EENG580 Activity 3: Audio Filtering

Activity summary

Overview: Design a filter to remove noise from an audio signal

Setting: Develop in class, then refine after class (any location).

Curricular elements: both tinkering and gaming

Prerequisites: basic familiarity with MATLAB

Topics/concepts covered: FIR filter design and windowing

Learning outcomes: After completing this activity, students should be able to:

- design a narrow-band notched filter
- choose a filter length and window type to balance computations and signal quality
- implement linear phase filters efficiently by exploiting symmetry

Expected time to complete: two 2-hour lab sessions and 4-8 hours of work outside of class

Required hardware/materials: Minimum: a microphone attached to or embedded within a computer, in such a way that the audio input can be queried by MATLAB. A speaker is necessary, but the speakers need not be standardized between groups.

Required instructor interaction: partially supervised, with occasional guidance

Common mistakes/pitfalls: If all students are using different hardware it can be difficult for the instructor to help debug problems. It is recommended to purchase enough microphones to provide one per group. The instructor should verify correct operation of the devices before giving them to students. If possible, the instructor should test the audio recording capability on both PCs and Macs. It may be necessary to turn "noise reduction" on in the microphone device options since signal quality is paramount in this activity.

Method of assessment: instructor graded, based on final product; plus peer voting on subjective evaluations of the competitions

Contributors: Richard Martin, Andrew Klein, Jennifer Hefner, and Cody Watson

---

The views expressed in this document are those of the authors, and do not reflect the official policy or position of the United States Air Force, United States Navy, Department of Defense, or the U.S. Government. This document has been approved for public release; distribution unlimited. This work was supported by ONR grant N000141512442, *Contact: andy.klein@wwu.edu and richard.martin@afit.edu.*

1 Introduction

This is a very open-ended activity. Everyone will have a different approach, or a different "answer." This is likely to be a very frustrating exercise at times, but that is ok; the instructor is here to help you, and you should not hesitate to ask questions. A lot of information is intentionally missing from this lab description; you will need to ask questions the complete this lab. The goal of the activities in this course is to encourage a "maker" mindset wherein you feel free to tinker and explore different approaches. We want you to fail, rapidly and frequently, because that is how you identify and correct errors. Your goal is to get comfortable with the idea of rapid prototyping and rapid response.

Pick a lab partner for this activity. You will collaborate on the process, but you will write up individual reports and receive separate grades.

Obtain a speaker and a microphone from the instructor; these are the same items you used in "Activity 1: Acoustic Impulse Responses." If you are comfortable with the operation of the equipment, you may skip to Section 3; otherwise, work through Section 2 first.

2 Review of the audio hardware

Unless noted otherwise, the basic audio equipment used in this class will be comparable to the following:

- Microphone: Cyber Acoustics Desktop Unidirectional Microphone (CVL-1064), about $7
- Speakers: AmazonBasics USB Powered Computer Speakers (A100), about $14

Ideally, everyone will use the same type of equipment rather than audio gear embedded in laptops, in order to maintain a level playing field. Plug in your equipment, make sure each device is turned on, and make sure each is set to the default device of its type.

Explore the following MATLAB commands:

- `audiorecorder`
- `record`
- `play`
- `stop`
- `resume`
- `pause`
- `getaudiodata`
- `sound`

The explanation in the help files should be straightforward; the only parameter you will need to choose is the sampling rate `fs`. To assist with this, look up the sampling rate of telephony, MPEG audio, voice over IP (VoIP), and compact discs.

Progress to the point where you can record a snippet of speech and then play it back over the speakers. If you can accomplish these tasks, you have sufficient understanding of the hardware interface to complete the entire activity.

3 Tasks

For the first week of this activity, you are expected to study the following material in advance: filtering basics (block diagrams and convolution), FIR filters via windowing, and conversion of low-pass filters into non-LPF types. Before the second week, you should study the material on the phase response, particularly the material on linear phase systems. Videos covering these concepts are available on the course website, and you are encouraged to read additional background material in any of the suggested reference textbooks.

The core task in this assignment is to design a filter that will remove noise from a music file. A simple narrowband noise source is available on the course webpage; for fairness, every group will initially use that same noise source. The file that implements the noise will also scale the noise signal relative to your audio clip and then add the two, so that everyone is working with the same signal to noise ratio.

As a first step, you will need to address the following issues:

(i) What is the spectrum of your signal? What is the spectrum of the noise? What filter specifications do you want to implement?
(ii) Assuming an infinitely long filter, what would be the functional form of the filter coefficients?
(iii) Truncate and delay the filter to make it finite and causal, and then window it. What is the resulting impulse response?

That should be enough to get a baseline filter in place. Once you are comfortable with that, consider the following issues:

(iv) What filter length and window type works best? How short of a filter can you make that still works adequately?
(v) Is your filter linear phase? How can you exploit this to reduce the computations and memory requirements within matlab?
(vi) How would you extend your approach to deal with other noise sources?
(vii) How do you quantify "signal quality?"

For purposes of the competitions, the class should determine a set of filter specifications that need to be met. You can use a different set for your primary design, but the standard specs will be used to judge the competitions.

4 Deliverables

You will turn in a report written using the LaTeX word processing system. A template is available on the course webpage. The report should conform to the style guide for IEEE Signal Processing Society conferences, such as ICASSP, ICIP, or GlobalSIP. At a minimum, explain your measurement methodology, your data processing methodology, and your results. Both format and content matter – in particular, use correct grammar and spelling, and revise your text as needed to make sure you are explaining yourself coherently. Include code sparingly, if at all; in general, it is better to include an algorithm in pseudo-code as a figure.

You should strive to always explain why you did what you did, not just what you did. Your results should all be repeatable – maybe not exactly, but comparably. That is, another researcher may not have access to the exact same environment as you or record the exact same realization of random noise, but you want to enable them to follow the exact same process as you. Analysis is also highly encouraged – explain why the results are what they are, not just what they are. For this activity, photos or sketches of the geometry may help with that.

To improve your grade, include a more detailed and coherent explanation of your results, try a more clever or innovative approach, and include more data and analysis. Feel free to diverge somewhat from the stated tasks if you think the situation warrants it. "Out of the box" thinking will always be rewarded.

5 Competitions

The written report constitutes 90% of the grade, and the remaining 10% will be based on competitions. Competitions will include the following:

- The shortest FIR filter that meets the standardized filter specifications. In this context, "shortest" means "fewest stored coefficients." Note that if you correctly exploit linear phase effects, you can effectively double the filter length without storing additional coefficients.
- Near the end of the activity, the instructor will provide an alternative set of filter specifications. The first group to produce a design (any length, and windowing) meeting these specs will win. Validity will be determined by a plot of the filter's magnitude response overlaid on the filter specifications.
- The best implementation of an extension or advanced technique from the literature, for example: filters with quantized coefficients, which are relevant to fixed-point platforms.

Competitions will be determined by presenting results to the class, immediately before the start of the next activity. Due to the subjective nature of the third competition, the winners will be determined by class vote. The first place group in each category will receive a 10% bonus (for each group member) and the second place group members will each receive 5%; and no individual may receive more than 10% total. Based on typical class sizes, that means about a quarter of you will get 10% and another quarter will get 5%.