Take-Home Final Exam

- You may have 72 hours to work on this exam. You must enter your name and sign-out and sign-in times on the log sheet in BIRB 1088.
- 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.
- Please remember to put your name on your exam solutions.
- Good luck!

1. [25] Weiner denoising, deblurring. Use the template finalp1_template.m, and add:

```
vv = conv2(vv,ones([3 1]),'same');     % color the noise
blurff = conv2(ff,ones([1 3]),'same'); % the blurred signal
gg = blurff + vv;                      % the measurement
g1 = gg(:);                            % make image a vector
```

You will find the MMSE estimate of ff from gg by two different methods.

a. Use the statistical resoration methods with column stacked data g1. We must first determine the blur matrix, **H**. Show, based on the superposition principle, that the elements of **H** are $h(n,m;k,l)$. Specifically, argue that unstacking (reshaping) of the $(kN+l)^{th}$ column of **H** corresponds to the image $h(n,m;k,l)$, e.g. the blurring system output for a delta function at location (k,l).

b. Calculate **H** in Matlab.

c. Determine the covariance matrices $\mathbf{K_v}$ and $\mathbf{K_f}$. If you are having difficulty building $\mathbf{K_v}$ and wish to continue with this problem, you may delete the matlab line that "colors the noise" (3 pt deduction).

d. Determine the MMSE estimate of ff from gg using the results of parts b and c. Display the results using imagesc and determine the MSE before and after filtering.

e. Determine the blur spectrum $B(w_x, w_y)$.

f. Determine the power spectra $P_v(w_x, w_y)$ and $P_f(w_x, w_y)$.

g. Determine the MMSE estimate of ff from gg using the results of parts e and f. Display the results using imagesc and determine the MSE after filtering. How does this result compare with part d?

h. Alter the variable nsd from 1.0 to 0.1 and redo the MMSE estimates (recalculate what ever parameters are necessary to yield the proper estimate). Compare the two MMSE estimate and explain any differences.

2. [40] Pyramid coding, interpolation, codewords, etc. Use the template finalp2_template.m, in which two stages of pyramid coding decomposition is implemented. In this template, f_0 is the quantity we wish to code and e_0, e_1, and f_2 are the quantities to be coded and quantized. In our particular example, we will assign zero bits for coding of e_1.
   a. Interpolate (e.g. upsample) f_2 to become $\hat{f}_{0,l}$ using linear interpolation.
   b. Determine the error image, e_0, required to make $\hat{f}_{0,l}$ equal to f_0. [If you are unable to solve part a., let $e_0 = (f_0 - f_{0,l})$ so that you can do the following parts.]
   c. We will now quantize and code f_2 and e_0. Suppose we want an average of 1.5 bits/pixel for coding f_0. We choose to use scalar quantization with 8 reconstruction levels in f_2 and 5 levels in e_0. Is this a reasonable scheme? Will we come close to our target bit rate?
   d. Perform an 8 level uniform scalar quantization on f_2 between min(f_2) and max(f_2).
   e. Determine the frequency of each reconstruction level and derive an optimal variable length code for these 8 reconstruction levels. What is the average bit rate for the quantized version of f_2: $\hat{f}_2$?
   f. Estimate the entropy of the image variable resulting from the quantization in d. How close to the bit rate for our code compare to the entropy? Is the variable length code here substantially better than a fixed length code?
   g. Perform parts a. and b. using $\hat{f}_2$. [If you are unable to solve part a., find e_0 based on the uniformly quantized $f_{0,l}$ so that you can do the following parts.]
   h. Perform a 5 level uniform scalar quantization on the e_0 from part g. between –max(abs(e_0)) and +max(abs(e_0)).
   i. Determine the frequency of each reconstruction level and derive an optimal variable length code for these 5 reconstruction levels. What is the average bit rate for the quantized version of e_0: $\hat{e}_0$?
   j. Deterimine the final coded image using $\hat{f}_2$ and $\hat{e}_0$. What is $D = E\left[\left|\hat{f}_0 - f_0\right|^2\right]$? What is the overall bit rate? Do we achieve our target bit rate of 1.5 bits/pixel? How does this method compare to VQ and PCM methods of HW#8?
   k. Compare $MSE_0 = E\left[\left|\hat{e}_0 - e_0\right|^2\right]$ with D. Comment on the relationship between $MSE_2 = E\left[\left|\hat{f}_2 - f_2\right|^2\right]$ and D. Comment on ways we might change the scheme in part c. to further reduce distortion. Does the usual bit allocatoin scheme (based on variance) make sense here?

3. [15] Let $f(x,y) = f(\mathbf{x})$, where $\mathbf{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$ and let $\mathbf{x}' = \mathbf{T}(q,a,b)\mathbf{x} = \begin{bmatrix} \cos q & \sin q & a \\ -\sin q & \cos q & b \\ 0 & 0 & 1 \end{bmatrix}\mathbf{x}$ be a coordinate tranformation. Let $g(\mathbf{x}) = f(\mathbf{T}(q,a,b)\mathbf{x})$.
   a. Determine $G(u,v)$ in terms of $F(u,v)$.
   b. Argue that $f(\mathbf{x})$ be written in this form: $f(\mathbf{x}) = g(\mathbf{T}(q',a',b')\mathbf{x})$. Determine q', a', b' for q = 45 degrees, a = 0.2, and b = -0.3. Does $q' = -q$, $a' = -a$, $b' = -b$? Why or why not?

c. Now, let f and g both be sampled with sample spacing $\Delta x = \Delta y = d$. Describe $G_s(u,v)$ and $F_s(u,v)$ in terms of $F(u,v)$.

d. Describe the class of band limited signals for $f(x,y)$ for which it is possible to recover $f(x,y)$ from $g_s(x,y)$. For this part, let q = 45 degrees, a = 0.2, and b = -0.3.

4. [20] Wavelet transforms. Consider the length-4 1D Haar transform:

$$\mathbf{T}{=}\mathbf{Ax}\text{, where} \qquad \mathbf{A} = Haar_4 = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} a_d(0) \\ a_d(1) \\ a_d(2) \\ a_d(3) \end{bmatrix}$$

where the a_d is a row of $\mathbf{A}$ and the subcript d indicates direction, either x or y.

a. First, consider the 1D case. Is this transform energy preserving? Are the rows orthogonol? Determine the inverse transform $\mathbf{B} = \mathbf{A}^{-1}$.

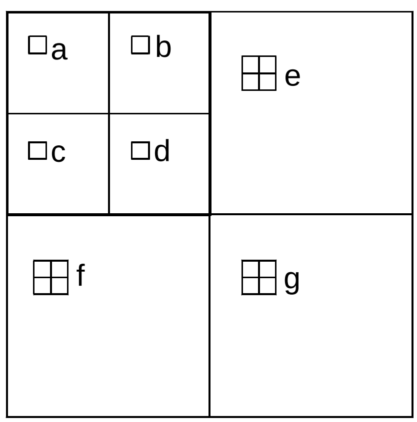
b. Using the image from HW#7, execute the following commands:

```
[a b] = size(hw7image);
h1 = haar2(hw7image);
h1(1:a/2,1:b/2) = haar2(h1(1:a/2,1:b/2));
```

`haar2.m` is on the website. This produces two levels of a 2D wavelet decomposition.

c. Is this tranformation energy preserving? If not, modify `haar2.m` to make it energy perserving.

d. Part b. produces an image in the following form:



Some 4x4 region of the image maps to 16 pixels denoted by letters a-g. Determine which basis functions from $Haar_4$ contibute to each of the pixels or pixel groups. For example, a hypothetical region may represent tranform coefficient associated with $a_x(3)$ and $a_y(0)$. If some pixels cannot be represented by the x and y $Haar_4$ bases, then describe what other length-4 basis functions are present. Does part b. implement a 4x4 2D Haar transform?

e. Are the coefficients that make up this tranformation orthogonal? (Another way of say this is to ask if the input 4x4 block is white noise, will there be correlation between any of the pixels marked by a-g?)

f. Invert this tranformation (create `invhaar2.m`) to yield the original image of the house.