Spectral Clustering for one mic Audio Blind Separation

MarC Vinyes

Columbia University

December 18, 2006

Problem

Audio Blind Separation:

- Original mixed audio *out* $\longrightarrow$ Audio signals s_i
- Restrictions s_i:
  1. $\sum_i^n s_i$ perceived similarly to *out*
  2. s_i $i = 1..n$ should mean something to a human (examples: tracks, instruments, auditory streams, physical sources, notes, chords, noises...)

Extraction of the audio signals
Time Frequency Masking

1. Signal splitted into overlapped frames of fixed size in time.
2. FFT
3. Binary mask applied

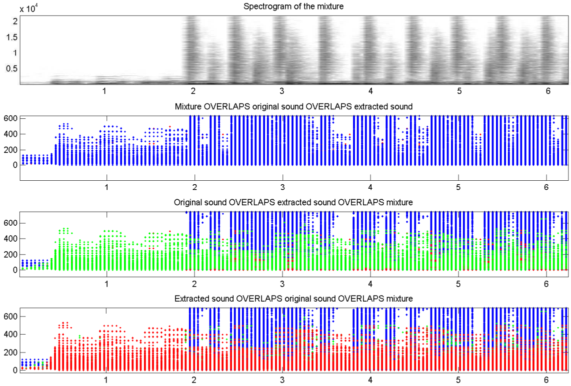
0.3+0.1j	0.2-j	0.2+0.2j	...	
↓	↓	↓		
1	0	1	...	Binary Mask
↓	↓	↓		
0.3+0.1j	0	0.2+0.2j	...	

4. IFFT
5. Overlap-and-add process.

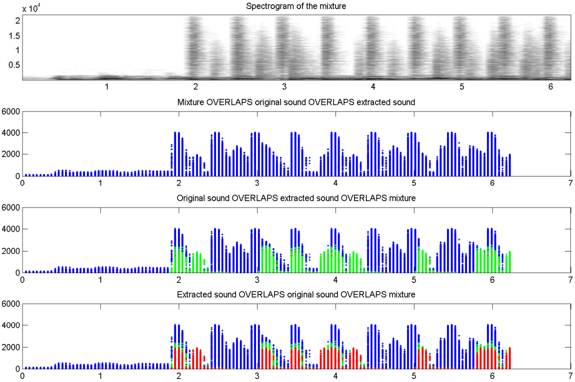
Data

- Mixture and sound track waveforms available.
  'mix.wav' = 'guitar.wav' + 'kick.wav' + 'snare.wav' + 'hh.wav'
- We know that it's possible to extract each of them.
  We know how to generate ideal binary masks if the target sound is available.

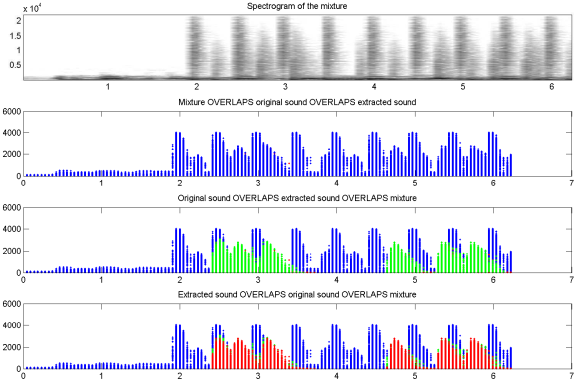
Example: ideal binary mask to extract 'guitar.wav'



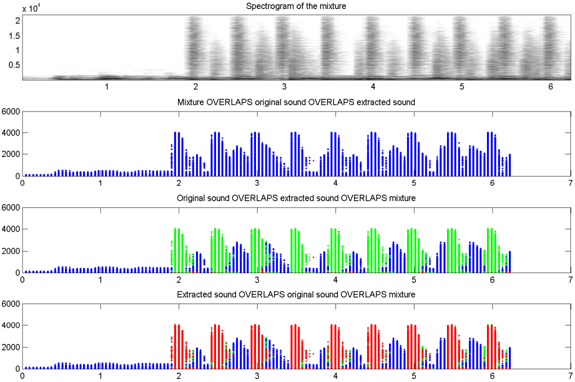
Example: ideal binary mask to extract 'kick.wav'



Example: ideal binary mask to extract 'snare.wav'



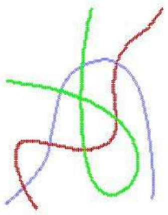
Example: ideal binary mask to extract 'hh.wav'



Machine learning to cluster the time-frequency points

Learning the binary mask...

- Clusters are not disjoint. We focus on extracting one single audio signal each time.
- SVM or Spectral Clustering? Spectral Clustering seem to be more appropiate when there are intersections.



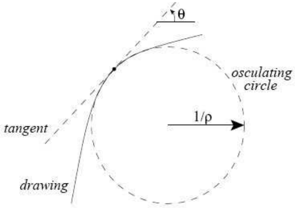


Figure: Labeled hand drawings by spectral clustering. Francis R.Bach, Michael I.Jordan 06.

Spectral Clustering

- Let $A = (A_r)_r \in 1 \cdots R$ be the R disjoint clusters of the points such that $\bigcup_r A_r = \{p_1, p_2, \cdots p_N\} = V$ which the algorithm should output.
  Let $W(A, B) = \sum_{i \in A} \sum_{j \in B} W_{ij}$ the total weight between the sets of points A and B.
  Let a similarity matrix W.
  Finally let D be a diagonal matrix whose i-th diagonal element is the sum of the elements in the i-th row of W.
- We want to minimize the R-way normalized cut:

$$C((A_r)_{r \in (1 \cdots R)}, W) = \sum_{r=1}^{R} \frac{W(A_r, V \setminus A_r)}{W(A_r, V)}$$

- Algorithm that solves it by computing the eigenvectors of $D^{-1/2} W D^{-1/2}$ and performing a weighted Kmeans clustering of them.

Spectral Clustering applied to audio

- W is huge! Solutions:
  - Analyze the audio in short frames.
  - Approximate W by a sparse matrix. "low-band rank decomposition" suggested by Francis R.Bach, Michael I.Jordan 06. Numerical methods that take advantage of it to find the eigenvectors of $D^{-1/2}WD^{-1/2}$.
- How we compute the distance between two points?
  - Use features that are related to how we group sounds. "Auditory Scene Analysis" by Bregman.
  - Automatically learn the weight of each feature. Francis R.Bach, Michael I.Jordan 06.

Simulations

Simplified implementation:

- We adapt spectral clustering used for image processing. L. Zelnik-Manor and P. Perona 04.
- We use a sparse W similarity matrix which sets a neighbourhood of 7x7 nonzero time-frequency points.
- We analyse a very limited amount frames.

Poor results:

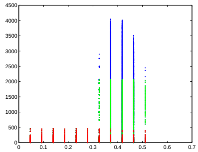


Figure: Output of our algorithm: spectral clustering of the time-frequency points (green). Blue points are the mixture points, and red points are guitar

Conclusion

Bad results but there's still room for improvement:

- More emphasis on finding a good similarity matrix, by intoducing pychoacustic features like pitch, common fate (onset, offset, frequency comodulation).
- Learn automatically their weight to fit the training data.

Main references

- Title: Learning Spectral Clustering, With Application to Speech Separation
  Authors: Francis R.Bach, Michael I.Jordan
  Year: 2006
- Title: Self-Tuning Spectral Clustering
  Authors: L. Zelnik-Manor and P. Perona
  Year: 2004