectors
- "Chord progression" vectors
- Can also use similarity matrices

Melody Similarity Matrix



Melody Changes

- Melody patterns vary greatly
- Shiu:
  - For short-scale, use 2D expected windows
  - For long-scale, use Viterbi algorithm
- For changes, look at diagonal ends
- Heavily relies on (down)beat and meter detection accuracy

Summary

- Tempo
- Beat
- Downbeat
- Dynamics
- Instrumentation (timbre)
- Melody

XML Description Format

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<song>
  <metadata>
    <scale confidence="0.676">major</scale>
    <feel confidence="0.412">soft,slow,sad</feel>
    <genre confidence="0.349">jazz,classical,ambient</genre>
    <tempo confidence="0.912">117.283</tempo>
  </metadata>
  <timeline>
    <beat confidence="0.865" importance="1.000" time="0.281" type="measure"/>
    <beat confidence="0.930" importance="0.250" time="1.216" type="beat"/>
    <beat confidence="0.985" importance="0.500" time="2.104" type="beat"/>
    <change confidence="0.623" importance="0.377" time="2.180" type="key"/>
    <beat confidence="0.997" importance="0.250" time="3.377" type="beat"/>
    <change confidence="0.781" importance="0.421" time="3.600" type="section"/>
    <beat confidence="0.967" importance="1.000" time="4.281" type="measure"/>
    <beat confidence="0.924" importance="0.250" time="5.389" type="beat"/>
    <beat confidence="0.985" importance="0.500" time="6.177" type="beat"/>
    <beat confidence="0.979" importance="0.250" time="7.433" type="beat"/>
  </timeline>
</song>
```

Thanks!

craffel@gmail.com
http://www.colinraffel.com