Low-Pass Filtering of Speech Signals Project

A "professional-looking" report is due Wednesday, November 24, 2004, at 12:00 noon.

The purpose of this project is to experiment with spectral representation and filtering of speech signals and to relate them to the sound and interpretability of processed speech signals to a human observer.

Tasks:

Spectrogram Analysis of Speech

1. Speak, digitize, and record the words "see" and "ooh" as separate utterances. Sample at rates of 16,000 Hz and of 8000 Hz. Do this twice for each of the two utterances.
2. Compute spectrograms. Use 40 msec. windows taken at intervals of 10 msec., i.e., they overlap 30 msec. Use both rectangular and raised cosine windows, i.e., Hanning windows. Use the FFT for computing the spectrogram. Compute the spectrograms for each of the four utterances.
3. Compare the spectrograms for the two utterances of the same word. What is the percentage variation in the energy in each spectral component of the FFT?
4. Average the energies in two adjacent spectral components and in 4 adjacent spectral components in each utterance. Now what is the percentage variation in the energy of these wider spectral bins? What can you say about the relation between spectral resolution and spectral energy stability, and why does this happen?
5. What is the width of the measured frequency bins? Explain.
6. What are the FFT spectral components measuring in regard to the original speech signal?
7. Roughly what is the frequency range of the utterances. Explain!
8. Explain why the energies in corresponding spectral components in the 8000 Hz and 16,000 Hz sampled signals are different, and make a rough calculation to account for the size of this difference.

Low-pass Digital Filtering of Speech

9. Speak, sample, and record the sentences "She said the food is good to eat. Please pass the food" at both 16,000 and 8000 samples per second. Speak the two sentences as a single utterance. Low-pass filter the spoken sentences at both sampling rates.
10. Design low-pass Hanning Window **and** Hamming Window digital filters having bandwidths of 6000 and 4000 Hz, respectively. Compute the filtering using the FFT. How many points are there in the FFT you are using? Explain.
    (a) Explain why the 8000 Hz and 16,000 Hz sampling rates result in different filtered signals.
    (b) How much energy in the original speech signals is filtered out by the filters? Explain how you measure this.
11. Digital low-pass filter the original 16,000 Hz sampled speech using the smallest filter bandwidth such that the speech is still intelligible. What is that filter bandwidth?

Explain your programs and include your code for everything!

Recording and Programming Tips

For recording you'll need a microphone, a speaker (or a headphone with a microphone) and a pc. If you already have access to them, that is great. If you don't have a speaker or a microphone, then you can borrow our headset (which has a microphone as well).
You can get the headset from Tina Trahan (Prof. Cooper's secretary, on third floor, Barus and Holley building) between 9am-4pm. Since we have only one headset, please try to return it as soon as possible after you borrow it from Tina since other people will also need to use it. (If you don't want to go through this borrowing process, you can also buy a simple headset with a microphone, they are very cheap.)

Once you have recording tools, plug them into your pc. There is a sound recorder program in Microsoft Windows (2000 or XP). You can record your utterances as follows:

1) Go to Start-Programs-Accessories-Entertainment,
2) Open "Sound Recorder" program,
3) Record your utterances,
4) Save them using File - Save As option. It will save the file as a *.wav file . You can change the sampling rate of your recording using "Change" button in this "Save As" window.

To read your recordings saved as *.wav files into a Matlab file, you can use Matlab's "wavread" function. To learn more about it, type "help wavread" in Matlab.

To listen to them in Matlab, you can use "soundsc" function in matlab which plays the sound. To learn more about it, type "help soundsc" in Matlab.

Some of the other functions in Matlab that will be useful for this project are:

fft, filter, hanning, plot

But you are not restricted to these functions.

Important: There is also "specgram" function in Matlab. You are not supposed to do things in your project with this function. But you can always use it to double-check your results.