

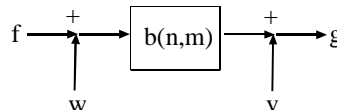
Take-Home Exam #2

- You may have 24 hours to work on this exam. You must enter your name and sign-out and sign-in times on the log sheet in BIRB 1089. There is a 1 hour grace period.
 - You may use your books, your own notes, web resources, computers and calculators.
 - You may not seek assistance, share or borrow notes from any current and former students or any other individual (instructors' notes available on the web are OK).
 - This exam is governed by the Engineering Honor Code, which requires that you do not seek assistance on this exam and that you report any violations of the Honor Code. (For more info see <http://www.engin.umich.edu/org/ehc/index.html>).
 - On the first page of your solutions, write "I have neither given nor received aid on this exam," and sign your name below it.
 - Show your work and hand in all Matlab code.
 - If parts of the exam are not clear and you cannot find the instructor to ask for clarification, please write your assumptions and proceed with the question. If you cannot solve a part that is needed later in the solution, define a parameter to represent the answer of that part and continue.
 - Write legibly (yes I know I don't, but...)
 - Please remember to put your name on your exam solutions.
 - Good luck!
1. In this problem you will investigate subpixel edge detection using interpolation methods we discussed in class. Matlab template `ex2p1_template.m` creates two images `xmini` and `xfull`, reduced (64x64) and full resolution (320x320) images, respectively.
- a. Implement a convolution interpolation method to increase the sampling density of `xmini` to that of `xfull`. Your solution must use the `conv2` function. Implement nearest neighbor, bi-linear and bi-cubic interpolation. For cubic interpolation, use this

$$\text{kernel: } h(x) = \begin{cases} 1 - 2.5x^2 + 1.5|x|^3, & |x| \leq 1 \\ 2 - 4|x| + 2.5x^2 - 0.5|x|^3, & 1 < |x| \leq 2. \\ 0 & \textit{otherwise} \end{cases}$$
- Display the resultant images (and `xfull`) using `imagesc`.
- b. Compare the different interpolation methods in terms of Mean Square Error (MSE) relative to `xfull` and in terms of computation time. For MSE calculations, be careful of shifts of your interpolated image relative to `xfull`. Which method is best? Comment on whether or not this is what was expected and why.
 - c. Apply a Sobel gradient-based edge detection algorithm to the interpolated images and `xfull`. You may need to make small modifications to the thresholding levels and edge thinning algorithms that were used in HW #6. Do not use the Matlab `edge` function and use exactly the same edge detection parameters/algorithms for each interpolated image. Display the resultant edge maps. Which method is best?
 - d. Using the `find` command applied to row number 152 (e.g. `find(edgemap(152, :))`) of the edge maps, determine the width in number of pixels of the object of for `xfull` and interpolated images. Compare to the theoretical width based on code in the template.

If there are too many edges, you may need to look at the gradient maps to determine the true edge.

2. Wiener filtering using empirically derived values. We will use the picture of the house again (hw7image.mat). Download `ex2p2_template.m`. This template creates a noisy version of the image. You will derive a linear space invariant Wiener filter for this image and blocked (space variant) Wiener filter.
 - a. We will use a signal model with an autocorrelation function $R_s(n, m) = \sigma_s^2 a \sqrt{n^2 + m^2}$. Give an expression for the autocorrelation function of the noisy image $y(n, m) = s(n, m) + w(n, m)$.
 - b. Choose a background (e.g. sky) area and estimate the noise variance $\hat{\sigma}_w^2$. Is the close to the theoretical variance derived from the code?
 - c. Since the image is not zero mean, we will want to use the affine form of the Wiener filter. After eliminating the mean, find estimates of $\hat{R}_y(0,0)$, $\hat{R}_y(1,0)$, and $\hat{R}_y(0,1)$. From these derive an estimate of \hat{a} and $\hat{\sigma}_s^2$, and thus $\hat{R}_s(n, m)$.
 - d. From your empirical estimates, derive the L.S.I. Wiener filter and apply to the image. Determine the MSE of the original noisy image and the filtered image relative to noise-free image.
 - e. Now, we will do similar filtering on 48x48 sized blocks (a total of 150). Repeat steps c. and d. for all blocks. Determine the MSE of the filtered image relative to noise-free image.
 - f. Comment on which approach is better. How might you improve on the results using the method of part e.?
3. Noise and deblurring. Consider the following system with two additive noise sources and a blurring function $b(n, m)$:



- a. Derive the L.S.I. filter, $H(\mathbf{w}_x, \mathbf{w}_y)$, that you would apply to $g(n, m)$ to produce an estimate of f , $\hat{f}(n, m)$ that minimizes the MSE. Assume that w and v are independent white noise processes with variances σ_w^2 and σ_v^2 , respectively. Also assume that they are independent of f .
- b. Now consider a case where w and v come from the following functions:

$$w(n, m) = A \cdot z_1(n, m)$$

$$v(n, m) = z_1(n, m) ** a(n, m) + B \cdot z_2(n, m)$$
 where z_1 and z_2 are independent, white, zero-mean processes, A and B are constants and a is some convolution function. Derive the autocorrelation functions, power spectra, cross correlation functions, and cross power spectral functions: $R_w, R_v, R_{wv}, P_w, P_v, P_{wv}$. Derive, also, expressions for the variances σ_w^2 and σ_v^2 .
- c. Repeat part a. for the new descriptions of w and v .