Matlab Exercise 4: PSOLA- Pitch Synchronous Overlap and Add

1 Introduction

- In this exercise, you must implement and return the following files:

    1. `ex4_main.m` – Complete the main script file to run the experiments. Remember to answer the questions within the file!

    2. `ex4_psola.m` – Function to modify pitch, based on PSOLA

- Encrypted reference files (`*_solution.p`) are provided to show you the "intended" functionality of the exercise code. You should run (`ex4_wrapper_solution.m`) to get the reference functions running and visualize and hear the reference results. Also, play around with the parameters in this script. Your solutions do not have to be identical, but it is good to check that you are within the ball park of the intended solutions.

- Do not forget to attempt/analyze the questions at the end of the script `ex4_main.m`. In case your scripts have inaccuracies, these answers could fetch you some points.

- Return your answers to MyCourses by 23:59 on Tuesday, October 9th, 2018.

2 Pitch Synchronous Overlap and Add

While an exhaustive explanation of PSOLA has already been presented in the lecture (refer to lecture notes on *Short-time Processing of Speech*), the goal in this section is to provide complementary information, elaborating on the step-wise comments provided in (`ex4_psola.m`), to successfully complete the exercise.

- The first step in modifying the pitch of a data stream is to estimate the true fundamental frequency. We obtain the estimates by implementing the exercise solutions from exercise 2. You may use either the autocorrelation method or the Cepstral method to this end. The true fundamental frequency is central in detecting and extracting pitch-period segments, for pitch synchronization.

- In the function, begin with an arbitrary start point. We provide this start point as idx $= \lceil \frac{fs}{\text{constant}} \rceil$. Feel free to vary the constant and observe the changes in results. Starting with this data sample, it is then required to find the frame the data sample corresponds to; Use information in *hop_size* to compute the frame-index.

- Once the frame-index is obtained, use the true fundamental frequency of the frame to compute the index (*center_index*) of pitch period boundary. This can be computed by searching for the sample point with maximum value within the pitch period.

- Next step is to extract data segment two pitch periods. To this end, use the previously computed *center_index* as the reference sample point to extract the pitch periods.

- When the data segment is extracted, apply windowing and use it to construct the pitch modified signal. Note that the length of the pitch modified signal *should* be equal to the length of the original signal.

3   Useful functions

zeros, ones, length, ceil, max, round, audioread, resample, size, figure, subplot, plot, title, xlabel, ylabel, fft, imagesc, abs, flip,