Audio Onset Detection

Music Information Retrieval

Tessy Troes

for the Music Information Retrieval course
2016/2017

Universitat Pompeu Fabra Barcelona

Task Description: Audio Onset Detection

main objective:

find the time locations of all sonic events in an audio signal

- **sonic event** = new note
  onset = a single instant chosen to mark the temporally extended transient

- detection **difficulties**: extended transient, ambiguous events (e.g. vibrato), polyphonic signal, asynchronous onsets

- **MIR applications**: segmentation of a single track in a mix, drum transcription or complex mixes databases segmentation

- proposed to **MIREX** in **2005** by Paul Brossier and Pierre Leveau.

Evaluation metrics

- Which **calculated onsets** are "**correct**" within a tolerance time-window of +/- 50 ms?

  within the frame: *correct detection* (CD)

  outside the frame: *false negative* (FN)

  "false alarm": *false positive* (FP)

- **Evaluation metrics:**

Precision: $P = Ocd / (Ocd + Ofp)$

Recall: $R = Ocd / (Ocd + Ofn)$,

F-measure: $F = 2*P*R/(P+R)$

with:

Ocd = # CD

Ofn = # FN

Ofp = # FP

Audio Onset Detection function structure

(1) Pre-processing:

- approximation to the **mechanics** of the human **cochlea**
- filter to distribute audio into **multiple frequency bands**

(2) Reduction:

- audio signal transformed into a **detection function**
- function manifests the **occurrence of transients** in the original signal
- based on HFC, phase, wavelet

(3) Peak-Picking:

- peak-picking in audio onset detection function
- either **flexible or adaptive**

Human Evaluation Strategies

hand-labeling:

- ***signal plot:*** efficient to label percussive signals
- ***spectogram:*** FT-frequency resolution trade-off
- listening to ***signal slices***

⇒ *time- and concentration-consuming task*

tools for (semi)-automated annotation:

- Sound Onset Labellizer
- SonicVisualiser
- Lucerne Audio Recording Analyzer

State-Of-The-Art

- **Approaches** in time domain, frequency domain, phase domain or combinations; current trend: **spectral flux**

- detection of **percussive onsets** is considered as solved
  (i.e. at MIREX16, f-score for most algorithms > 85)
- **Open challenges**:
  singing voice as well as soft onsets (in woodwinds and bowed string instruments)
  => algorithms incorporating phase or pitch information

- **2013:** convolutional neural networks (Jan Schlüter, Sebastian Böck)
- **2013 - 2016:** QMUL's Note Onset Detector algorithm:
  calculates an onset likelihood function for each spectral frame then picks peaks in a smoothed version
- **2016:** best performances by Böck's SuperFlux and ComplexFlux

Chosen methods

(a) SuperFlux:

- presented during **ISMIR 2013**:
  recognized as the **best open-source implementation** available
- based on the universal **spectral flux** onset detection algorithm
- enhanced by a **vibrato suppression filter**:
  "instead of calculating the difference from the same bin of a previous frame it includes a special trajectory-tracking stage"

(b) Essentia:

- implementations compute **various onset detection functions**:
  HFC, complex, complex_phase, flux, melflux, rms
- chosen: HFC - **High Frequency Content** detection

Available datasets

- **10 annotated databases** found:
  mirex05 onset, beatboxset1, CMMSD, DREANSS, ENST-Drums, holzapfel:onset, IDMT-SMT-Drums, MusicNet, ODB, Mirex15 Onset

- 3 chosen for this task:
  - **mirex05 onset:**
    - used for MirEx Onset Detection in 2005, 2006, 2007, 2009, 2010, 2011, 2012, 2013, 2014, 2015
    - various instruments (cello, saxophone, piano, guitar, ...)
  - **beatboxset1:**
    - 14 vocal percussion percussion / beatboxing recordings
  - **ENST-Drums:**
    - 3 professional drummers recorded 75 minutes respectively on their own drum kit

First results - SuperFlux

mirex05	SuperFlux	SuperFlux	SuperFlux
	F-measure	*Precision*	*Recall*
AVERAGE	**0.752**	**0.756**	**0.798**
STANDARD DEVIATION	**0.212**	**0.170**	**0.257**

beatboxset1	SuperFlux	SuperFlux	SuperFlux
	F-measure	*Precision*	*Recall*
AVERAGE	**0.874**	**0.804**	**0.968**
STANDARD DEVIATION	**0.061**	**0.106**	**0.029**

ESNT-Drums	SuperFlux	SuperFlux	SuperFlux
	F-measure	*Precision*	*Recall*
AVERAGE	0.730	0.966	0.599
STANDARD DEVIATION	0.098	0.036	0.126

– low recall for string instruments
– low precision for saxophone and clarinet
– low recall for complex drum patterns

First results - Essentia

mirex05	**Essentia** *F-measure*	**Essentia** *Precision*	**Essentia** *Recall*
AVERAGE	**0.736**	**0.757**	**0.744**
STANDARD DEVIATION	**0.222**	**0.254**	**0.197**

beatboxset1	**Essentia** *F-measure*	**Essentia** *Precision*	**Essentia** *Recall*
AVERAGE	0.793	0.793	0.795
STANDARD DEVIATION	0.151	0.151	0.156

ESNT-Drums	**Essentia** *F-measure*	**Essentia** *Precision*	**Essentia** *Recall*
AVERAGE	0.596	0.985	0.450
STANDARD DEVIATION	0.171	0.015	0.170

– low recall and precision for classic
– low precision for saxophone and clarinet
– low recall for complex drum patterns

Comparison between methods

- both methods show **similar behaviour** towards the datasets
- **Essentia** algorithm has a slightly **better precision** score for dataset #1 (by 0.001) and dataset #3 (by 0.019)
- **SuperFlux** performs notably **better** when it comes to **recall**, especially for datasets #2 and #3.
- **SuperFlux** had a **higher f-score** and **less standard deviations**

The algorithm of our choice: SuperFlux

- detection of **positive changes in energy** over time
  ⇒ based on common spectral flux algorithm
- special **trajectory-tracking** stage
  ⇒ processing the signal in frame-wise manner

  - improvement on *softer onsets*
  - apt for both *off- and on-line* use

Universitat Pompeu Fabra Barcelona

Robustness tests

- use of **audio degradation toolbox** for Matlab
- four **degradations** applied:

  live recording, radio broadcast,
  smartphone playback, vinyl recording

- MIREX05: **smartphone playback improved F-score**!

  ⇒ investigate degradation as pre-processing stage

Robustness test results:

MIREX05:

MIREX05												
	liveRecording			radioBroadcast			smartphone			vinyl		
Average	0.702	0.868	0.604	0.511	0.62	0.445	0.738	0.932	0.639	0.708	0.788	0.656
Standard Deviation	0.079	0.07	0.119	0.08	0.111	0.088	0.08	0.061	0.146	0.073	0.067	0.125

ESNT - Drums:

ESNT-Drums												
	liveRecording			radioBroadcast			smartphone playback			vinyl		
Average	0.748	0.719	0.792	0.359	0.323	0.433	0.782	0.708	0.914	0.652	0.548	0.896
Standard Deviation	0.150	0.175	0.145	0.148	0.162	0.122	0.194	0.219	0.112	0.257	0.267	0.126

Parameter tuning - Strategies

- **local group delay weighting scheme**
- add **uniform filter** to code
- play around with **fixed threshold** from 0.1 up to 2

  ⇒ *different combination of methods improved instrument-specifically*

Parameter tuning - Results

- **MIREX05:**
    - **F-score:** + 0.037
    - **Recall:** + 0.025
    - **Precision:** + 0.018

      (biggest improvement for clarinet: + 0.164)

- **ESNT-Drums:**
    - less improvement

      - constant trade-off between precision and recall
    - **F-score:** + 0.001

Contributions to the state-of-the-art

- **degradation** units at **pre-processing** stage
- instrument-specific **parameter tuning**
- improvement on **13 out of 15** MIREX05 instruments:
  - distguit1: + 0.028
  - guitar3: + 0.009
  - jazz3: + 0.037
  - trumpet1: + 0.007
  - classic2: +0.029
  - pop1: +0.012
  - piano1: +0.047
  - sax1: +0.055
  - rock1: +0.007
  - synthbass: +0.019
  - jazz3: +0.061
  - violin2: + 0.139
  - clarinet1: + 0.164

Future work

- experiment with further **degradation units**
- **impact** of different **degradation units** on onset detection
- define an instrument-specific, **adaptive threshold**

Audio Onset Detection

Music Information Retrieval

Tessy Troes

for the Music Information Retrieval course
2016/2017

Universitat
Pompeu Fabra
Barcelona