
Identifying “Cover Songs” with Beat-Synchronous Chroma Features

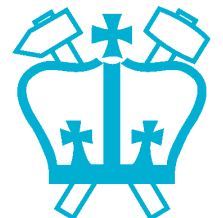
Dan Ellis and Graham Poliner

Laboratory for Recognition and Organization of Speech and Audio
Dept. Electrical Eng., Columbia Univ., NY USA

{dpwe, graham}@ee.columbia.edu

<http://labrosa.ee.columbia.edu/>

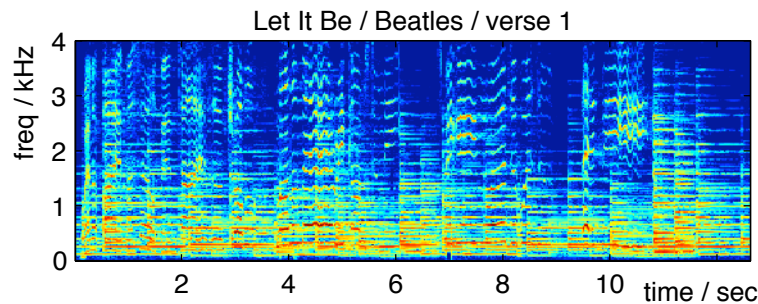
1. Cover Songs
2. Chroma Features
3. Beat Tracking
4. Matching Cover Songs



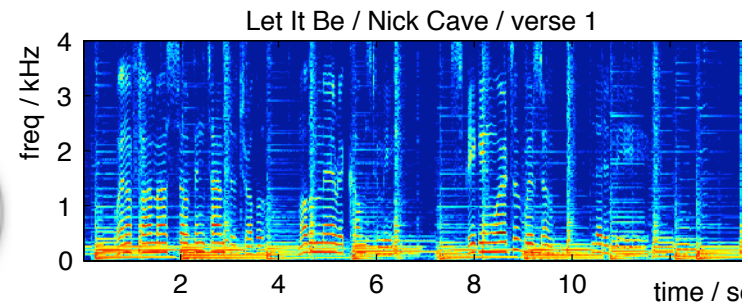
Cover Songs

- “Cover Songs” = **reinterpretation** of a piece
 - different instrumentation, character
 - no match with “timbral” features

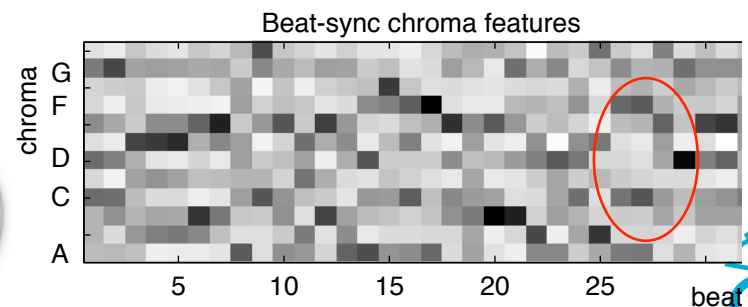
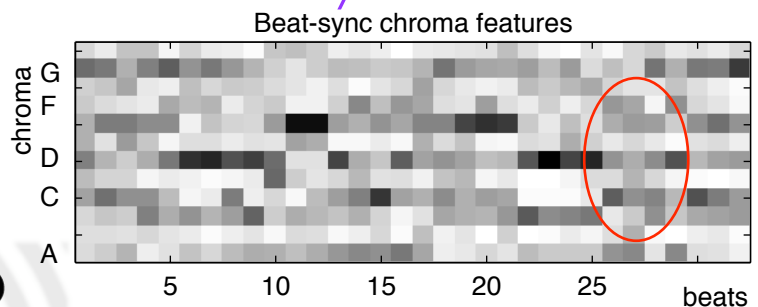
Let It Be - The Beatles



Let It Be - Nick Cave



- **Need a different representation!**
 - beat-synchronous chroma features



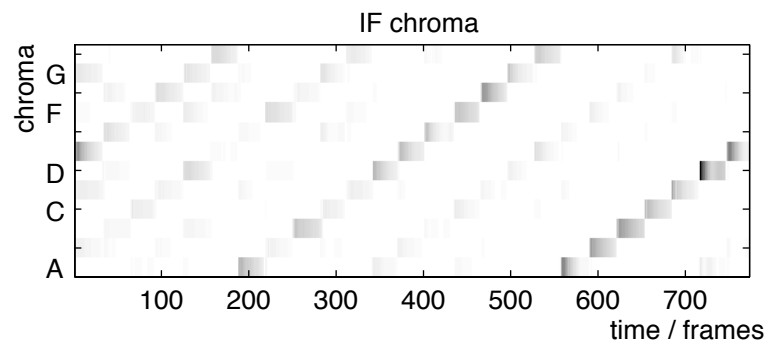
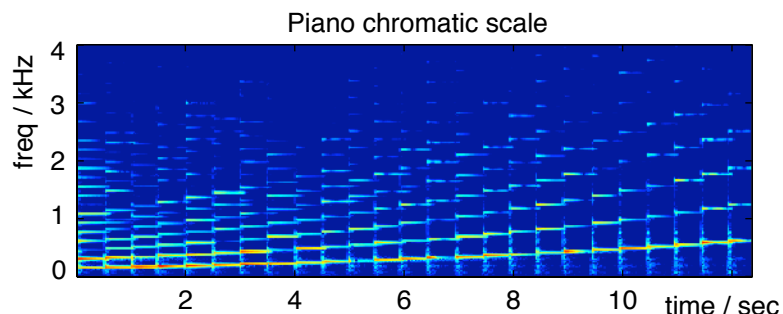
Identifying Cover Songs - Ellis & Poliner

2007-04-20 - 2/16

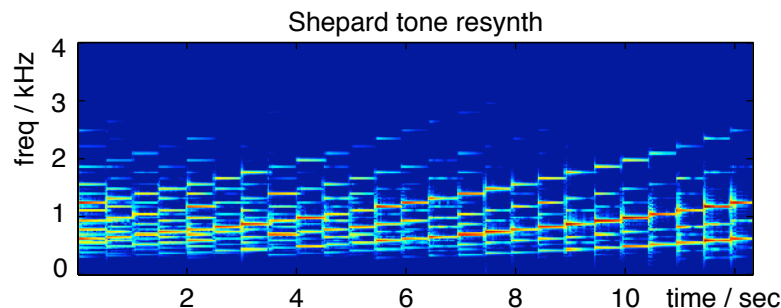
Chroma Features

- Chroma features map spectral energy into one **canonical octave**
 - i.e. 12 semitone bins

Piano scale

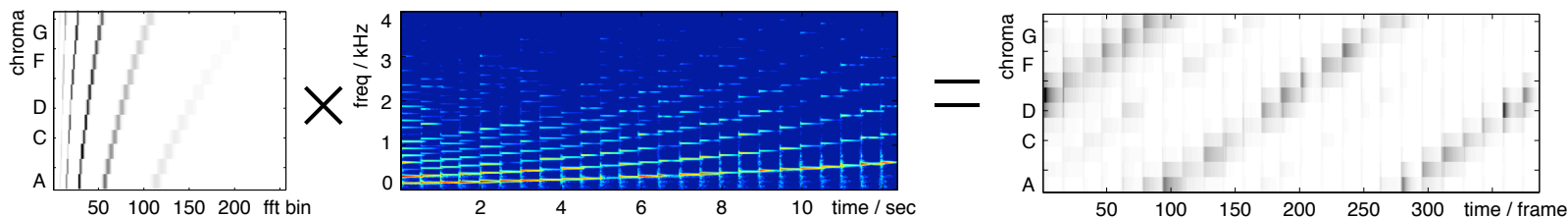


- Can resynthesize as **“Shepard Tones”**
 - all octaves at once

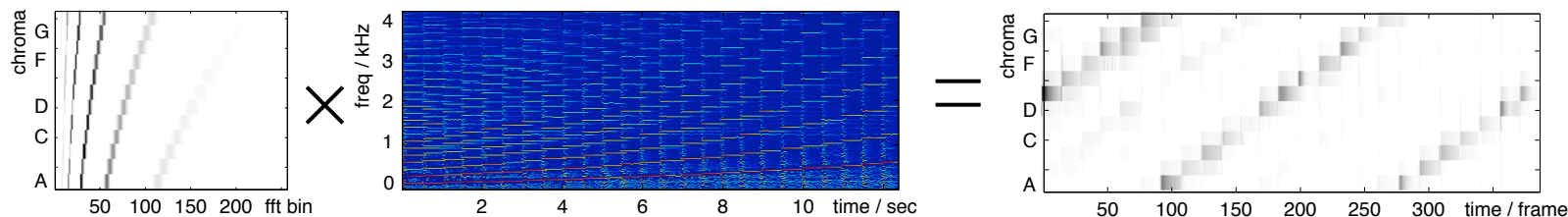


Calculating Chroma Features

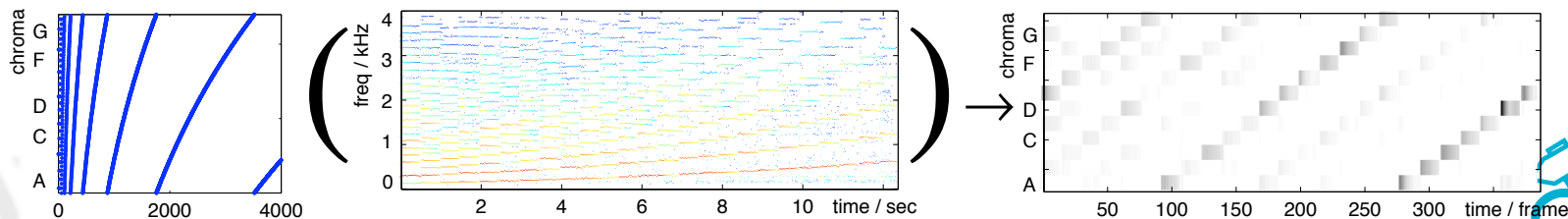
- **Method 1: Map every STFT bin**
 - blurs non-tonal energy



- **Method 2: Map only STFT peaks**
 - still blurry at low frequencies

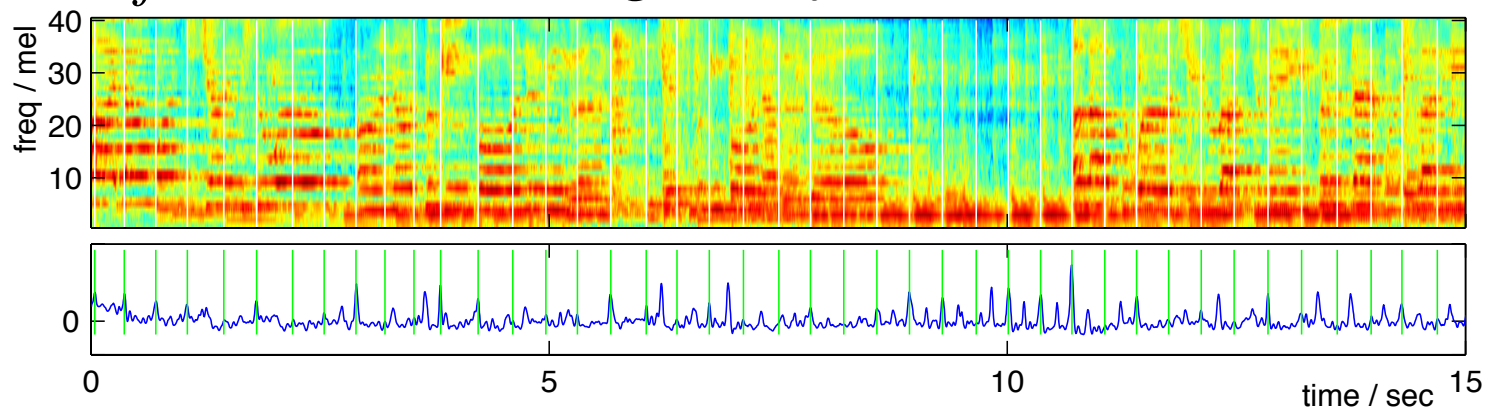


- **Method 3: Instantaneous Frequency $\delta\theta/\delta t$**
 - escapes frequency resolution limit

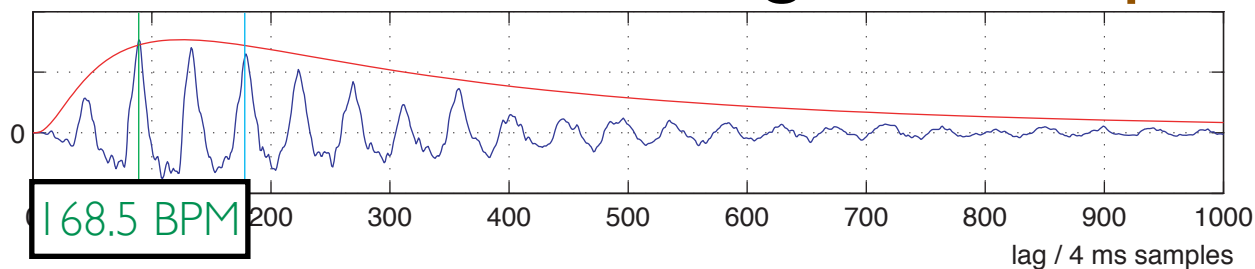


Beat Tracking (I)

- Goal: One feature vector per ‘beat’ (tatum)
 - for tempo normalization, efficiency
- “Onset Strength Envelope”
 - $\text{sum}_f(\max(0, \text{diff}_t(\log |X(t, f)|)))$

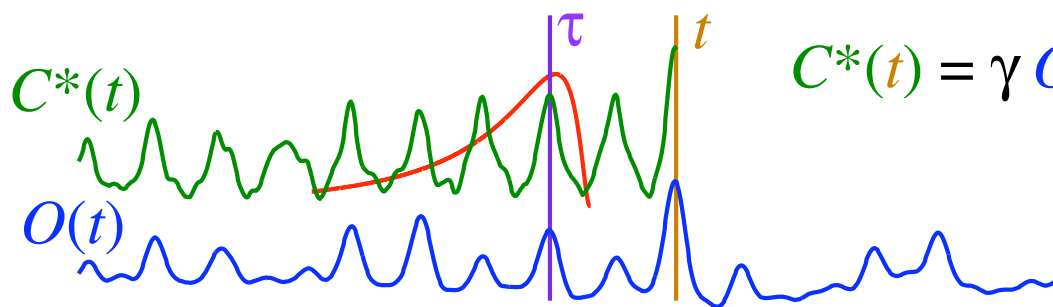


- Autocorr. + window \rightarrow global tempo estimate



Beat Tracking (2)

- **Dynamic Programming** finds beat times $\{t_i\}$
 - optimizes $\sum_i O(t_i) + \alpha \sum_i W((t_{i+1} - t_i - \tau_p)/\beta)$
 - where $O(t)$ is onset strength envelope (local score)
 $W(t)$ is a log-Gaussian window (transition cost)
 τ_p is the **default beat period** per measured tempo
 - incrementally find best predecessor at every time
 - **backtrace** from largest final score to get beats

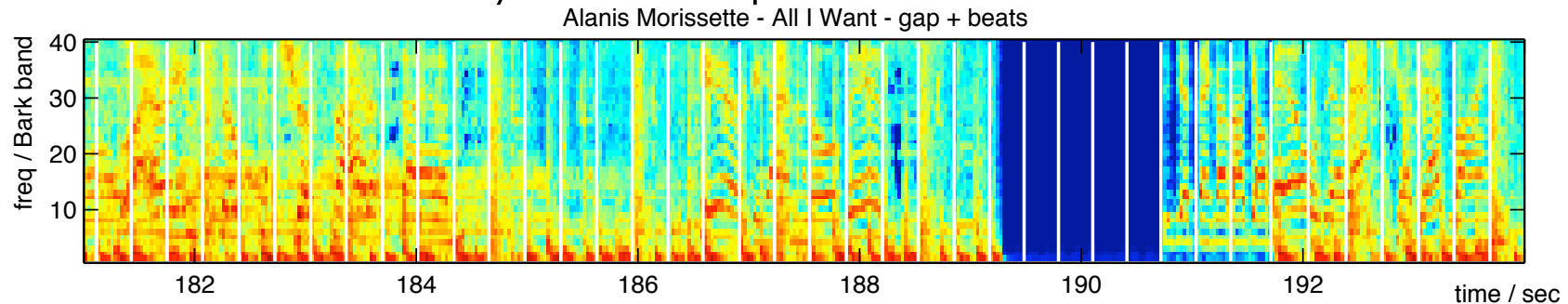


$$C^*(t) = \gamma O(t) + (1-\gamma) \max_{\tau} \{ W((\tau - \tau_p)/\beta) C^*(\tau) \}$$

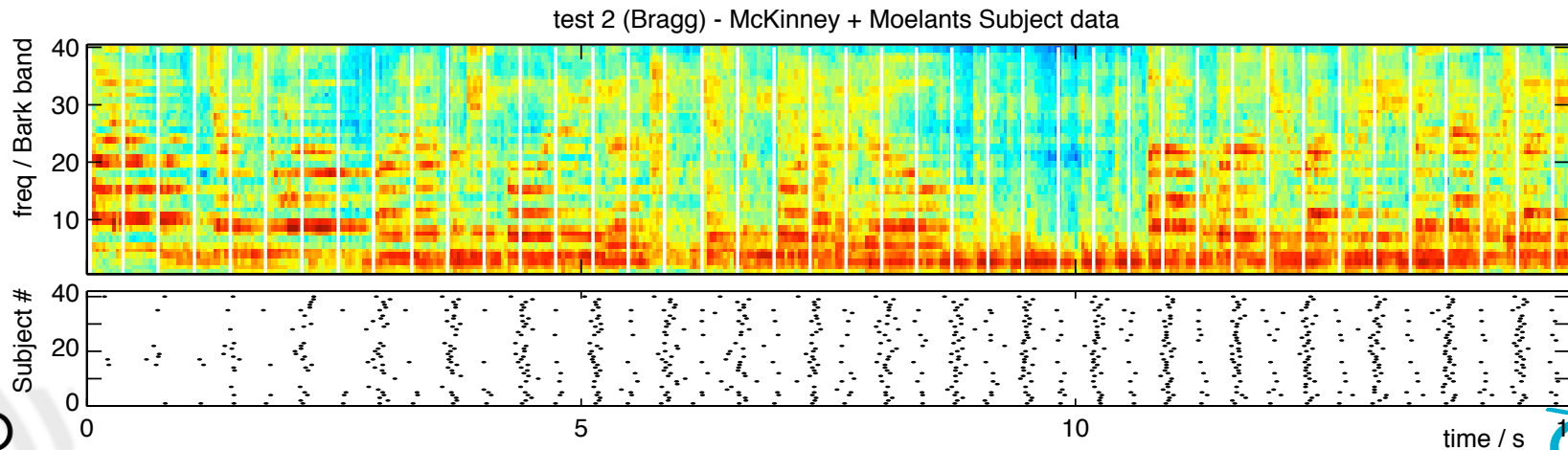
$$P(t) = \operatorname{argmax}_{\tau} \{ W((\tau - \tau_p)/\beta) C^*(\tau) \}$$

Beat Tracking Results

- DP will **bridge gaps** (non-causal)
 - there is always a best path ...

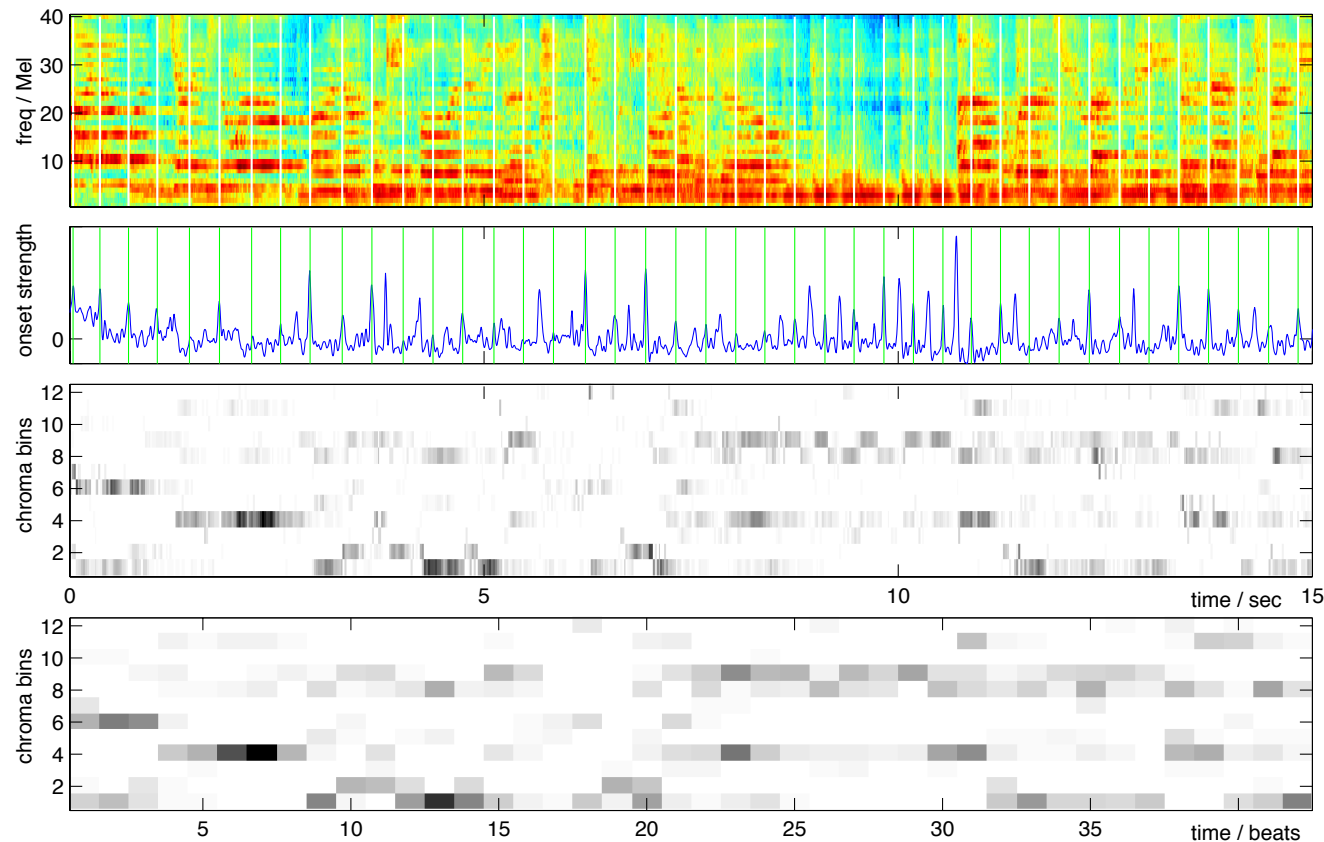


- 2nd place in MIREX 2006 Beat Tracking
 - compared to McKinney & Moelants human data



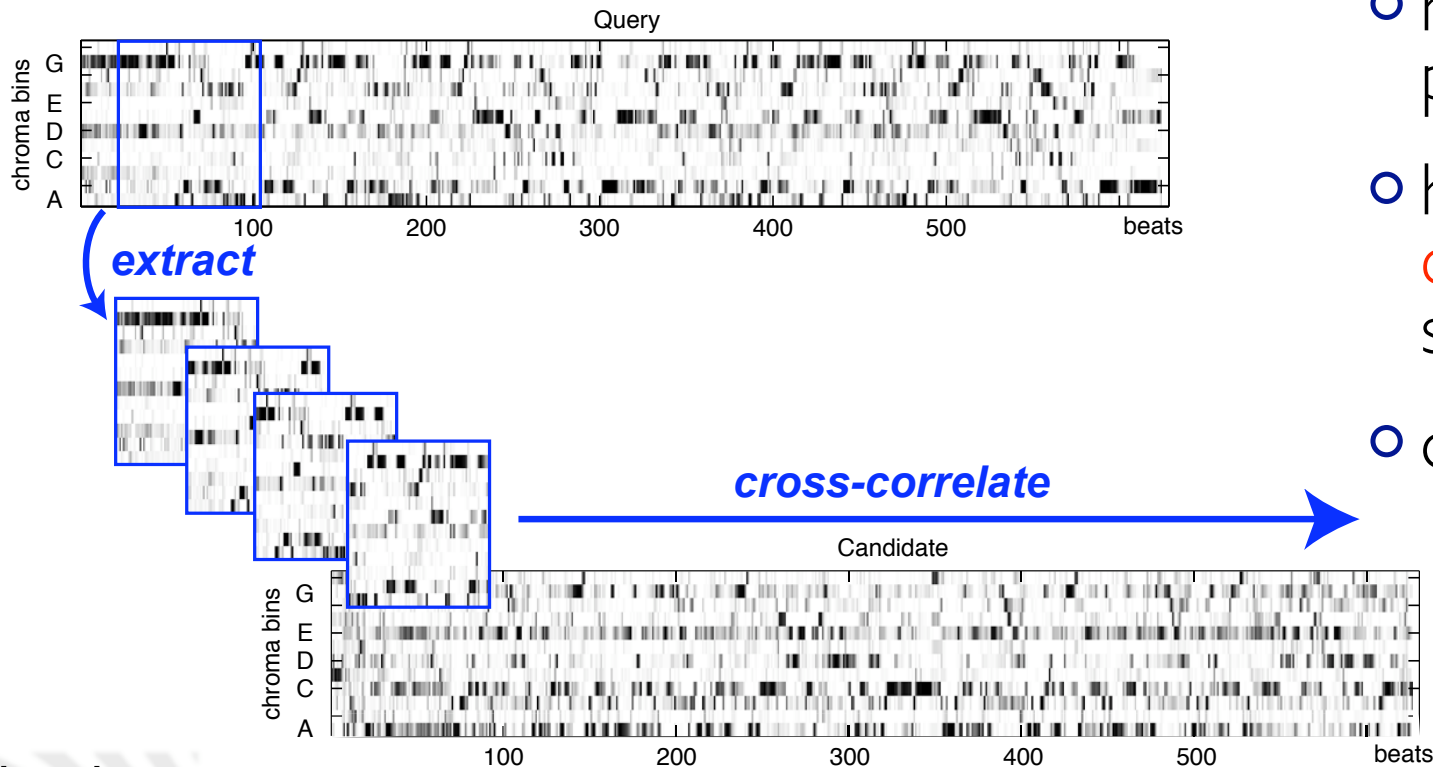
Beat-Synchronous Chroma Features

- **Beat + chroma features** / 30ms frames
 - **average chroma** within each beat
- compact; sufficient?



Matching (I): Little Fragments

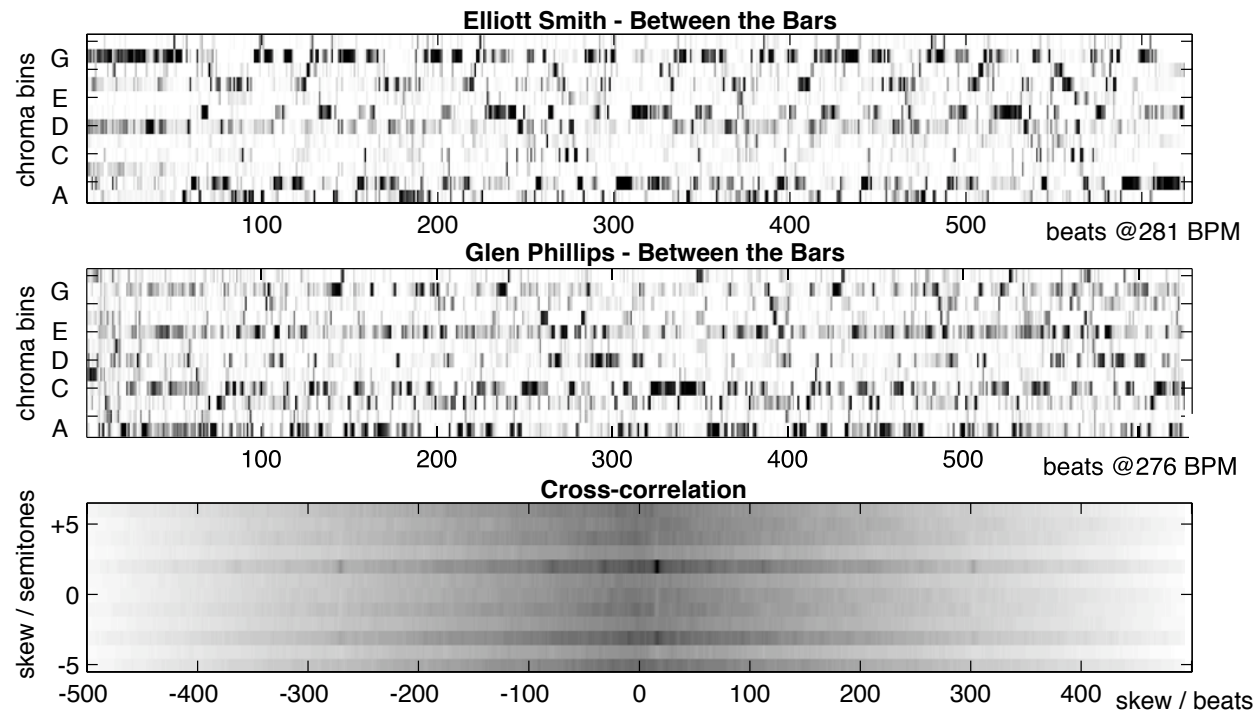
- Cover versions may change song **structure**
 - multiple local matches at different alignments
- Match query and target as **many small pieces?**



- how **big** are the pieces?
- how do we **combine** individual scores?
- do we have **all day?**

Matching (2): Global Correlation

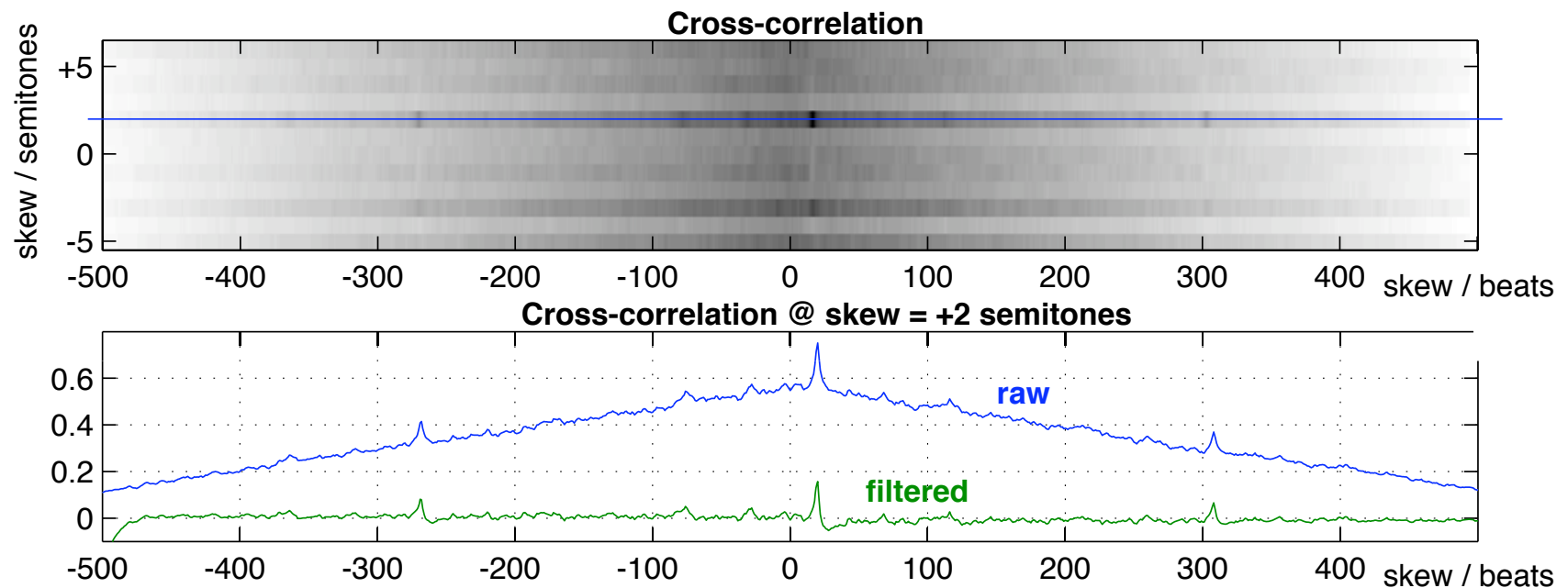
- Cross-correlate *entire* beat-chroma matrices
 - ... at all possible transpositions
 - implicit combination of match quality and duration



- One good matching fragment is sufficient...?

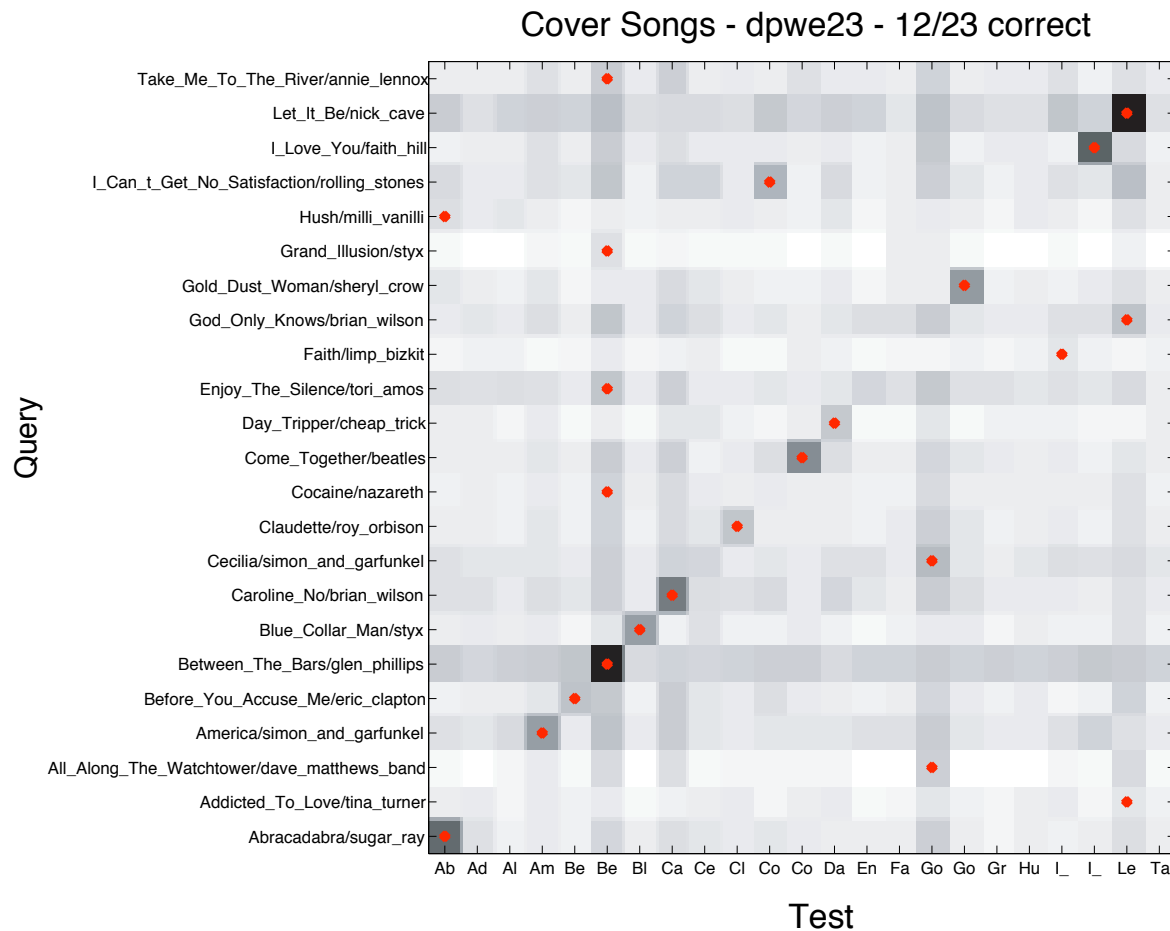
Filtered Cross-Correlation

- Raw correlation not as important as precise **local match**
 - looking for large **contrast** at ± 1 beat skew
 - i.e. **high-pass filter**



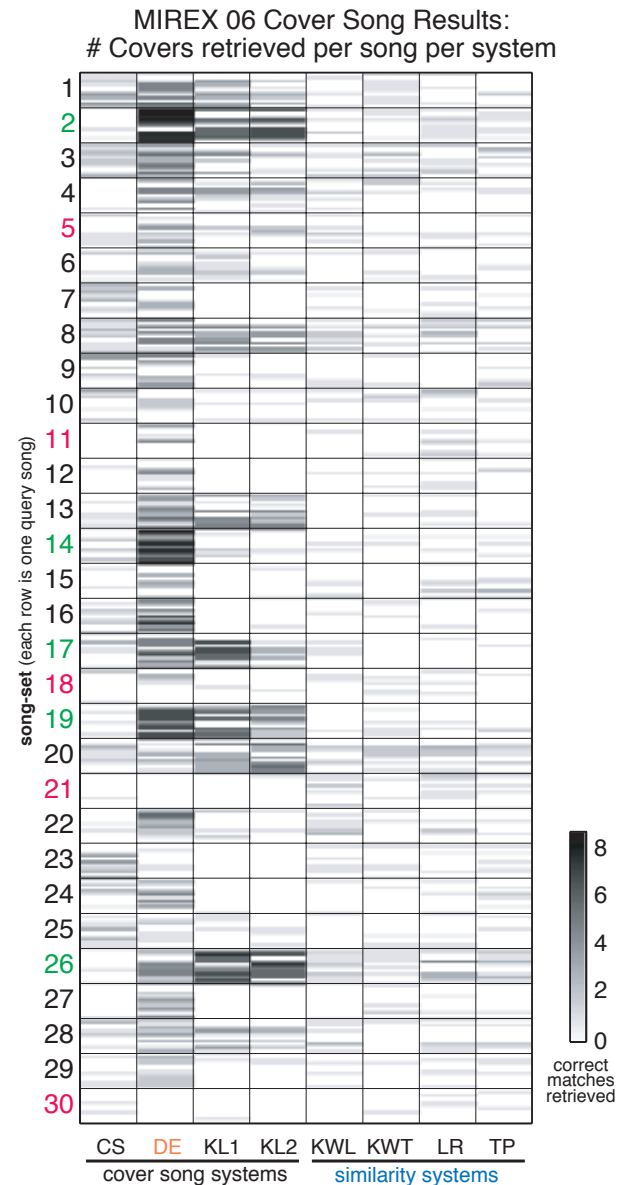
Results (I): Ellis 23 set

- 23 pairs of cover songs from uspop2002 +...
 - one correct match per query



Results (2): MIREX 06

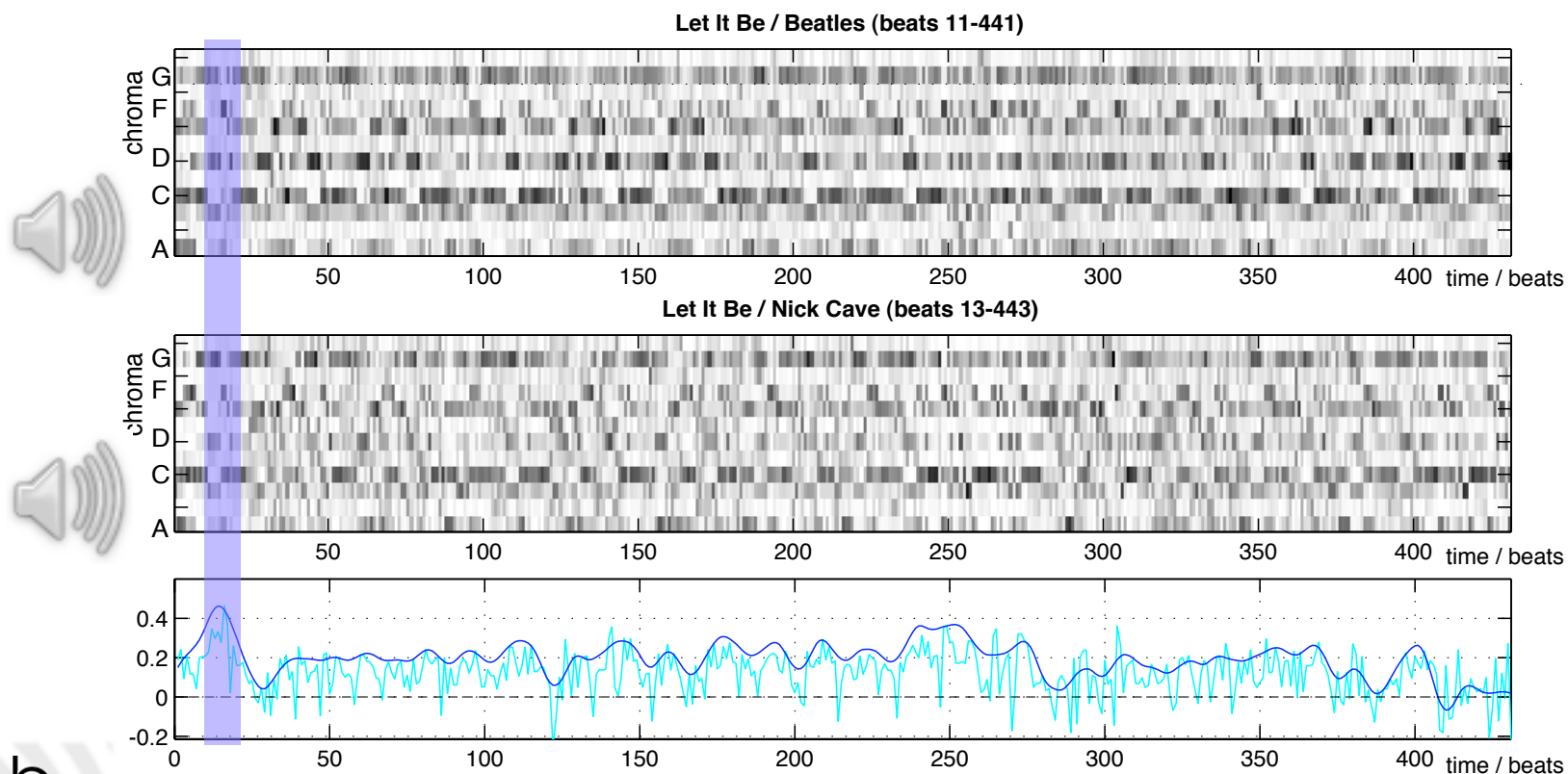
- **Cover song contest**
 - 30 songs x 11 versions of each (!)
 - (data has not been disclosed)
 - # true covers in top 10
 - 8 systems compared (4 cover song + 4 similarity)
- **Found 761/3300**
= **23% recall**
 - next best: 11%
guess: 3%



Where are the matches?

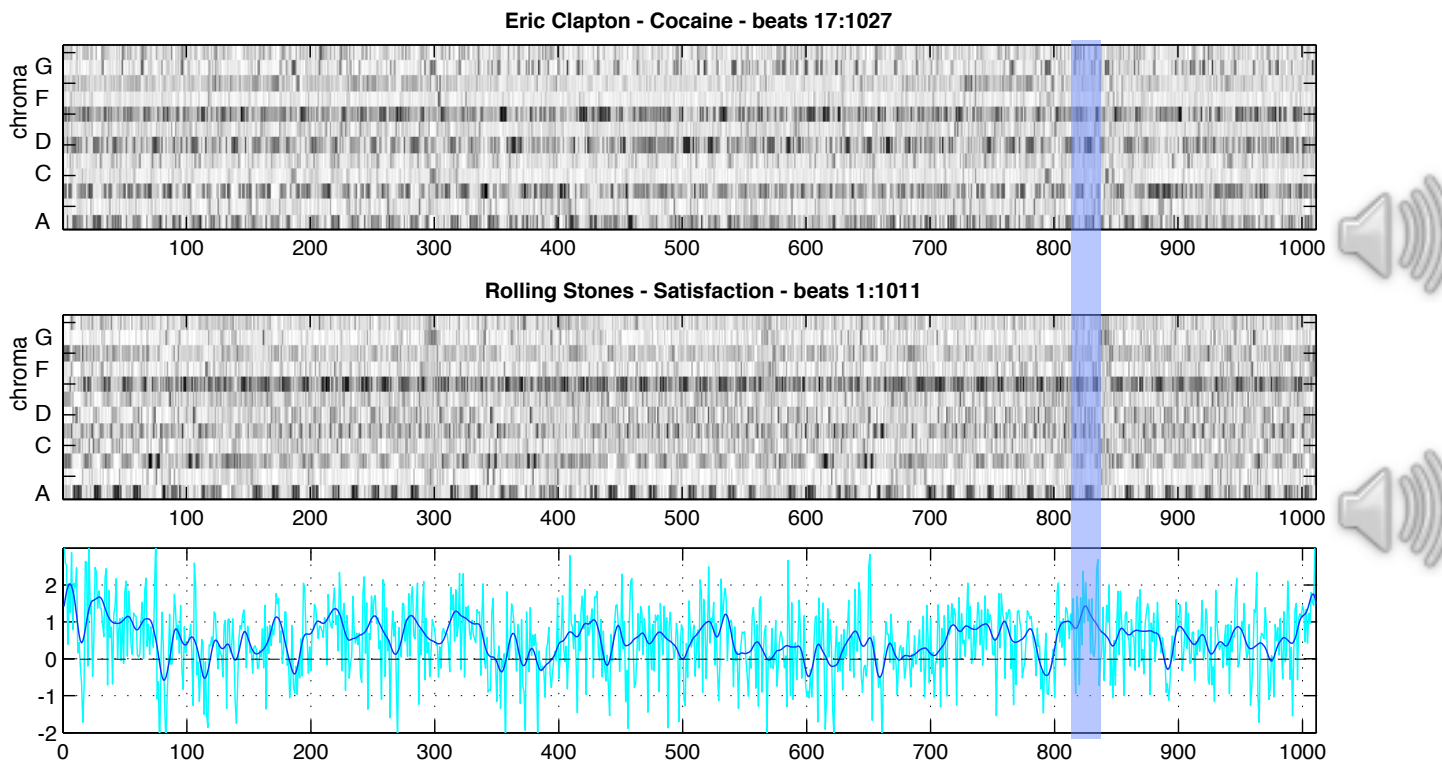
- **Look inside** global cross-correlation to find matching fragments...

- $\text{xcorr} = \sum_t \sum_f (C_1(t, f) \cdot C_2(t, f))$ - view along **time**



What are the mistakes?

- False reject - missed true match
 - cover version is too different, beat tracking wrong ...
- False alarm - invalid match
 - “Cocaine” (Clapton) vs. “Satisfaction” (Stones)



Conclusions and Future Work

- **Beat-synchronous chroma features**
are successful for matching cover songs
 - captures **melody-harmony**, not instruments
- **Further uses:**
Beat-chroma fragments
as **musical building blocks**
 - e.g. VQ over large body of music
 - find recurrent **motifs**
 - artist identification?
- **Code available!** Google “matlab cover song”

