

ECE180 Lab 6/7: Audio Effects “Demo Tape”

This two-week lab provides an opportunity for you to showcase your MATLAB skills by creating an informative “demo tape” (actually an M4A file) to demonstrate a variety of audio effects. Your demo tape will include instructive narration interleaved with the audio clips to explain what the listener should expect to hear next.

This lab format is much more open-ended than in the past, and you are encouraged to be as innovative as possible. Your work will be evaluated on the degree to which your demo tape informs and instructs the listener, your written report and documented MATLAB code, and your ability to independently design and debug your project.

Deliverables:

1. One .m4a audio file uploaded to the Moodle dropbox. Be sure to test your finished product by opening and playing the .m4a file with a media player or mobile device.
2. Word document uploaded to Turnitin that contains:
 - a. Your names and campus mailbox numbers,
 - b. A paragraph or two to discuss your results for each of the seven effects,
 - c. Citations for all materials that contribute to your project, e.g., the URL(s) for the room impulse response (IR) files, and
 - d. All MATLAB code that you developed for the project; use the copy/paste technique that preserves the syntax color coding.

Constraints:

1. Use speech clips from the ECE180 DFS folder as the source audio.
2. The .m4a audio file duration may not exceed three minutes and must be free of clipping distortion.
3. Starred techniques must be demonstrated with at least three variations that produce obviously distinct results.
4. Both lab partners must contribute equally to the narration.

Techniques to Demonstrate:

1. Reverberation:
 - a. * *Convolution reverb* – convolve a speech clip with a *monaural* (1-channel) room impulse response (IR)
 - b. *Reversed reverb* – time-reverse the source audio, apply convolution reverb, and time-reverse this result)
 - c. * *Wet/dry mix* – sum of the source (“dry”) audio and the reverb (“wet”) audio; a “mix” parameter m controls the percentage of dry and wet, e.g., $m \times \text{dry} + (1-m) \times \text{wet}$
 - d. *Stereo convolution reverb*, i.e., based on a *stereo* (2-channel) room impulse response
2. * *Echo chamber* – multiple distinct echoes
3. * *Modulation* – multiply the source audio by a sinusoid of frequency f_0 .
4. * *Haas effect* – play the same source audio to both stereo left and right channels but with one of the channels time delayed compared to the other; consider delays in different realms such as milliseconds, tens of milliseconds, hundreds of milliseconds, and even longer

Video tutorials:

- Audio players (<http://youtu.be/X-KMWKFe6ml>):
 - play audio array with `sound`
 - avoid clipping with `soundsc`
 - use `audioplayer` object for additional flexibility
- Audio file I/O (<http://youtu.be/-PqHK3IHV4o>):
 - load a built-in audio clip from a `.MAT` file
 - create an audio file with `audiowrite`
 - normalize and resample an audio array
 - read an audio file with `audioread`
- Stereo audio (http://youtu.be/gD_Ae3vG50U):
 - create a stereo (2-channel) audio signal
 - distinguish between left and right channels
 - play audio with `sound`
 - direct monaural (mono, 1-channel) audio to left or right channels
- REAPER digital audio workstation to assemble your audio clips (could also use Audacity):
 - Set up a new project: <https://youtu.be/Moeczw4shdk>
 - Max tracks, adjust level and panning, and render to an audio file:
<https://youtu.be/OyskzDs7Apo>

Useful MATLAB functions and techniques:

- Initialize the workspace (first step of a script):
 - `clc,clear,close all`
- Audio file reading and writing:
 - `audioread`
 - `audiowrite`
 - `audioinfo`
- Resample the speech clips to match the sampling rate of the room IR clips:
 - `resample`
- Record your narration:
 - `audiodeviceinfo`
 - `audiorecorder`
- Normalize audio to a maximum value of unity to avoid clipping distortion; apply this normalization just before you listen to the audio or write the audio file:
 - `y = y / max(abs(y(:))) ;`
- Convolution reverb and echo chamber:
 - `conv`
- Reversing and shifting:
 - `flip`
 - `circshift`