Multiscale Signatures of Beatles Songs in a Cover Song Task

Katherine M. Kinnaird
Department of Mathematics, Dartmouth College

Abstract

While there are many Music Information Retrieval (MIR) classification tasks, this work focuses on the cover song task. Given a particular piece of music, the goal of the cover song task is to find all the different recordings of the given song by a variety of artists in a set of music. For this work, our data set is a collection of songs by the Beatles. Our approach begins by building a novel multiscale signature for each song that captures repetitive structure at several scales, while also being of a manageable dimension. We then apply a metric that is appropriate for the cover song task to this representation space of song signatures, allowing for fine-tuned comparison between songs. This multiscale approach differs from those in the literature that largely consider single-scale, single-feature representations.

Preliminaries

The Beatles Data Set

A *song* is represented by a matrix of pairwise squared-Euclidean distances between the song's audio shingles. These *audio shingles* are comprised of 30 concatenated feature vectors, created by splitting the audio track into tenth of a second windows and extracting the mel-frequency cepstral coefficients (MFCCs) for each window.[1,2,3]

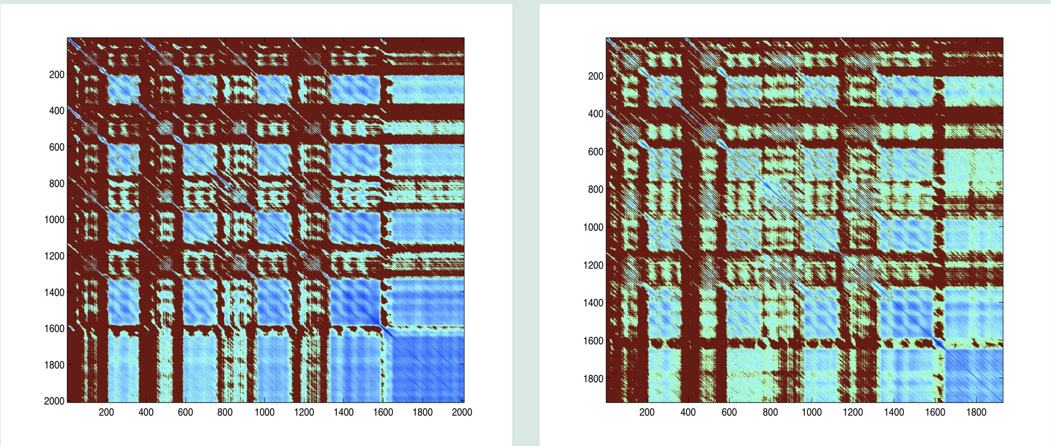

Figure 1: "Hello, Goodbye" on *1967-1970 (D1)* and *Anthology 2 (D2)*

We *clean* these matrices, by separating recorded zeros from non-recorded distances by setting the non-recorded distances to 3. The songs in this work are from the Beatles data set, created by Prof. Michael Casey, Department of Music, Dartmouth College.

Cover Song Task

Beginning with a set of music and a target song, we seek to find all of the performances of the target song by a variety of artists. In this work, we only have one artist: The Beatles. Therefore if the target song is the recording of "Hello, Goodbye" on one album, say *1967-1970 (D1)*, then we want to find all performances of "Hello, Goodbye" on other albums, such as *1* and *Anthology 2 (D2)*.

Our Approach

1. Create multiscale signatures for each song that encode relevant, size-appropriate information
2. Compute similarity scores for pairs of songs based on multiscale signatures
3. Create directed network based on nearest neighbors

Resulting Network

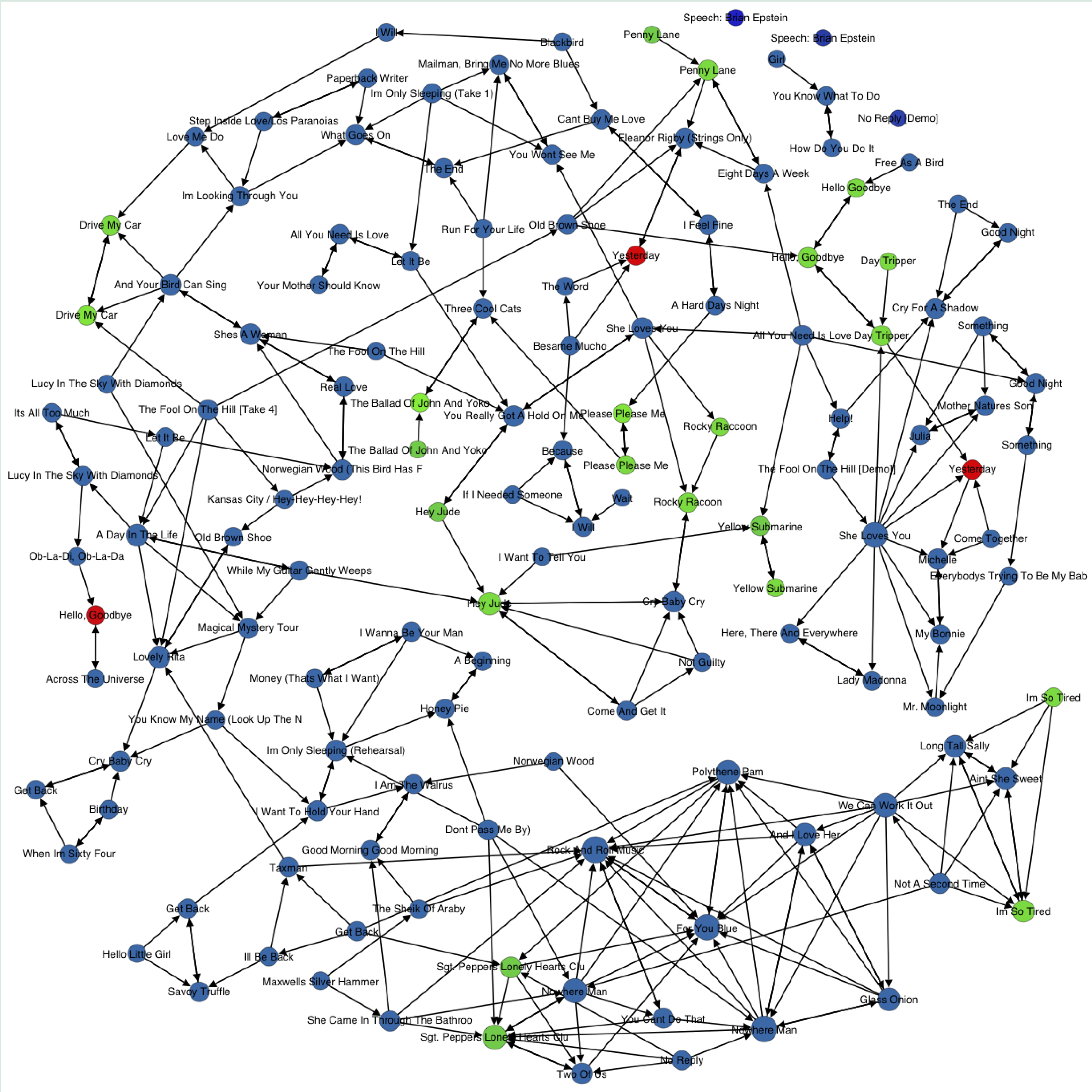

Figure 2: Resulting Network with parameters set: `bandwidth` = [50:10:150], $\varepsilon = 0.1$, $\mathcal{P} = 0.1$, $\vartheta = 0.6$

Song Examples

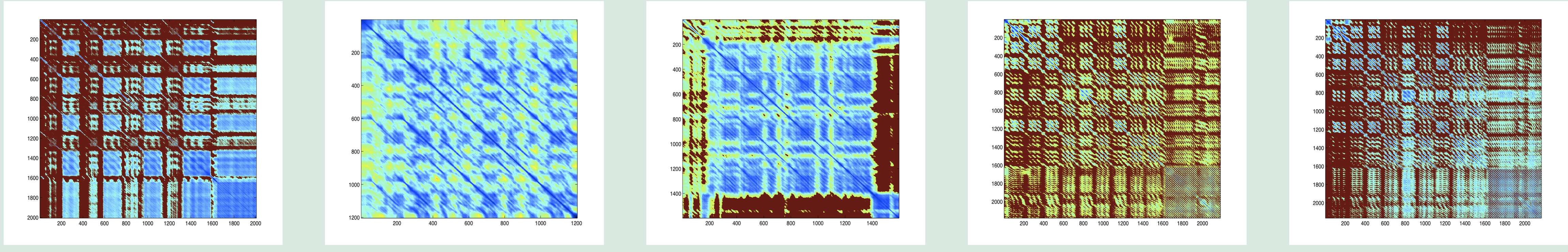

Figure 3: (a) "Hello, Goodbye" on *1*, (b,c) "Yesterday" on *1* and *Anthology 2 (D1)*, (d,e) "All You Need Is Love" on *Yellow Submarine* and *1967-1970 (D1)*

Multiscale Signatures for Songs

We represent each song with a *multiscale signature*. These multiscale signatures are a concatenation of independently created *single-scale signatures*.

Thresholding Diagonals

Since diagonals of (near) zeros denote repetitions,[4,5,6] we compute and threshold the squared-Euclidean norm for diagonals of square-sub-matrices of the fixed scale of size $K \in$ `bandwidth` (relevant scale sizes). The threshold is unique for each K and is dependent on ε (square-norm diagonal tolerance per unit) and $\mathcal{P}$ (percent of maximum value distances allowed).

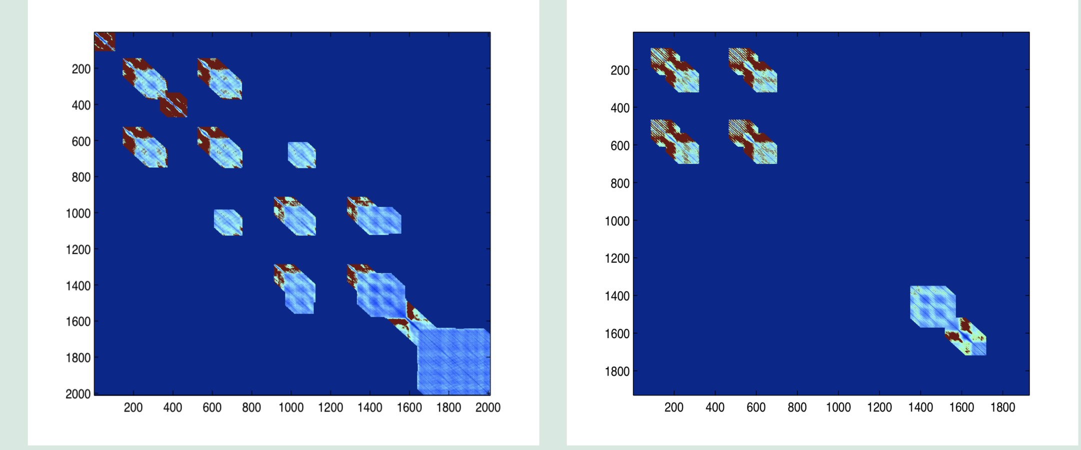

Figure 4: "Hello, Goodbye" on *1967-1970 (D1)* and *Anthology 2 (D2)* with $K = 100$, $\varepsilon = 0.1$, and $\mathcal{P} = 0.1$.

Pattern Vector

The resulting thresholded matrix is then used to create the *song pattern vector (SPV)* which encodes the repeat identification numbers (repeat IDs) in the order that the repeats of size K occur. Creating the song pattern vectors relies on ϑ (repeat overlap parameter). We also create a *clean song pattern vector (CPV)*, which removes all repeat identification numbers that only appear once in SPV. For the above two recordings of "Hello, Goodbye" and $\vartheta = 0.6$, we have the following song patterns:

1967-1970 (D1): $[1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]$
$[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$

Anthology 2 (D2): $[1, 1, 1, 1, 2, 2, 2, 2]$

Comparing Multiscale Signatures

We compare two songs by comparing their SPVs and CPVs. We assign a similarity score based on 1) the number of unique repeat IDs, 2) the number of times repeat IDs appear, and 3) the order of repeat IDs.

Working with SPVs and allowing each song to be the target song, we compute the similarity scores of the target song compared to the other songs. Those with the highest similarity score are identified as the target song's nearest neighbors. Repeating the process for the CPVs, we combine the two results to get the final network.

References

[1] M. Casey, C. Rhodes, and M. Slaney, *Analysis of minimum distances in high-dimensional musical spaces*, IEEE Transactions on Audio, Speech, and Language Processing **16** (2008), no. 5, 1015 – 1028.
[2] M. Casey and M. Slaney, *Song intersection by approximate nearest neighbor search*, Proceedings of ISMIR 2006, 144–149.
[3] M. Casey and M. Slaney, *Fast recognition of remixed audio*, 2007 IEEE International Conference on Audio, Speech and Signal Processing (May 2007), IV–1425 – IV–1428.
[4] M. Cooper and J. Foote, *Summarizing popular music via structural similarity analysis*, 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (Oct. 2003), 127–130.
[5] M. Müller, P, Grosche, and N. Jiang, *A Segment-based fitness measure for capturing repetitive structures of music recordings*, ISMIR 2011, Miami, Oct, 2011.
[6] J. Paulus, M. Müller, and A. Klapuri, *Audio-based music structure analysis*, ISMIR 2010, 625 – 636.

Project NExT/Young Mathematician's Network Poster Session at JMM 2013, San Diego