Real-time Onset Detection in Musical Signals

EE779 : Course Project

Yash Bhalgat (13D070014)
Kalpesh Patil (130040019)

October 27, 2016

Overview

Problem Statement

The aim of the project is to:

- Study and present various state-of-art methods used in **real-time onset detection** in audio (music) signals.
- Formulation of ODFs (Onset-detection functions) by different techniques and using their peaks for the localization of onsets in the signals.

Motivation

- Onset detection is very useful in temporal segmentation of audio signals, which is an important step in **beat-synchronous analysis**
- Onsets also play a crucial role in **score following**. It is the synchronisation of a computer with a performer playing a known musical score.

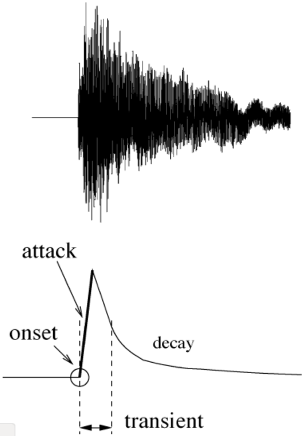
Score following is still an active area of research which stands at the heart of *AI, pattern recognition* and *signal processing*!

Background

Onsets: instant marking the beginning of a transient or a note.

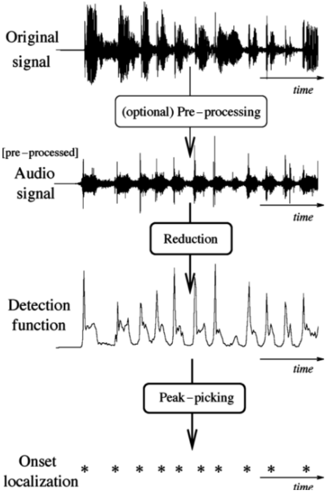
Transient: short interval during which the signal behaves in a relatively unpredictable way.

Attack: The time interval during which the amplitude envelope increases



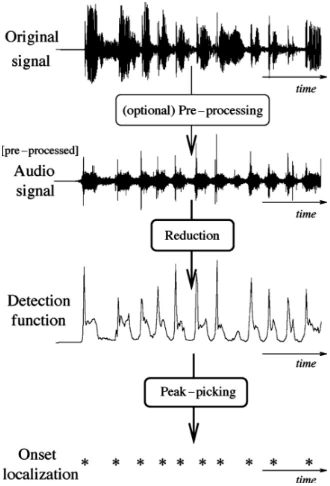
Onset Detection Function

- **Onset Detection Functions**
  - Highly subsampled detection sequence
  . showing the occurrence of transients
  - A measure of the unpredictability of
  . that frame indicating onsets
  - Vector passed to peak detection
  . algorithm for onset detection
- **Peak Detection**
- **Threholding**



Peak Detection

- **Onset Detection Functions**
- **Peak Detection**
  - Appropriate ODFs give identifiable peaks at the onsets or abrupt events
  - Identify local maxima in ODF
  - Comparing current ODF value with neighbouring sample values (this results in latency)
- **Threholding**



Thresholding

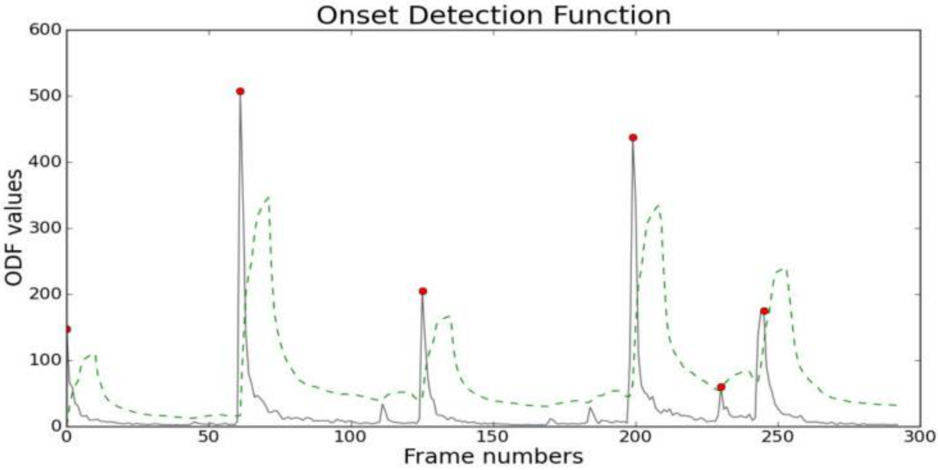
- **Onset Detection Functions**
- **Peak Detection**
- **Threholding**

$$thresh_n = \lambda \times median(O[n_m]) + \alpha \times mean(O[n_m]) + N$$

$O[n_m]$ *is the previous m values of the ODF at frame n*

Every ODF peak that is above this threshold is taken to be a note onset. See figure below for demo illustration.

Thresholding



Energy Method

- Most simple and computationally efficient

$$E[n] = \sum_{m=0}^{N} x(m)^2$$

- Onsets often have more energy than the steady-state component of the note, because that is when the excitation is applied
- Larger changes in the amplitude envelope of the signal are expected to coincide with onset locations

$$ODF_E(n) = |E(n) - E(n-1)|$$

Spectral Difference Method

- Successful in detecting onsets in polyphonic signals (multiple notes simultaneously) and 'soft' onsets created by instruments which do not have a percussive attack

$$X(k,n) = \sum_{m=0}^{N-1} x(m)w(m)e^{\frac{-2j\pi mk}{N}}$$

- Spectral difference ODF (ODFSD) is calculated by examining frame-to-frame changes in the Short-Time Fourier Transform

$$ODF_{SD}(n) = \sum_{k=0}^{N/2} ||X(k,n)| - |X(k,n-1)||$$

Complex Domain Method

- Combined magnitude and phase information
- Magnitude prediction remains same. Assuming a constant rate of phase change between frames for phase prediction:

$$\hat{R}(k,n) = |X(k,n-1)|$$
$$\hat{\phi}(k,n) = princarg[2\phi(k,n-1) - \phi(k,n-2)]$$

- Euclidian distances in predicted and actual phasors gives ODF:

$$\Gamma(k,n) = \sqrt{\hat{R}(k,n)^2 + R(k,n)^2 - 2R(k,n)\hat{R}(k,n)\cos(\phi(k,n) - \hat{\phi}(k,n))}$$

$$ODF_{CD}(n) = \sum_{k=0}^{N/2} \Gamma(n,k)$$

Linear Prediction Method

- LP "error"' is useful in creating ODFs.
- ODF is then the absolute value of the differences between the actual frame measurements and the LP predictions
- The ODF values are low when the LP prediction is accurate, but larger in regions of the signal that are more **unpredictable**, which should correspond with note **onset locations**.
- All the previous methods can be combined with LP prediction to give better results.

Linear Prediction Method

$$\hat{x}(n) = \sum_{k=1}^{p} a_k x(n-k)$$

Hence, if we apply LP to each of the methods, we get:

$$ODF_{ELP}(n) = |E(n) - P_E(n)|$$

$$ODF_{SDLP}(n) = \sum_{k=0}^{N/2} ||X(k,n)| - |P_{SD}(k,n)||$$

$$ODF_{CDLP}(n) = \sum_{k=0}^{N/2} ||\Gamma(k,n)| - |P_{CD}(k,n)||$$

Sinusoidal Modelling

Fourier's Theorem: Any periodic waveform can be modelled as the sum of sinusoids at various amplitudes and harmonic frequencies.

Additive synthesis: adding together many sinusoidal components modulated by relatively slowly varying amplitude and frequency envelopes

$$y(t) = \sum_{i=1}^{N} A_i(t) sin\left[\theta_i(t)\right]$$

$$where \quad \theta_i(t) = \int_0^t \omega_i\left(t\right) dt + \theta_i\left(0\right)$$

Partial tracking: Calculating these parameters for each frame is referred to as peak detection, while the process of connecting these peaks between frames is called partial tracking.

Offline Processing Technique

- Transient signals in the time domain can be mapped onto sinusoidal signals in a frequency domain using DCT.
- Not suitable for real time since DCT frame length required is very large
- A multi-resolution sinusoidal model is then applied to the signal to isolate the harmonic component of the sound
- Onset Location is determined by abrupt increase in energy of the frame.

Online Processing Technique

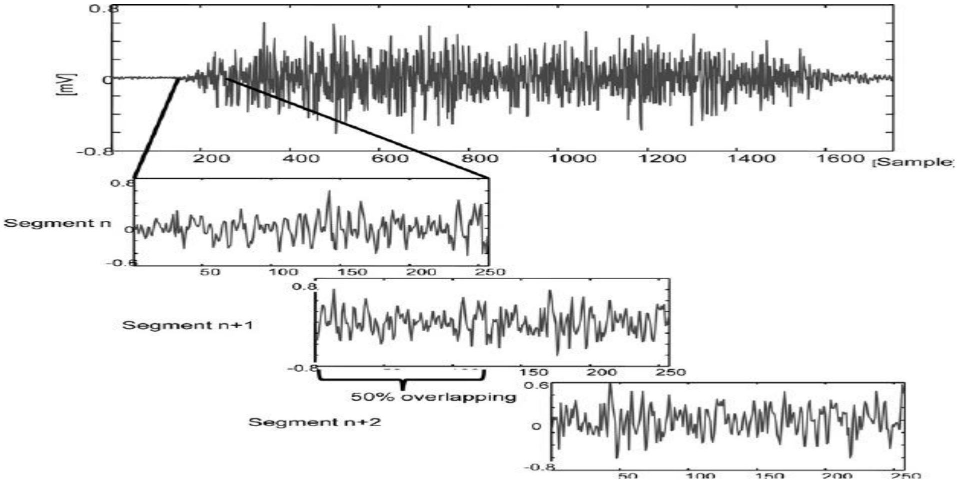
Proposed in the paper

- During the steady state of a musical note, the harmonic signal component can be well modelled as a sum of sinusoids, whose amplitude and frequency are slowly evovling in time.
- Absolute values of the frame-to-frame differences in the sinusoidal peak amplitudes and frequencies should be quite low for steady state
- Amplitudes of detected sinusoidal partials increases during attack region
- Large amplitude and/or frequency deviations in the partials implies existence of Onsets

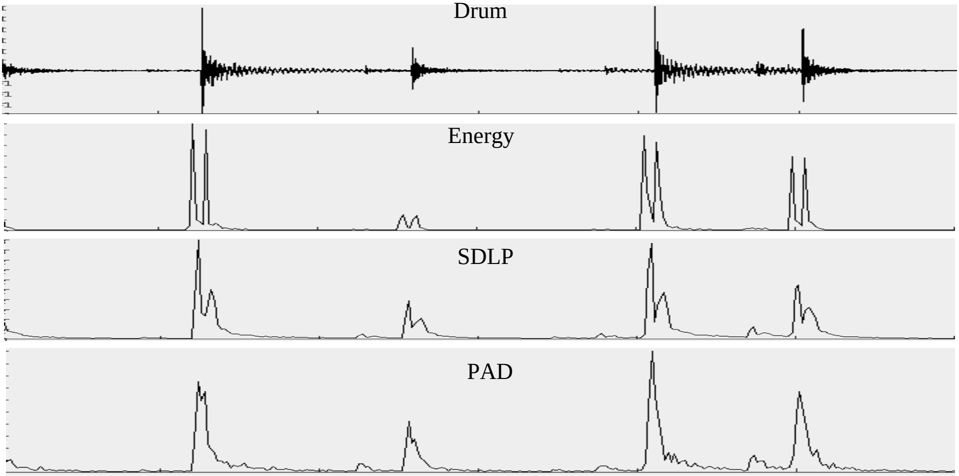
Pre-processing

- The signal is converted to overlapping, fixed-sized frames of audio, each having 4 buffers - 512 x 4 in duration.
- We used a 50% overlap for the frames and a samples.
- It consists of the most recent audio buffer which is passed directly to the algorithm, combined with the previous three buffers which are saved in memory.
- The time taken by the algorithm to process one frame of audio must be less than the duration of audio that is held in each buffer. Sampling rate = 44kHz and buffer size = 512 samples. So, the algorithm must be able to process a frame in 11.6 ms or less when operating.

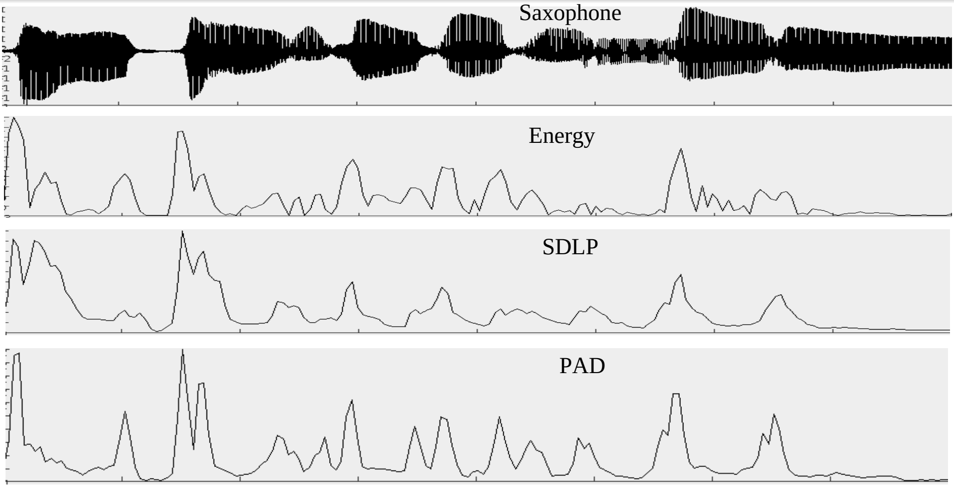
Pre-processing



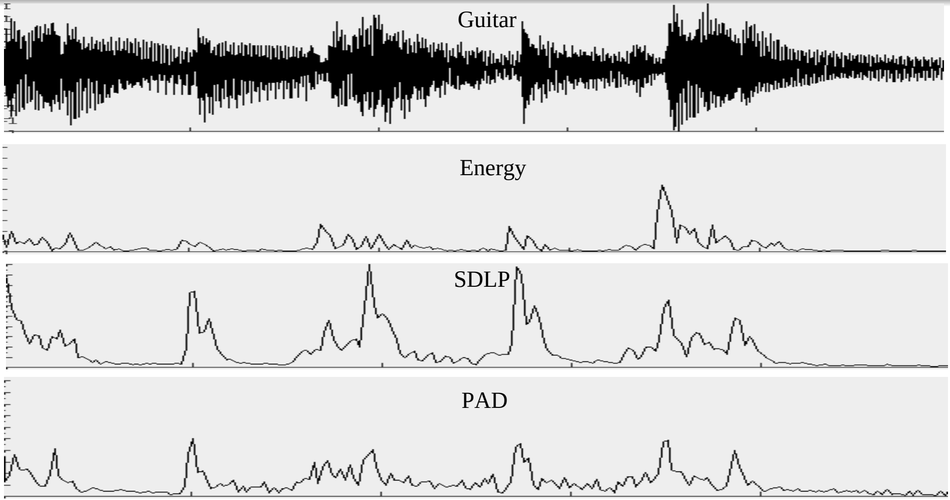
Observations



Observations



Observations



Detection Results

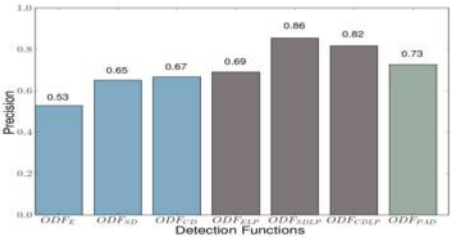
- The detection accuracy of the ODFs was measured by comparing the onsets detected using each method with the reference samples in the Modal database.
- To be marked as 'correctly detected', the onset must be located within 50 ms of a reference onset.

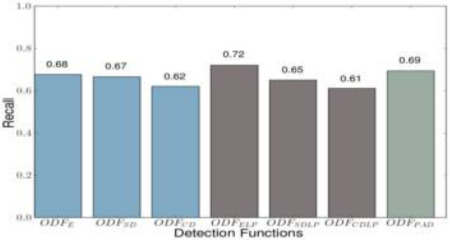
$$P = \frac{C}{C + f_p}$$

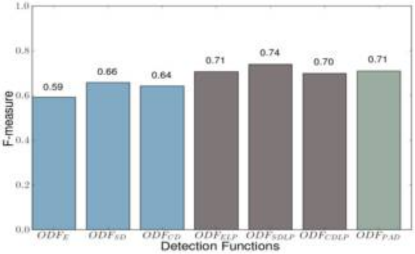
$$R = \frac{C}{C + f_n}$$

$$F = \frac{2PR}{P + R}$$

Results


Precision
0.53
0.65
0.67
0.69
0.86
0.82
0.73
ODF_E ODF_{SD} ODF_{CD} ODF_{ELP} ODF_{SDLP} ODF_{CDLP} ODF_{PAD}
Detection Functions


Recall
0.68
0.67
0.62
0.72
0.65
0.61
0.69
ODF_E ODF_{SD} ODF_{CD} ODF_{ELP} ODF_{SDLP} ODF_{CDLP} ODF_{PAD}
Detection Functions


F-measure
0.59
0.66
0.64
0.71
0.74
0.70
0.71
ODF_E ODF_{SD} ODF_{CD} ODF_{ELP} ODF_{SDLP} ODF_{CDLP} ODF_{PAD}
Detection Functions

References

[1] *John Glover, Victor Lazzarini, Joseph Timoney* - Real-time detection of musical onsets with linear prediction and sinusoidal modeling, EURASIP '11

[2] *Robert McAulay, Thomas Quatieri* - Speech Analysis/Synthesis based on sinusoidal representation, IEEE Transactions '86

[3] Sinusoidal Modelling - http://www.music.mcgill.ca/ ich/classes/dafx_book.pdf

[4] *Juan Pablo Bello, Laurent Daudet et al* - A Tutorial on Onset Detection in Music Signals, IEEE Transactions '05

Thank You!