CS 591 S1 – Computational Audio

Wayne Snyder
Computer Science Department
Boston University

Lecture 15: Onset Detection Continued

Onset detection using spectral energy

Onset Detection

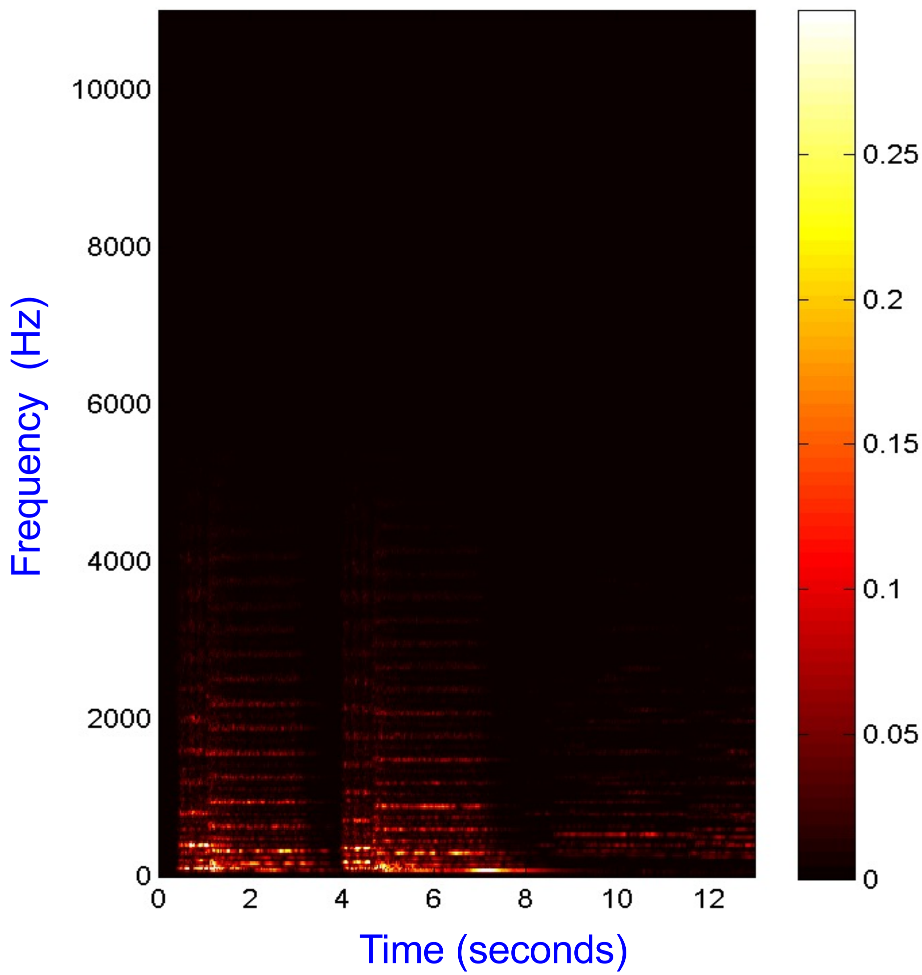
Computer Science

- Energy curves often only work for percussive music
- Many instruments have weak note onsets: wind, strings, voice.
  - Example: Shakuhachi Flute
- Biggest problem: pitch or timbre changes (corresponding to note onset) may not correlate with energy changes, e.g., a singer may change the loudness without changing pitch/note, or change pitch/note without appreciable change in loudness.
- More refined methods needed that capture changes in energy spread over the spectrum [Bello et al., IEEE-TASLP 2005]

Onset Detection (Spectral-Based)

Computer Science

Magnitude spectrogram $|X|$



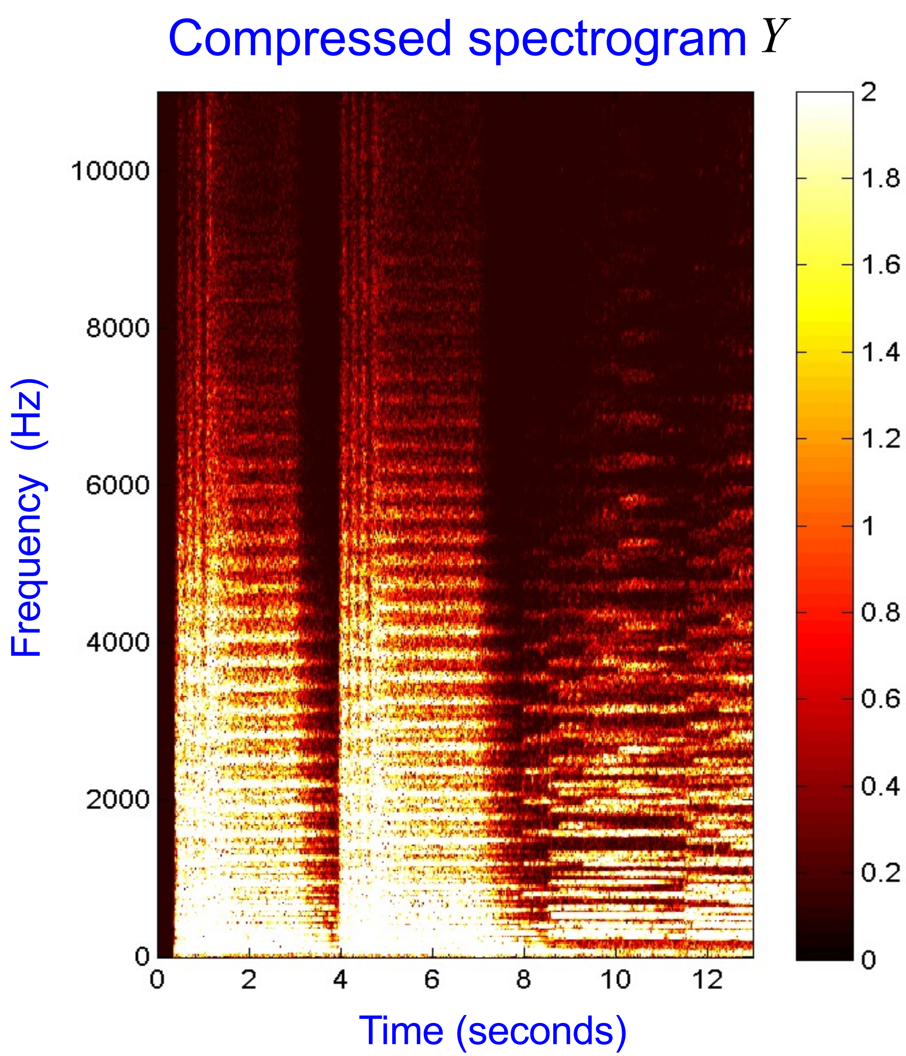
Steps:

1. Spectrogram

- *Aspects concerning pitch, harmony, or timbre are captured by spectrogram*
- *Allows for detecting local energy changes in certain frequency ranges*

Onset Detection (Spectral-Based)

Computer Science



Steps:

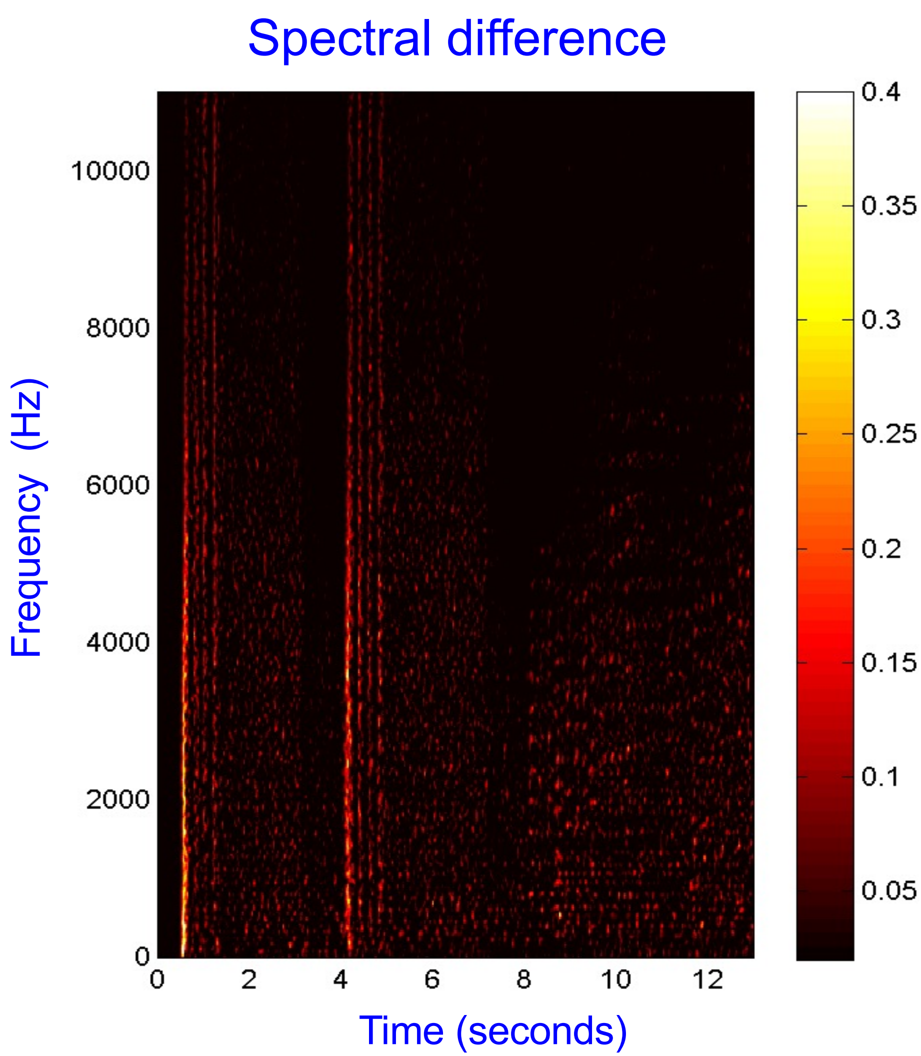
1. Spectrogram
2. Logarithmic compression

$$Y = \log(1 + C \cdot |X|)$$

- *Accounts for the human logarithmic sensation of sound intensity*
- *Dynamic range compression*
- *Enhancement of low-intensity values*
- *Often leading to enhancement of high-frequency spectrum*

Onset Detection (Spectral-Based)

Computer Science



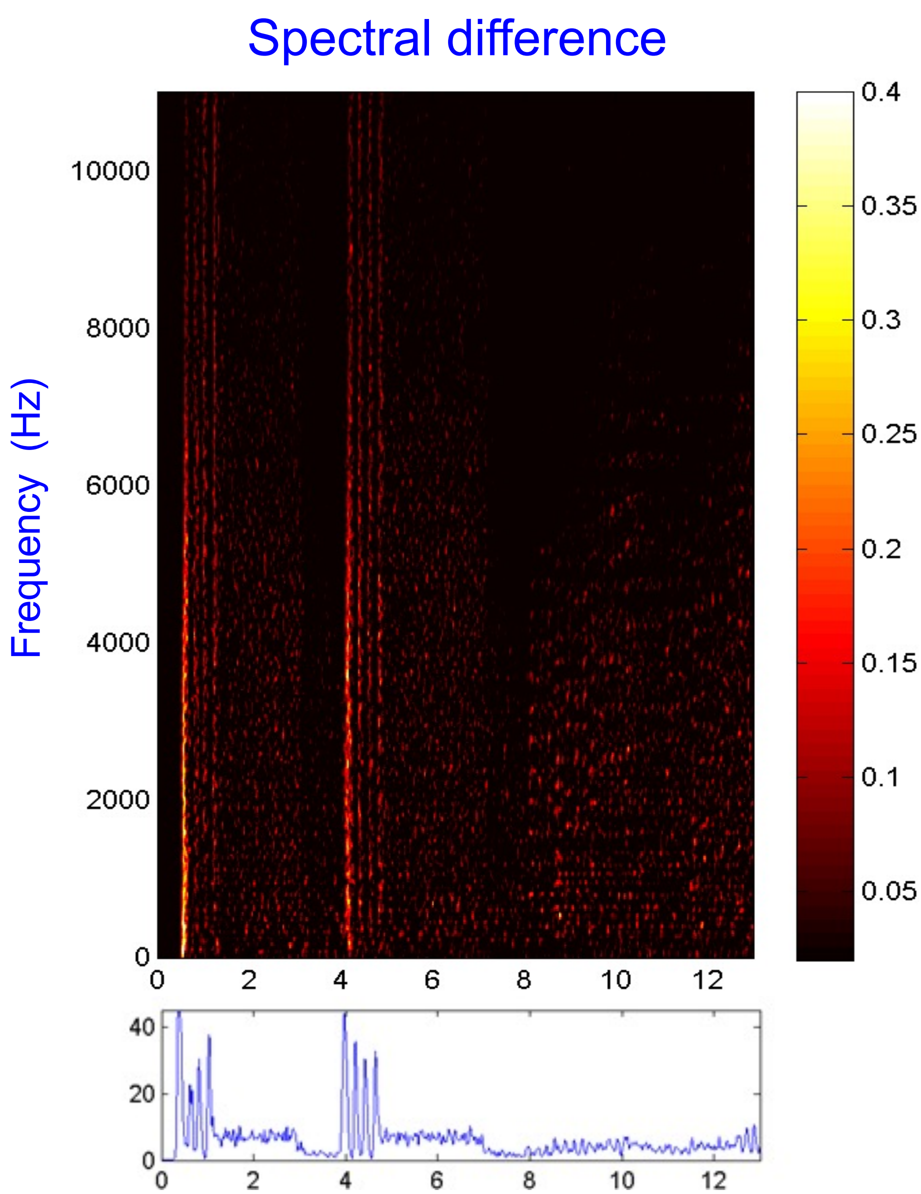
Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation

- *First-order temporal difference*
- *Captures changes of the spectral content*
- *Only positive intensity changes considered*

Onset Detection (Spectral-Based)

Computer Science



Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation
4. Accumulation: spectral differences summarized by a number.

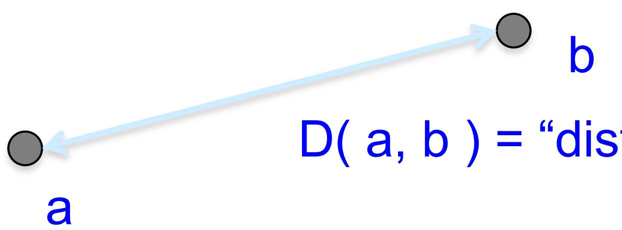
- *Frame-wise accumulation of all positive intensity changes*
- *Encodes changes of the spectral content*

Novelty curve

Digression: Difference/Distance Metrics

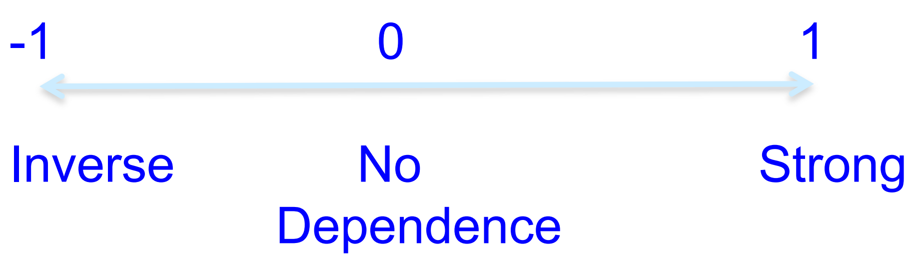
One of the most important issues in analyzing data, especially, multi-dimension and/or time-series data, is understand **how similar two pieces of data are** (represented typically by a vector or multi-dimensional array). There are two principle methods for such comparisons:

Distance Metrics: Similar data vectors are regarded as closer in a geometrical sense; the range is [0 .. ∞), where distance = 0 means the vectors are identical:



D(a, b) = “distance” between a and b

Dependence Metrics: Similar data vectors exhibit dependence: they “move together” in similar ways; the range of the coefficients is [-1 .. 1]:



Distance Metrics

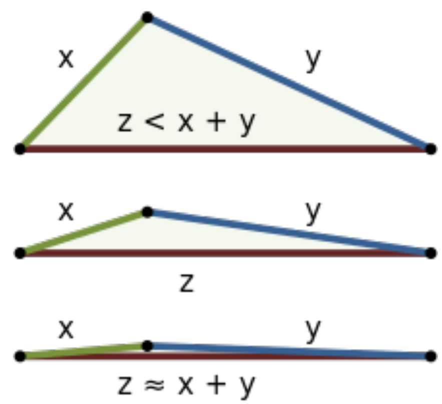
A **Distance Metric** obeys typical geometric laws:

A **metric** on a set X is a function (called the *distance function* or simply **distance**)

$$d : X \times X \to \mathbf{R},$$

where **R** is the set of real numbers, and for all x, y, z in X, the following conditions are satisfied:

1. $d(x, y) \geq 0$ (*non-negativity*, or separation axiom)
2. $d(x, y) = 0$ if and only if $x = y$ (*identity of indiscernibles*, or coincidence axiom)
3. $d(x, y) = d(y, x)$ (*symmetry*)
4. $d(x, z) \leq d(x, y) + d(y, z)$ (*subadditivity* / *triangle inequality*).



A set with an associated Distance Metric is called a **Metric Space**.

Distance Metrics

Computer Science

A variety of metrics have been developed, from fields as diverse as game playing to pattern recognition, and the most important of these is as follows:

Sum of Absolute Difference (Manhattan Distance, L_1 Norm):

$$d_{\mathbf{SAD}} : (x, y) \mapsto \|x - y\|_1 = \sum_{i=1}^{n} |x_i - y_i|$$

Sum of Squared Difference:

$$d_{\mathbf{SSD}} : (x, y) \mapsto \|x - y\|_2^2 = \langle x - y, x - y \rangle = \sum_{i=1}^{n} (x_i - y_i)^2$$

Mean Absolute Error:

$$d_{\mathbf{MAE}} : (x, y) \mapsto \frac{d_{\mathbf{SAD}}}{n} = \frac{\|x - y\|_1}{n} = \frac{1}{n} \sum_{i=1}^{n} |x_i - y_i|$$

Mean Squared Error:

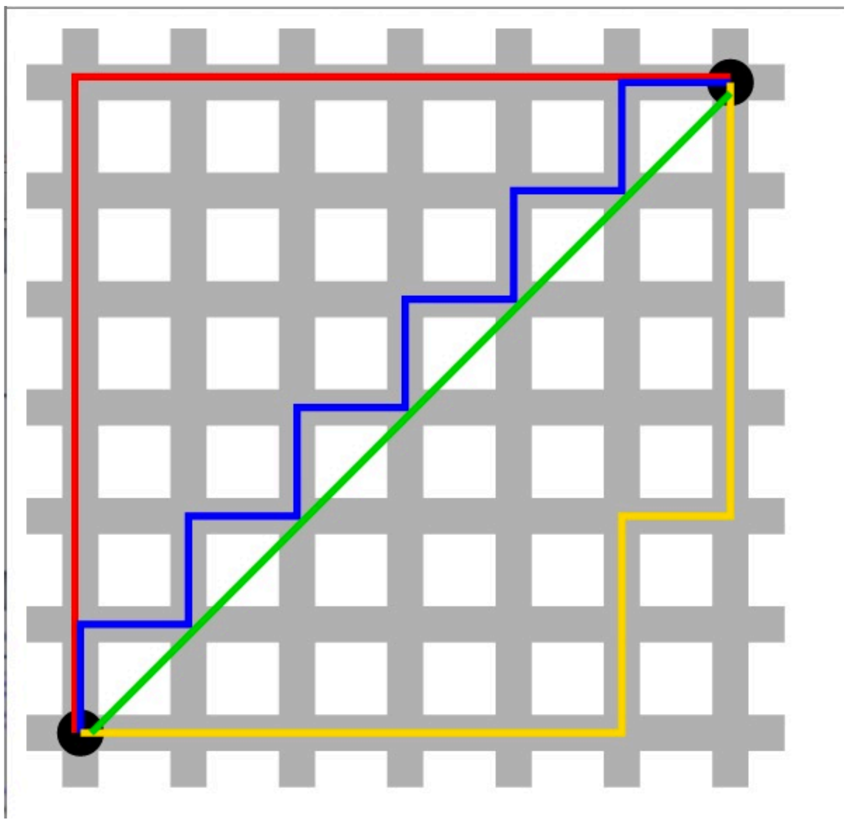
$$d_{\mathbf{MSE}} : (x, y) \mapsto \frac{d_{\mathbf{SSD}}}{n} = \frac{\|x - y\|_2^2}{n} = \frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2$$

Euclidean Distance: (L_2 Norm)

$$d_{\mathbf{2}} : (x, y) \mapsto \|x - y\|_2 = \sqrt{d_{\mathbf{SSD}}} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Distance Metrics

These measures extend our common understanding of the notion of distance to complex mathematical domains (such as vector spaces) and give us tools to understand how similar or dissimilar two objects are.



Dependence Metrics

Two common dependence metrics are as follows:

Correlation (Pearson's Product-Moment Correlation Coefficient):

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Correlation measures the linear dependence of two vectors or random variables X and Y.

Cosine Similarity:

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}}$$

Cosine similarity measures the cosine of the angle between two vectors of length N in N-dimensional space.

NOTE that these are similar calculations, except that correlation subtracts the mean from each point. For musical signals of any length, the mean will be very close to 0, and so these are effectively the same.

Distance Metrics

Dependence metrics can be converted (almost) into distance metrics by the simple expediency of subtracting them from 1.0:

Cosine Distance = 1.0 + Cosine Similarity

Pearson's Distance = 1.0 + Correlation Coefficient

Now these are in the range [0..2], with 2 indicating the strongest possible dependence; these are not actually distance metrics, since they do not satisfy the triangle inequality; however, this does not prevent them from being extremely useful!!

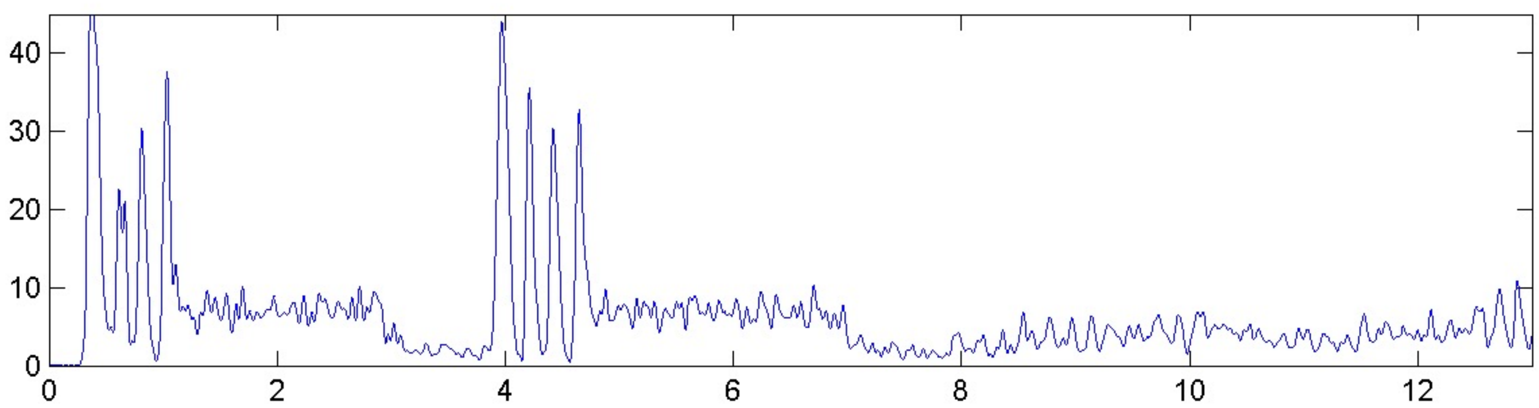
Onset Detection (Spectral-Based)

Computer Science

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation
4. Accumulation

Novelty curve



Onset Detection (Spectral-Based)

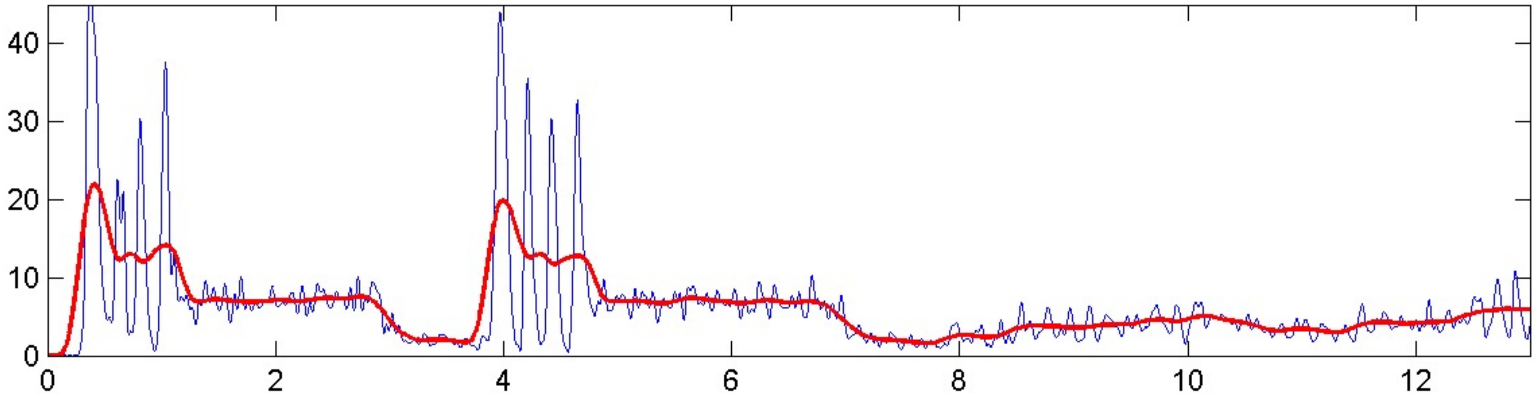
Computer Science

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation
4. Accumulation
5. Normalization

Novelty curve

Subtraction of local average



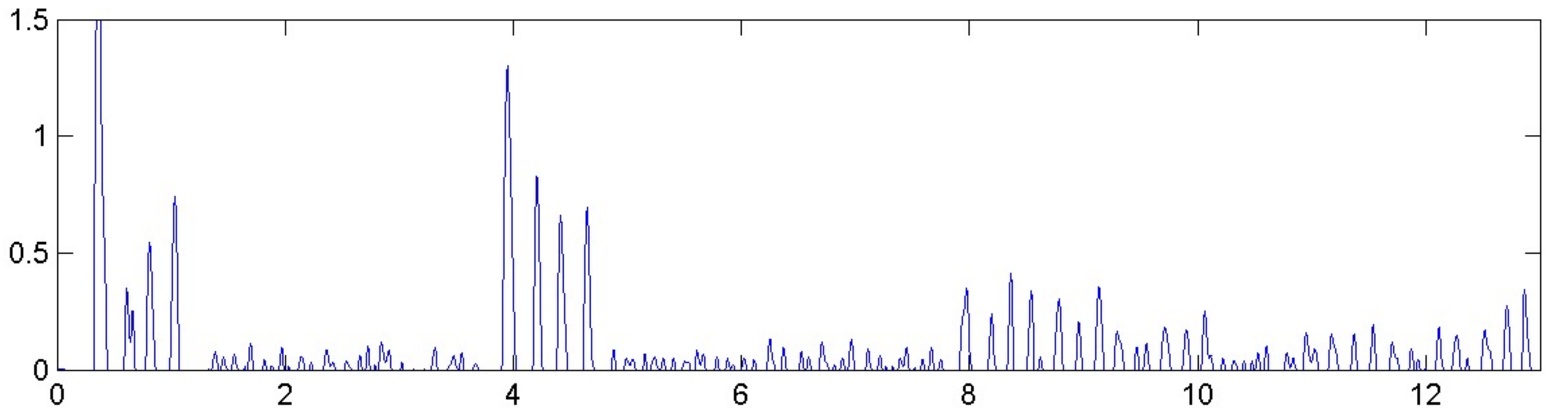
Onset Detection (Spectral-Based)

Computer Science

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation
4. Accumulation
5. Normalization

Normalized novelty curve



Onset Detection (Spectral-Based)

Computer Science

Steps:

1. Spectrogram
2. Logarithmic compression
3. Differentiation
4. Accumulation
5. Normalization
6. Peak picking

Normalized novelty curve