

Projects will NOT be accepted late. It is due at the beginning of class.

## Project 1

### Matched Filters: Applications of Cross-Correlation

One possible application of cross-correlation is the detection of a signal embedded within noise. The signal can be estimated by cross-correlating a “template” signal with the input signal. The template signal,  $h[n]$ , is considered to be known *a-priori*. Detection is accomplished by time-reversing the template and convolving it with the input signal. Maxima observed in the output signal indicate lag indices where the template and the shape of interest in the noisy signal are aligned.  $h[n]$  is referred to as a *matched filter*, i.e., a filter which is custom-designed to detect the presence of a specific structure within the input signal, recall from class that  $x[n] \star h[-n] = r_{xh}[n]$ .

#### ECGR 4124/5124 Task 1

This task has students detect the phase and period of a sinusoid embedded within a noisy signal. To do so, we generate a sinusoidal with known frequency but unknown phase and corrupt this sinusoid with additive random noise (steps (a,b)). We then compute a zero phase template of the sinusoid consisting of  $K$  samples using the known frequency of the sinusoid (step (c)). We then attempt to detect the presence of the unknown sinusoid by cross-correlating the template and the noisy signal (step (d)) by detecting peaks, i.e., local maxima in the output signal from the cross-correlation.

1. Generate a sinusoidal sequence  $x[n] = \cos(\frac{\pi}{15}n + \phi)$   $1 \leq n \leq 200$  and  $\phi = 0$ .
2. Corrupt this signal with additive uniformly distributed noise  $e[n]$  of amplitude  $\pm 15$  with distribution  $U \sim [-0.5, 0.5]$ . For a length  $N$  sequence, you can generate the noise with the following MATLAB code: “`noisevals = 30*(rand(N,1)-0.5);`”
3. Compute matched filters of varying length  $h_K[n] = \cos(\frac{\pi}{15}n)$  where  $0 \leq n < K$  and  $K = 10, 20, 40, 80$ .
4. Compute the cross-correlation of the noisy signal,  $x[n] + e$ , and  $h[n]$ .

#### ECGR 4124/5124 Task 1 (30 pts)

##### To Turn in:

You are responsible for turning in the following objects IN THE ORDER LISTED BELOW. **Deductions will be made for reports which are messy and difficult to comprehend.** Using your MATLAB program for steps 1-4 provide the following:

1. (5pts) Turn in your MATLAB code for computing steps 1-4. NO MATLAB CODE IMPLIES NO CREDIT FOR ANY OF YOUR SOLUTIONS. These points are awarded for code that is clearly written and follows the guidelines of the project.
2. (5pts) Using subplot(), plot the cross-correlation signals for each value of  $K$  as a 4x1 (4-row,1-column) matrix of plots.
3. (5pts) Use the location of peaks in the output to estimate the period of the noisy sinusoidal input  $x[n]$ . State the estimated value of the sinusoid period.
4. (5pts) Provide a 1-paragraph explanation of how the variable  $K$  impacts the estimation of the period.
5. (5pts) If the noise amplitude were increased, how would you change  $h[n]$  to get a better estimate?
6. (5pts) Is it also possible to estimate the phase of the function if  $\phi \neq 0$ ? If so, give an idea of how you would do this?  
*Hint :* From the program above, let  $\phi = \pi$  or  $\phi = \frac{\pi}{2}$  and observe how the value of the indices of your peaks change.

### ECGR 4124/5124 Task 2

This task has students detect the frequency of a sinusoid embedded within a noisy signal. To do so, we generate a set of candidate sinusoidal functions, each representing our guess for the unknown frequency. We then cross-correlate these functions with the input signal and then use the magnitude of the peaks in the output as a score for the presence of each frequency in the input signal.

Let our signal be a sinusoid  $x(t)$  containing an unknown frequency that lies in the range from 1 Hz to 2 Hz. This signal is sampled at a rate of 30 samples per second by an analog-to-digital converter to create a digital signal  $x[n]$ . In this circumstance we will proceed as follows:

1. Hypothesize a sinusoidal sequence having frequencies ranging from  $f = [1\text{Hz}, 2\text{Hz}]$  in increments of 0.2 Hz. Create a digital filter  $h[n]$  to cross-correlate with the input  $x[n]$ .
2. Adjust values of your filter coefficients so that the sum of the filter coefficients is equal to 0. This can be accomplished by sampling the sinusoid symmetrically about the origin since the sine function is odd symmetric, i.e.,  $\sin(x) = -\sin(-x)$ . Note that since  $\sin(0) = 0$  the sample  $h[0]$  has no contribution to either your filter or the output value.
3. For each sequence, cross-correlate your signal with the input signal.
4. Observe the output signal and find the cross-correlation signal having largest peak response. This frequency is then the “best” guess for the unknown frequency in the input signal.

### ECGR 4124/5124 Task 2 (40pts)

#### To Turn In:

Turn in your MATLAB code. NO MATLAB CODE IMPLIES NO CREDIT FOR ANY OF YOUR SOLUTIONS.

1. (10pts) On one plot show your 6 filters  $h_i[n]$  representing your hypothesized frequencies. Let your filters be 100 samples long (or 101 samples if you count the sample  $h[0] = 0$ ).
2. (10pts) Using the data file provided, provide a single plot showing the ‘valid’ part, i.e., non-transient part, of your cross-correlations for each of your filters from (1) with the data file containing the unknown frequency. Hint: Use `conv(A,B,'valid')`.
3. (10pts) Indicate which filter has a frequency closest to the unknown frequency present in the data file. Clearly state the frequency indicated as your “best” guess from the outputs of (2).
4. (10pts) Refine your “best” guess answer for the unknown frequency by changing your MATLAB code. Indicate your improved guess (different than that from (3)). Explain how you arrived at your improved solution.

### ECGR 4124/5124 Task 3

We now will take your “best” guess from Task 2 to process video image data as if each image pixel were a single time-domain function. For example, let a hypothetical camera take 10x10 pixel videos. Each image will have 100 pixels and each time a new image is captured the pixel value represents a digital sample from the analog world. Let  $x_i[n]$  denote the signal for image pixel  $i$  at sample index  $n$ . For our 10x10 images this creates 100 signals  $x_1[n]$  (the top left corner pixel) to  $x_{100}[n]$  (the bottom right corner pixel). We will then assume that a signal having your “best guess” frequency exists within every one of these 100 signals and we want to “magnify” the presence of this normally undetectable signal. To do so we proceed as follows:

1. Read in the video data file and convert it to a list of signals, if the image has N columns and M rows we convert this into a list of NM signals  $x_1[n], x_2[n], \dots, x_{NM}[n]$ .
2. Cross-correlate your filter with each of the input signals creating output signals  $y_1[n], y_2[n], \dots, y_{NM}[n]$ .

At this point we could try and look at each of these signals for the unknown frequency but for even small images this is too much data to analyze. Instead we will try re-incorporate the magnified frequency into the original video and view the video to see the output.

1. Multiply your output signals  $y_1[n], y_2[n], \dots, y_{NM}[n]$  by a constant  $\alpha$  and add it to the original input signal creating a video output signal  $v_1[n], v_2[n], \dots, v_{NM}[n]$  where  $v_i[n] = x_i[n] + \alpha y_i[n]$ .
2. Use the provided MATLAB code to view the output video signal.

### ECGR 4124/5124 Task 3 (30 pts)

#### To Turn In:

Turn in your MATLAB code. NO MATLAB CODE IMPLIES NO CREDIT FOR ANY OF YOUR SOLUTIONS.

1. (5pts) Plot the cross-correlation between the red pixel signal and your chosen filter.
2. (5pts) Pick off sample indices corresponding to maximum and minimum locations in the cross-correlation plot. Use the data cursor to pick off these locations and show the annotated plot indicating the  $(x, y)$  locations of the curve max and min that you selected.
3. (10pts) Create a 2x2 matrix of image plots. The first column will show the unmodified original image and the “magnified” image using the maximum and minimum index values you selected from (2) above. The second column will show the “magnified” image using the maximum and minimum index values you selected from (2) above. Label each image in your 2x2 matrix as ‘original image’, ‘max image’, and ‘min image’ as appropriate. Hint: *If  $h[n]$  is a row vector then you can call `conv2(A,B,'same')` (2-dimensional convolution) as part of your code to cross-correlate  $h[n]$  with all pixels of the input image in one function call.*
4. (5pts) What physical effect is magnified by your filter within the image? Explain why this effect has become visible whereas it was not previously observable in the original image. You must explain why other phenomenon in the image are not magnified and how the magnification is specific to only the “magnified” phenomenon.
5. (5pts) What is the heart rate of the subject in the video (in beats per minute; bpm)? How can you compute this from the magnified video images. Explain your calculation relating the digital computation back to the analog value of the heart rate.

### ECGR 5124 Task 4

We wish to design a heart rate monitor. To do so, we are provided an electrocardiogram of a patient as an input function  $x[n]$ . From this signal, we wish to estimate the periodicity of the heart beat signal in beats per minute. We approximate the heartbeat signal as the sum of several Gaussian pulses. Using these pulses, we can construct the expected shape of the heartbeat signal, i.e., the template, as follows :

- the number of Gaussian pulses in the template,  $N_g$ .
- the mean,  $\mu_i$ , for each pulse.
- the standard deviation,  $\sigma_i$ , for each pulse.
- the peak amplitude of each Gaussian pulse,  $A_i$ , for each pulse.

$$g_i(\mu_i, \sigma_i, A_i) = A_i * \frac{1}{\sigma_i \sqrt{2\pi}} e^{-\frac{1}{2\sigma_i^2}(t-\mu_i)^2}$$

$$T_g = \sum_{i=1}^{N_g} g_i(\mu_i, \sigma_i, A_i)$$

Note that the subscript  $i$  denotes the  $i^{th}$  pulse,  $i = 1, 2, \dots, N_g$ .

1. You are provided with an input signal  $x[n]$  sampled at a rate of  $f_s = 40$  samples/sec in the form of a datafile : `project1_data.mat`. This signal contains a periodic sequence of Gaussian pulses of the form specified above where  $N_g = 3$ ,  $\mu = [0.1, 0.2, 0.23]$ ,  $\sigma^2 = [0.005, 0.006, 0.01]$ ,  $A = [2, -2, 1]$ .

(i) Write a function to compute the coefficients of a template with 41 values by sampling  $T_g$  at the same frequency as  $x[n]$ , i.e.,  $n = [0, \dots, 40]$ . Using a MATLAB function call to construct each Gaussian pulse, a function template has been provided in the file: *construct\_pulse.m*. Assemble the template function,  $t[n]$ , by summing the  $N_g$  constructed Gaussian pulses. Plot the resulting template function using `stem()` in MATLAB. Convert this template into a digital filter by time reversing the template. Plot the filter coefficients of your matched filter,  $h[n]$ , in a second plot.

(ii-iii) Detect the pulse signals within the data signal and generate a list containing the sample lag indices of the detected pulses. Use these indices to estimate the period of the noise-perturbed signal,  $\hat{N}$ . (*Hint* :  $N > 5 \text{ samples}$ ). The detection and estimation scheme which one may use is as follows :

1. Find the maximum correlation value.
2. Find all sample indices where the signal value, i.e., find all  $n$  such that  $y[n] \geq 0.8 \max(y[n])$  where  $y[n] = x[n] * h[-n]$ . These are indices of correlation values  $\geq 80\%$  of the maximum observed correlation value. We consider the resulting set of indices,  $K = \{k_1, k_2, \dots\}$ , as possible lag indices where the template and the embedded pulse signal have aligned.
3. Compute the period estimates for each of the resulting index pairs by subtracting neighboring indices from  $K$ . Here we assume that indices with large correlation values indicate the presence of the template in the signal. Differences between these indices indicates the period of the unknown signal (in samples).
4. Use the hint to discard some invalid values for  $N$ .
5. Find the fundamental period from the remaining values, i.e., the smallest value.
6. Take all values within 30% of the fundamental period. Use their mean value as the estimate of the period,  $\hat{N}$ .
7. Use  $\hat{N}$  and the sample rate to compute the heart rate of your patient in *beats per minute*.

Feel free to design your own detection scheme. Any reasonable estimation scheme is acceptable.

(iv) Vary the length of your computed matched filter by taking the first  $N_k$  values from the template  $t[n]$  where  $N_k = [4, 10, 41]$ . Run your program from (ii-iii). Take your period estimates from step 6 and compute the standard deviation of the period estimates as a function of the length of the template. Make a comment on your observations.

**Input files are available for download at the ECGR4124/5124 website.**

#### **ECGR 5124 Task 4 (100 pts)**

##### **To turn in:**

You are responsible for turning in the following objects IN THE ORDER AS LISTED BELOW. **Deductions will be made for reports which are messy and difficult to comprehend.**

Turn in your MATLAB code. NO MATLAB CODE IMPLIES NO CREDIT FOR ANY OF YOUR SOLUTIONS.

1. (30pts) Plot of your matched filter and template from (i).
2. (30pts) Period estimates and plots as requested in (ii),(iii),(iv) and related comments and observations. Make sure to provide the estimated heart rate in beats per minute.
3. (20pts) Plot of the standard deviation of the period as a function of the template length from (iv) and the associated mean estimates of the period for different length templates.
4. (20pts) Your programs to solve (i)-(iv).