# Multimedia Systems

## Assignment A3. Pitch and Rhythm

FEUP 2019/2020 – António Sá Pinto 18/05/2020

## 1. Goal

The goal of this assignment is the exploration of classic algorithms (e.g. autocorrelation, DFT-based) to extract fundamental information from musical signals:

- *Pitch* (in monophonic signals);
- *Beats* and *Tempo*;

The autocorrelation of a signal describes the similarity of a signal against a time-shifted version of itself. For a signal $x$, the autocorrelation $r$ is:

$$r(k) = \sum_n x\,(n)x(n-k)$$

In this equation, $k$ is often called the **lag** parameter. $r(k)$ is maximized at $k = 0$ and is symmetric about $k$. The autocorrelation is useful for finding repeated patterns in a signal. For example, at short lags, the autocorrelation can tell us something about the signal's fundamental frequency. For longer lags, the autocorrelation may tell us something about the tempo of a musical signal.

## 2. Resources

**Available implementations:**

- **pitch_estimation.ipynb** Notebook (week 1)
- code from librosa.beat.beat_track

**Sound material:**

- A3_sounds.zip
  - mp_oboe.wav
  - mp_organ.wav
  - rhythm.wav
  - voiced.wav

**Further resources:**

- FMP Notebook

## 3. Instructions

- When you're asked to choose sound examples, you are expected to opt for "illustrative" examples and not "simpler" examples, in a way that helps your explanation and demonstrates your understanding of the task at hand.
- Your plots should be clear and illustrate your comments, thus adjust the parameters of the analysis accordingly, in a way that an appropriate amount of detail (i.e. the "zoom") in both axes is exposed. You do not need to include many plots (which sometimes are redundant).
- Depart from **pitch_estimation.ipynb** and compute the rest of this assignment in the same notebook.

- If it is not stated otherwise, you're allowed to use other external libraries in your code. All you have to do is indicate their use and cite them in the report.
- If you get stuck in some programming error, and are unable to produce the required outputs, try to explain as good as possible what were the expected results.

## Task 1 – Pitch

### Instant Analysis

Depart from **pitch_estimation.ipynb** and for the 2 monophonic signals (**mp_oboe.wav**; **mp_organ.wav**), select a relevant frame (i.e. not silence), and plot the output of the autocorrelation function.
Answer the additional questions:

1) By peak-picking (find the location of the maximum), what would be the estimation of the pitch/f0 in
    a. Hz?
    b. MIDI? (*ex. A0*)
2) Is this value correct or incorrect?
    a. If incorrect, explain why this algorithm didn't work.

### Global Analysis

On the previous task, your pitch estimates were frame-based. For the same signals, extend this frame analysis to the full sound, and for each sound:

1) obtain a plot of the pitch estimation; on the x-axis you'll have the frame number, on the y-axis the pitch/f0 in Hz.
2) obtain a plot of the pitch estimation; on the x-axis you'll have the frame number, on the y-axis the pitch/f0 in MIDI Note (ex. A0, B0, B1)
3) Is any of these representations ( 1) or 2) ) a *chromagram*? Explain your answer.

## Task 2 – Rhythm

As you've seen, the ACF may be used to estimate the periodicity of a signal. Last week, I asked you to think about other main characteristics of the music signals may be characterised by periodicities, and if it would make sense to use the autocorrelation function for those potential tasks. I assume you already know the answer: yes!

Try to reuse as many code from the Task 1 and last class' exercises as possible!

Download **rhythm.wav**, and reapply the analysis from last class' *Task2 – Beat tracking* (based on librosa.beat.beat_track). Produce 2 plots containing the following information, respectively:

- Onsets;
- Beats estimation;

Answer the following questions:

1. Do a subjective evaluation on the quality of the beat tracking.
2. This rhythm shows a steady constant tempo. Obtain the global tempo estimative in bpm (you can do it by hand, no need to code it).
3. How many *downbeats* are in this musical example? At what times?
4. What is the novelty function that you used? (i.e., the library implements). What functions do you know that could be used for the same end?
5. What would you obtain if you applied the autocorrelation function to this novelty function? Explain your answer.

### Task 3 – Pitch and Rhythm I

Download **voice.wav**. This music excerpt contains 9 notes (part of a musical scale), singed by a feminine voice. With the help of Sonic Visualiser, answer the following:

1- What is the chroma and height of the 9 notes?
2- Assuming each note represents a beat, what is the tempo (in bpm) of this excerpt.

Explain your answers and include the visualisations that best justify your rationale.

### Task 4 – Pitch and Rhythm II

Implement Task 3 in code (your Jupyter notebook) that computes the answers.

*Hint: You already have most of the code needed for this to work. For a quick fix to obtain the expected result you'll need to add the following:*

- *Choose the frames in which you have to estimate the pitch (exclude the silences).*
- *Choose a way to average the estimates for each note across the different frames.*

## 4. Delivery/Deadline

Write the Report. Maximum 6 pages, with references.
You're advised to use a compact layout, such as in a 2-column conference/journal paper template (e.g. ISMIR).

You have to deliver your working code in Jupiter Notebook (*.ipynb), as well as the report in pdf.
Deliver your working code (python notebook) in a zip.

Send me a single zip with both your report and code zip by email (file named A3_GXX.zip) until Jun 4th at 23:59.

## 5. Evaluation Criteria

- 5 points: Task1
- 5 points: Task2
- 4 points: Task3
- 3 points: Task4
- 3 points: quality of written report