Finals

Due date: Tuesday, 12/12/17 at 6pm EST == 3pm PST.

Notes on Problem Set. There are SIX problems. Please answer any FOUR. All problems have the same number of points. If you answer more than four, then we will simply use the top four scoring problems.

Submission instructions. (1) All submissions are via email to Aswin at saswin@andrew.cmu.edu . (2) Please send all your files inside a SINGLE zip file named as "lastname.firstname.zip" (3) If you dont have a scanner handy, there are a number of effective cellphone scanning app that you could use. (4) You are given more than 2 days to solve this problem set. Use the time wisely. You should not need more than 3 hours. And do not miss the submission deadline.

Late submissions. lized at 3% per minute. So, any submi time-stamp on the email a

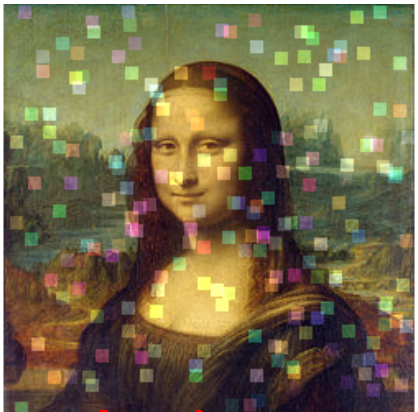
Discussions and exam integrity. **No discussions are p our work should be entirely your own. Failure to abide by this cheating/unauthorized assistance.** If you refer to m as, please cite them. Please read up on CMU policy on academic integrity at **this link**.

Missing information, bugs etc. We have proof read the problem set multiple times to ensure there are no bugs or missing information. Some details are however left out intentionally and are for you to figure out. So, if you really feel some piece of information is missing and need to make an assumption to solve the problem — please go ahead; there is no need to run it by the instructors. Do not forget to mention the assumption that you needed to make to solve the problem.

In spite of above, if you still have any questions, pl email Aswin. However, any answer, should we decide to reply, will be made public to all students.

Software packages: You are expected to code your own solutions. In particular, you are welcome to reuse any code developed during the semester by you or us — but no software outside of that. Ok to use any inbuilt MATLAB command.

Q1 [Image restoration] Restore these images. (Please use the images in Q1.zip)



Deliverable: A res
restored image. A brief par
expected to detail logic/intuition behind denoiser, modeling ation
technique used.

Note #1: Evaluation for your submission will take into a
restored images as well as the restoration approaches employed.

Note #2: The noise / corruption present in each image is different. Understanding the right noise model is key to solving this problem accurately. Your submission should detail how you tuned/chose the restoration algorithm to specifics of the corruption.

Note #3: Feel free to restore each image using a different approach.

Q2 [Line detection and radon transform] Devise a scheme using the radon transform to find the exact locations of the 2 lines in this image (use the image in Q2.mat).
Deliverable # 1. Explain the steps of your methods and why you choose to do so
Deliverable # 2. Estimate and report the parameters (slope and intersept) of the lines.

Well explained methods will receive credits and "I measured it with a ruler" will not score you any points.

Figure 1: Noisy image of 2 lin

Hint: You might find the MATLAB function `findpeaks` helpful.

Q3 [Hyperspectral image denoising] Hyper-spectral (HS) capture both the spatial and spectral variations in a scene. In this problem, we are going to look at using group-sparsity to denoise HS images.

Let $\mathbf{x}_\lambda$ be a $n \times n$-image (Let $N = n^2$) correspond ed by λ. Each $\mathbf{x}_\lambda$ is, like all images, compressible in a wavelet basis. Ho can expect the support to be similar across images corresponding to different wavebands, i.e, we can expect $\mathbf{x}_{\lambda_i}$ and $\mathbf{x}_{\lambda_j}$ to have similar support in a wavelet basis.

Suppose now we can a noise HS image such that the noisy measurement $\mathbf{y}_k \in \mathbb{R}^N$ of the form

$$\mathbf{y}_k = \mathbf{x}_{\lambda_k} + \mathbf{e}_k.$$

Clearly, recovering all the spectral channels simultaneously enforcing similar support patterns should help!

Deliverable #1: Formulate an algorithm for recovering the HS cube given noisy measurements that enforces not just individual wavelet sparsity but joint wavelet sparsity.

`Q3.mat` has a test hyper-spectral cube in the variable `HSdata`. The first two dimensions correspond to spatial variation and the third dimension correspond to spectral variation. Now, simulate noisy measurements by adding random gaussian noise. If X is the noise free HS image and $\widehat{X}$ is the noisy measurement, then we measure the amount of noise added using the Input SNR (in dB) defined as

$$\text{Input SNR}(\widehat{X}) = 20 \log_{10} \left(\frac{\|\text{vec}(X)\|_2}{\|\text{vec}(X - \widehat{X})\|_2} \right),$$

where vec($\cdot$) is vectorizes the HS image. Recover the HS data from these measurements using two recovery algorithms: (1) A Naive recovery that assumes wavelet sparsity on each spectral image but NO joint sparsity model; and (2) the solution to the formulation in deliverable 1 that enforces joint support across spectral bands.

Deliverable #2: Plot reconstruction performance in terms of SNR (in dB) for each algorithm for input noise SNR taking values in {0 dB, 10 dB, 20 dB, 30 dB, 40 db, 50 dB }.

Q4 [Compressive recovery] Let $\mathbf{X}_0$ be a 64×64 pixel natural image. We vectorize X_0 to obtain $\mathbf{x}_0 \in \mathbb{R}^{4096}$.

Given in `Q4.mat` are measurements $\mathbf{y} \in \mathbb{R}^{2048}$ and a measurement matrix A of size 2048×4096 such that

$$\mathbf{y} = A\mathbf{x}_0 + \mathbf{n}.$$

This is data from a single pixel camera. We do not know much about the noise $\mathbf{n}$ except that its ℓ_2-norm is bounded.

Deliverable: Estimate $\mathbf{x}_0$ and describe in detail the procedure adopted to estimate it. You are expected to provide the reconstructed image as a .png file.

Hint: Once you estimate $\mathbf{x}_0$, you can get a 2D image by using the command

```
>> X0 = reshape(x0, 64 64);
```

Q5 [RIP] Let A be an $M \times N$ matrix that satisfies RIP of order $2K$ with constant δ_{2K}, i.e.,

$$\forall \|\mathbf{x}\|_0 \leq 2K, (1-\delta_{2K})\|\mathbf{x}\|_2^2 \leq \|A\mathbf{x}\|_2^2 \leq (1+\delta_{2K})\|\mathbf{x}\|_2^2.$$

Let $\mathbf{a}$ and $\mathbf{b}$ be K-sparse vectors with disjoint support, i.e.,

$$\|\mathbf{a}\|_0 \leq K, \|\mathbf{b}\|_0 \leq K, \text{ and } \mathrm{supp}(\mathbf{a}) \cap \mathrm{supp}(\mathbf{b}) = \phi.$$

Show that

$$\frac{|\langle A\mathbf{a}, A\mathbf{b}\rangle|}{\|\mathbf{a}\|_2\|\mathbf{b}\|_2} \leq \delta_{2K}.$$

In simple words, what this suggests is that sparse vectors that are orthogonal remain nearly-orthogonal after the operation of A, provided it satisfies RIP.

Q6 [Computational imaging] Let $x[n]$ be a 1D signal of length 256 pixels and $h[n]$ be a 1D blur kernel of length 11 pixels. We will assume that $h[n] \in \{0, 1\}$, i.e., $h[n]$ is binary valued.

We obtain measurement $y[n]$ of length 266 (= 256+11-1) of the form

$$y[n] = (x * h)[n] + e[n] = \sum_{k=0}^{10} h[k]x[n-k] + e[n],$$

where $*$ is linear convolution and $e[n]$ is additive white noise with mean zero and variance σ^2.

Design a kernel $h[n]$ that is optimal for recovering $x[n]$ from $y[n]$ and justify its optimality/design. Please clearly state the criteria under which your kernel is optimal. (You are welcome to use MATLAB).