E85.2607 – Advanced Digital Signal Theory
Assignment 3: PSOLA and friends
Due: 2010-03-28 by 11:59pm

Instructions

Send me an email containing your solutions by the due date listed above. Please attach a zip file containing your solutions, including the following:

1. A writeup including any requested plots as a **PDF** file. All Word documents will be ignored.
2. Any Matlab code you wrote as separate plain text files.
3. Any interesting sounds you generated.

Your submission should be your own work. If you're having trouble please send me an email or come to my office hours.

Lateness policy

Please note the due date at the top of this page. Late submissions will be penalized by 10% per day, unless an extension has been granted. *This means that if you submit your homework at 12:01am, you will lose points.*

Get started early. Extensions will not be granted on the day the assignment is due. If you think that you will need extra time ask for an extension in advance.

1 Time-stretching showdown: OLA vs SOLA

For this problem you are going to create time-stretching sound examples similar to those in the lecture, except this time we're going to continue to play with Freddy Mercury.

1. Lets start off easy.

   Implement a simple time stretching function to synthesize a sound at half speed by chopping the signal up into blocks and repeating them. Your function should have the following signature:

```
function y = halfspeed(x, L)
% y = halfspeed(x, L)
%
% Stretches the input signal to play at half speed, without changing
% the pitch, by breaking the input into non-overlapping blocks and
% repeating them.
%
% Inputs:
% x - input signal
% L - block size (samples)
```

   Beware of off by one errors!

2. Now lets recreate the scenario depicted in slide 6 of lecture 6. Write a function that uses overlap-add to stretch out the input signal to 50% original speed.

   The function should chop the signal into blocks of the given length, repeat each block twice, overlapping by 50% with each of its neighbors. A schematic of this process is shown in the slide.

```
function y = halfspeed_ola(x, L)
% y = halfspeed_ola(x, L)
%
% Stretches the input signal to play at half speed, without changing
% the pitch, by breaking the input into overlapping blocks and
% overlaping-and-adding them.  Uses a simple linear cross-fade between
% blocks.
%
% Inputs:
% x - input signal
% L - block size (samples)
```

3. Extend `halfspeed_ola` to smooth out artifacts by doing synchronous overlap-add time-stretching.

```
function y = halfspeed_sola(x, L, Kmax)
% y = halfspeed_sola(x, L, Kmax)
%
% Stretches the input signal to play at half speed, without changing
% the pitch.  Uses synchronous overlap-add (SOLA) time stretching
% to compensate for artifacts introduced by block misalignment.
%
% Inputs:
% x    - input signal
% L    - block size (samples)
% Kmax - maximum lag for cross-correlation alignment
```

   What do you need to change to support arbitrary speedup/slowdown factors?

4. Run the first 5 seconds of this sound through the different timescale modification algorithms: `http://www.ee.columbia.edu/~ronw/adst/homeworks/hw2-vocals.wav` Include the wave files with your submission.

   Show spectrograms of the output and comment on the effectiveness of the different algorithms. Can you see a difference in the spectrograms?

   Vary the block size `L` and maximum lag for SOLA. What effect do these parameters have on the sound?

2 Pitch shifting

Use your SOLA code from problem 1 to write a function to shift the pitch without changing the timescale. Since the timescale modification in `halfspeed_sola` is fixed, what will the corresponding shift in pitch be?

```
function y = pitchshift_sola(x, L, Kmax)
% y = pitchshift_sola(x, L, Kmax)
%
% Uses the SOLA time-stretching algorithm to pitch shift the input
% signal x.
%
% Inputs:
% x    - input signal
% L    - block size (samples)
% Kmax - maximum lag for cross-correlation alignment
```

Run the signal from problem one through your function to make Freddy Mercury sound crazy. Now shift the pitch without changing the time scale using your `halfspeed` function. Does it sound different?

Include all of the wave files you generated with your submission.