ELEC484/532 Audio Signal Processing

Assignment 3. Summer 2007 (10 pts)

The main goal of this assignment is to motivate you to experiment with various digital audio effects for creative purposes and to gain insight about some of the ideas we dicussed in class.

Read the questions carefully and do not hesitate to contact me via email if you have any questions or need some clarifications.

The assignment is due **July 26th** in class. The question involve light experimentation with MATLAB and a few calculation here and there. The assignment should not take more than 1 hour or 2 with the exception of 1 point which will take a little bit longer (my guess would be 3-4 hours).

Enjoy, George Tzanetakis

Question 1 (3) pt)

This question explores how the DFT behaves when representing a phasor with a frequency that is between the discrete frequency bins and how windowing can improve the situation. The easiest way to do this question is using MATLAB or Octave.

Plot the DFT magnitude response in dB (DFT size 2048) of a sine wave equal to the DFT bin 600. On the same plot overlap the DFT magnitude response in dB of a sine wave with frequency that would correspond to bin 600.5 (in a sense falling between the crack of two neighboring frequency bins). Finally on the same plot overlap the DFT magnitue rsponse in dB of the second sine wave (600.5) windowed by a hanning window of size 2048. Write two or three brief sentences describing and explaining the three plots.

Question 2 (7pts)

This question asks you to experiment/modify the MATLAB code for the heterodyne filter bank implementation of the phasevocoder (VX_het_nothing.m. Add another output array in addition to DAFx_out called channel_out. Inside the loop fill it up similarly to how the DAFx_out array is filled. Outside the loop play the corresponding channel_out using soundsc. Use the redwheel.wav sample instead of la.wav. For each case write a short sentence explaining the effect:

1. Play the bandpass output corresponding to channel 5 of the filterbank
2. Play the baseband output (before multiplication by het2) corresponding to channel 5 of the filterbank
3. For each grain find the channel index that has the maximum RMS energy. Use that varying index as the selected channel for each grain. In other words instead of using constant 5 for the channel of each grain pick the index of the maximum that can change from grain to grain. What famous character from Star Wars does this voice remind you ? Optional but fun for you and your friends. Record your own voice saying something and transform it this way.
4. For each grain calculate the DFT and find the index of the maximum peak. Compare the sequence of indices generated this way with the indices from the previous question by doing a plot at the end of the two sequences. Use the DFT-based varying index as in the previous question.
5. Look at the sequence of maximum indices generated in the previous question. By eye pick the 3 most common indices and output the summation of the corresponding channels. In this case the channels will remain constant from grain to grain and you will output their sum.
6. Read the following article about a secret until 1976 vocoder-based communication system used between the USA and the UK during World War II for secure voice communications `http://www.nsa.gov/publications/publi00019.cfm`. Briefly describe (2-3 sentences) how you could modify the code of the heterodyne phasevocoder to perform a similar type of encryption. Pay especially attention to Appendix B that describes the encryption process for a signle channel.

7. Actually modify the code. This question is worth only 1 point but will take considerably longer time than the other ones although it's not that difficult if you decide to do it. Implement a SIGSALY inspird encryption system based on the heterodyne phasevocoder implementation. The two main things you will need to implement is quantization to 6 levels for the channel outputs. Ensure that using 6 levels appropriately defined (possibly on a log-scale between the min and max values of all the channels for a recording) gives an intelligible output. Then generate a random-key with six levels and add the key mod 6 to each channel output. Then write the decoding part.

Provide just one .m file with all the cases. There is no need to do any fancy switching between the cases just leave one active and comment the other ones. If you do the last part of question 2 provide a separate .m file for that case.