Homework 9

Submission instructions.

- Submissions are due on Thursday 11/07 at 7.00pm ET == 4.00pm PT
- Please upload scans of your solution in GradeScope (via Canvas)
- Please ensure that your scans are readable.

Instructions

- Please show all necessary steps to get the final answer. However, there is no need to be overly elaborate. Crisp and complete answers.
- For all MATLAB problems, include all code written to generate solutions.
- Please post all questions on the discussion board on the Piazza course website, rather than email-ing the course staff. This will allow other students with the same question to see the response and any ensuing discussion.
- **If you feel some information is missing, you are welcome to make reasonable assumptions and proceed. Sometimes the omissions are intentional. Needless to say, only reasonable assumptions will be accepted.**

1. (**9.9 pts**) *(Iterative Hard Thresholding)*

   Implement iterative hard thresholding in MATLAB under wavelet sparsity prior.

   Your function for iterative hard thresholding that solves the problem

   $$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - A(\mathbf{x})\|_2^2 \qquad \text{s.t} \qquad \|\Psi(\mathbf{x})\|_0 \leq K,$$

   might look like this

   ```
   function xstar = IHT(y, K, Psi, PsiAdj, A, AAdj, MaxIter, xinit)
   ```

   Here `xinit` is an initial guess of the solution. You can use the zero input as the guess.

   Deliverable 1. Matlab function for IHT. Please label the code for readability.

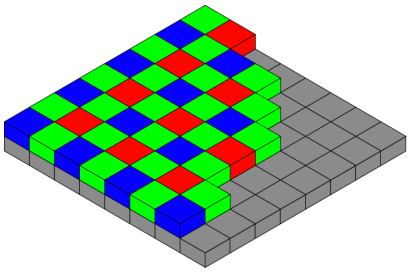
   We will test the code on three problems. Please look in `hw9.mat` for the MATLAB file that has the companion files.

   Problem 1. Non-blind deblurring. You are given a blurred grayscale image in the variable `blurred` and a blur kernel `k0`. Deblur it.

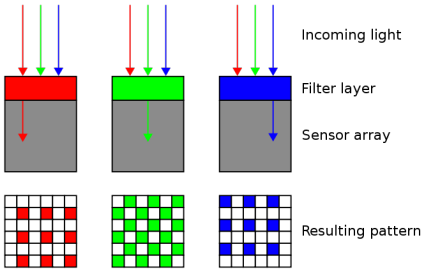
Problem 2. Non-blind inpainting. You are given a grayscale image in the variable `corrupted` and a mask image `mask` that indicate where the corruptions are. Inpaint with IHT.

Problem 3. Demosaicking. For the last problem, we are going to do something different. Color images are sensed by using a color filter array on the sensor. So each pixel only senses one among the red, blue and green colors of the true 3-color image. So given a unknown 3-color image I that has $N \times N \times 3$ pixels, the measured image Y has only $N \times N$ pixels such that

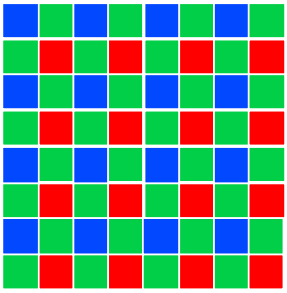
$$Y[m,n] = \begin{cases} I[m,n,3] & m,n \text{ are both odd} \\ I[m,n,1] & m,n \text{ are both even} \\ I[m,n,2] & \text{otherwise} \end{cases}$$



(a) The Bayer arrangement of color filters on the pixel array of an image sensor



(b) Profile/cross-section of sensor



(c) Color code for pixels on the sensor

Demosaik the image in the variable `mosaik`. Figure out a strategy how to handle color data with IHT. =)

Deliverable 2. For all THREE problems, provide MATLAB code for implementing the operator A and its adjoint A^*.

Deliverable 3. Describe your strategy for handling the color in the demosaicking problem.

Deliverable 4. Restore image for each problem using your IHT code. Do sweep through values of K for best results. In your submission you are expected to show the restored image for each of the three problems.

2. **(0.1 points)** How many hours did this homework take?