Problems for Discussion 10, 12/04/13

Compiled by Mai Le

1 Some Types of Filters

Notch
- Goal: remove particular frequencies. Frequency response of filter should have a "notch" (missing piece) at the undesired frequency.
- Place pairs of complex conjugate zeros on the unit circle
- Place poles near zeros to reduce the bandwidth of the notch
- Example: to eliminate the component at 0.4π

$$H(z) = \frac{(z - e^{j0.4\pi})(z - e^{-j0.4\pi})}{(z - 0.9e^{j0.4\pi})(z - 0.9e^{-j0.4\pi})}$$

Low pass
- Goal: Remove high frequencies.
- Place zeros near high frequencies
- Place poles near low frequencies
- Example:

$$H(z) = \frac{(z - e^{j0.5\pi})(z - e^{-j0.5\pi})(z - e^{j0.75\pi})(z - e^{-j0.75\pi})(z - e^{j\pi})}{(z - 0.6)(z - 0.8e^{j0.25\pi})(z - 0.8e^{-j0.25\pi})(z - 0.8e^{j0.5\pi})(z - 0.8e^{-j0.5\pi})}$$

Comb
- Goal: add delayed version of signal to itself, used in feedforward and feedback systems.
- A notch filter in which the nulls occur periodically, like a comb
- Example:

2 Matlab example of filter anlaysis

Let's use the LPF example:

$$H(z) = \frac{(z - e^{j0.5\pi})(z - e^{-j0.5\pi})(z - e^{j0.75\pi})(z - e^{-j0.75\pi})(z - e^{j\pi})}{(z - 0.6)(z - 0.8e^{j0.25\pi})(z - 0.8e^{-j0.25\pi})(z - 0.8e^{j0.5\pi})(z - 0.8e^{-j0.5\pi})}$$

First we need to go from

$$H(z) = \frac{b_0 z^{N-M}}{a_0} \frac{\prod_{k=1}^{M}(z - c_k)}{\prod_{k=1}^{M}(z - d_k)} \rightarrow H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}}$$

Then we can use the coefficients of the numerator and denominator, b_k and a_k, respectively in `filter(b,a,x)` or to plot the transfer function with `freqz(b,a,n)`.

For rest of this problem, see Matlab script `disc10_filter_example.m`.

3 Notch Filter Design

Determine the coefficients of the following notch filter if

$$y[n] = a_1 y[n-1] = x[n] + b_1 x[n-1] + b_2 x[n-2]$$

a) It completely rejects the frequency component at $\omega = \frac{\pi}{4}$.

b) It amplifies a DC signal by 2.

4 Effect of Noise on Deconvolution

Suppose a klutzy cameraman moved the camera while snapping a photograph. From sensors, we happen to know the motion was to the right then downwards-right, corresponding to a convolution kernel that looks something like this:

$$h[n,m] = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

(a)

Let's use the idea of deconvolution to remove blur from the image. See `disc10_deconv_example.m`.

(a)

Suppose that the image was also corrupted by white Gaussian noise. How does this affect our deconvolution method? How does the SNR affect the performance of the algorithm?

5 Separability of the DSFT and 2D-DFT

As a reminder, here are the definitions of the DSFT (2D-DTFT) and the 2D-DFT:

$$X(u,v) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x[n,m] e^{jum} e^{jvn}$$

$$X[k,l] = X(u,v)\Big|_{u=\frac{2\pi k}{N}, v=\frac{2\pi l}{M}} = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x[n,m] e^{j\frac{2\pi k}{N}m} e^{j\frac{2\pi l}{M}n}$$

Note that we can have different N and M (dimensions of the 2D-DFT) but lecture had N=M.

(a) Show that the DSFT is equivalent to taking the DTFT in the n-direction, followed by a DTFT in the m-direction (or m then n). Show that the 2D-DFT is equivalent to taking the DFT in the n-direction, followed by a DFT in the m-direction.

(b) Show that if $x[n,m]$ can be written as $x[n,m] = x_1[n]x_2[m]$, then $X(u,v)$ can be written as $X(u,v) = X_1(u)X_2(v)$.

6 Practical Tip: Using `fftshift`

What does `fftshift` do in 1D? What about 2D?

Can we reverse an `fftshift` with another `fftshift`? I.e. Does `fftshift(fftshift(a)) = a`?