Digital Signal Processing
Instituto Superior Técnico
Lab assignment 3 – Filtering
Author: Pedro M. Q. Aguiar.[1] Modified by Luís B. Almeida and Margarida Silveira.

Introduction

In this assignment, we will consider two types of systems: linear time-invariant (LTI) filters and median filters. The goal is to compare these two kinds of systems in what concerns their ability to remove or attenuate the noise that typically corrupts old vinyl records. In the audio signal that will be used in our experiments, this kind of noise is simulated by setting some randomly chosen samples to large, randomly chosen values.

Notes

The report of this lab assignment should not exceed three A4 pages. In this course's assignments, you'll often find between brackets, as in {`command`}, suggestions of Matlab commands that may be useful to perform the requested tasks. You should use Matlab's help, when necessary, to obtain a description of how to use these commands.

Experimental work

1. **Observation of the corrupted signal.**

   R1.a) Load the sound file `fugee.wav`. {`audioread`}

   Listen to the signal. Comment on what you hear. {`soundsc`}

   R1.b) Plot segments of the signal. Comment on what you observe. {`plot`}

   R1.c) Visualize the signal's magnitude spectrum. Comment on what you observe. {`semilogy`}, {`fft`}, {`fftshift`}

   It is often useful to visualize the spectra of sounds with a logarithmic scale for the magnitude, because the sensitivity of our hearing system is approximately logarithmic, and therefore a logarithmic magnitude scale gives a better idea of what we hear than a linear scale.

2. **Filtering with an LTI filter.** Consider an LTI filter with a generic rational transfer function

$$H(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}},$$

   where, naturally, the parameters $\{b_0, b_1, ..., b_M, a_1, ..., a_N\}$ determine the filter's frequency response.

[1] Partly based on a lab assignment authored by João Sanches.

R2.a) Compute the parameters of a Butterworth low-pass filter of order 10 with cutoff frequency $\pi/2$. You don't need to write the values of the parameters in your report.[2] {`butter`}

Visualize the magnitude of the filter's frequency response, using a linear magnitude scale. Comment on what you observe. {`[h,w]=freqz(...)`}

The command `freqz`, by itself, plots the magnitude of the frequency response in a logarithmic scale. However, if you use the form of the command suggested above, you'll obtain the frequency response of the filter in the array `h`, and you can then plot its magnitude in a linear scale.

R2.b) Filter the audio signal with the filter that you designed. {`filter`}

The signal contained in the file `fugee.wav` is a discrete-time signal that was obtained by sampling a continuous-time audio signal. Low-pass filtering the discrete-time signal with a certain cutoff frequency is equivalent to low-pass filtering the continuous-time signal with another cutoff frequency.

What is the cutoff frequency, in Hz, of the continuous-time low-pass filtering that is equivalent to the filtering operation that you just performed in discrete time?

R2.c) Visualize superimposed graphs of the original and filtered signals, and then zoom in to observe the details. Check what happened to the noise (examine a zone in which the amplitude of the music signal is very small), and what happened to the music signal. Comment on what you observe. {`hold`}

R2.d) Visualize superimposed graphs of the magnitude spectra of the original and filtered signals. Comment on what you observe.

R2.e) Listen to the filtered signal. Comment on what you hear.

R2.f) Repeat a)–e) for other values of cutoff frequency, attempting to improve the results. Comment on what you observe.

3. **Filtering with a median filter.**

A median filter is implemented by sliding a window over the input sequence one sample at a time. At each instant, the input samples inside the window are ordered from largest to smallest, and the sample in the middle is the median value. The input-output relationship for the median filter is the following:

$$y(n) = \text{median}[x(n-M), x(n-M+1), ..., x(n), ..., x(n+M)].$$

where the filter's order is given by the number of input samples used to compute each output sample, i.e., $2M+1$.

R3.a) Characterize median filters in terms of causality, stability, linearity, and time invariance. Justify your answer.

R3.b) Filter the audio signal from step R1.a) with a 3rd.-order median filter. {`medfilt1`}

Plot superimposed graphs of the original and filtered signals, and then zoom in to observe the details. Comment on what you observe.

R3.c) Visualize superimposed graphs of the magnitude spectra of the original and filtered signals. Comment on what you observe.

R3.d) Listen to the filtered signal. Comment on what you observe.

R3.e) Try median filters of other orders, attempting to improve the results. Comment on what you observe.

R3.f) Which of the two kinds of filters (LTI low-pass or median) is best for removing the kind of noise that the audio signal contains? Can you give an explanation for that?

---

[2] Butterworth filters are one of the most widely used kinds of filters, but you don't need to worry, at this point, about what are the specific characteristics of this class of filters.