

## ***Music 257***

### **LAB 5: DESIGNING AN EXPERIMENT**

Keep thinking about your final project! What do you want to do? Maybe talk to some other students about collaborating!

#### I. DESIGNING AN EXPERIMENT: Due 5/10/15

In this lab, you and your group will be designing and capturing data to measure one of four perceptual metrics identified below.

#### *OPTIONS:*

- 2-3 equal loudness contours derived from data captured at:

[100 500 1000 2500 5000 10000] Hz

[Intensities of your choice – one softer, one loud but comfortable]

- A listener's audiogram captured at the following frequencies:

[50 100 250 500 1000 2000 4000 8000 16000] Hz

- A critical band curve (frequency as a function of bandwidth) captured for the following two frequencies:

[800 8000] Hz

- A curve depicting the fission boundary of proximity streaming as a function of speed across the following frequency pairs:

[400 500] Hz

[400 650] Hz

[400 800] Hz

[400 2000] Hz

Once you have decided on a perceptual metric you will capture, decide how you will build your test. Unity, GameSalad, or any other platform can be used. Matlab or any other software may be used for stimulus generation or test interface.

Consider what the curves you identify from your measurements might be expected to look like. What method will you use for your measurements? How many trials will you capture for each comparison point? You should capture your data on at least 2-3 subjects. You may use yourselves as subjects. After you have collected all your data, you will need to write a brief report [1-2 pages] summarizing your findings.

This should include:

#### *Intro*

- A brief description of the question you were trying to answer – what and why you collected your measurements. [1 paragraph]

#### *Methods*

- The way the experiment was designed. What type of methodology did you use and why?
- Potential problems with the experiment (Confounds? Error?)
- Information about the sample size and demographics of your experiment [1-2 paragraphs]

#### *Results*

- A plot or plots summarizing the data you captured
- A few sentences summarizing what you found [1-2 plots, 1-2 paragraphs]

E-mail us the report, and please be sure to send a short video of your experiment running!

## II. MATLAB: Due 5/12/16 at Midnight

### **BUILDING A BASIC PITCH TRACKER:**

For this (final!) Matlab assignment, we're going to be building a function called `trackPitch`:

```
[output] = trackPitch(input,fs, windowSize)
```

This function is going to go through the `input` signal, break it up into chunks that are `windowSize` samples long, and write to output a `sawtoothTone` at the loudest frequency in that window.

To do this, we will need to do the following:

1. Create a for loop that repeats `length(input)/windowSize` times.
2. For each repetition of the for loop we get a chunk of the input signal. For example, for the first repetition we get `chunk = input(1:windowSize)`, for the second iteration we get `chunk = input(windowSize+1:2*windowSize)`, so on and so forth. How can we use our for loop to get the right chunk based on what number iteration we are on?
3. Next, we use our `getSpectrum` function from lab 3 on that chunk. We want to get the loudest frequency in that chunk, so use `[M,I] = max(Y)`, where `Y` is the `Y` returned from `getSpectrum`. The `I` value will be the index of the biggest value in `Y`, so we are looking for the corresponding value in the `F` from `getSpectrum`, which will be `F(I)`.
4. Append a sawtooth at the frequency we found to output. We can do this using the `[A,B]` notation, which will concatenate vectors `A` and `B`. Thus, we can recursively append a sawtooth to our output on each iteration of our for loop by typing `output = [output, sawtoothTone(F(I),0, windowSize/fs, 8, fs)]`;

Here I've used 8 harmonics in my sawtooth tone, but feel free to use however many you want.

After you're done with the function, send me a recording of yourself singing a melody for 15 seconds or so (or playing one on a monophonic instrument), and a wav file of the output of `trackPitch` where the input and `fs` is from your recording. You can choose the window size, but let me know what it is!