

Lab 7: Pitch tracking and modifications

Part 1. Making a pitch tracker

In this section we will design a pitch tracker. Get the following sound:

<http://courses.engr.illinois.edu/cs498ps3/sounds/beauty.wav>

We want to find the pitch of the singing over time, as well as if the sound is pitched or not at any point in time. To do so we will design an autocorrelation pitch tracker.

Do the following:

1. Perform an STFT of the sound
2. Using the DFT representation of each frame compute the frame's autocorrelation
3. Invert each frame's autocorrelation back to the time domain
4. Find the first peak after the main peak (which will be at the first element)
 - We will only look for pitches between 100 and 300Hz
 - Which means that you only need to look for a peak at a specific range
5. If the peak is more than 70% of the main peak we have a pitched frame
 - Otherwise it is an un-pitched frame

Plot the pitch value over time and see if it correlates with what you are hearing (we also have the correct plot in the lecture slides).

Part 2. Changing pitch

We will now correct the pitch of that last horrible note. We will use the zero crossing method to find where the periods are and then we will shift them appropriately to create a pitch of 274Hz.

1. Get samples 303,000 to 445,000 which is the horrible part.
2. Lowpass filter to facilitate the zero crossing finding
 - Pick an appropriate cutoff frequency that will help
 - Use the `filtfilt()` function which will not time-shift the input
3. Find the zero crossing points which should be the starts of each period
4. For each period get the signal from the previous period's start till this period's end
5. Window that signal with a Hann window to remove any discontinuities
6. Overlap add each of the above at a rate appropriate for 274Hz

Play the sound, does it sound more in tune? (it better!)