Math 473 HW 6

Let b be the incomplete Fourier data related to the underlying image x by the model is $b = Ax + \epsilon = PFx + \epsilon$ with F the 2D discrete Fourier transform operator, P a selection matrix, ϵ a random Gaussian noise. Solve the inverse problem of finding x from b using wavelet compressive sensing reconstruction model: $\min_y \frac{1}{2}\|PFx - b\|_2^2 + \alpha\|Wx\|_1$ with W denoting a unitary Haar wavelet transform operator. You may use ADMM or primal-dual algorithm (preferred) to implement. Describe and derive the numerical algorithms in details and submit. You will run the code on two kinds of data simulated below:

Data simulation. Download mask creation code MRImask.m (courtesy of Wotao Yin, UCLA). Start with a clean 256×256 Shepp-Logan phantom image $X = phantom(256)$ and simulate incomplete Fourier data by first computing $fft2(X)$, and then dot multiplying it by a 0-1 mask. Make sure the mask and the Fourier data match well, especially the locations of low frequency align correctly. Two examples of masks, one radial and one random, are simulated.

1. Simulate a radial sampling mask using MRImask.m with Ls radial lines and noise standard deviation σ. Note that MRImask.m creates a mask that samples more in the center. It assumes that the center corresponds to low frequency. You need to use fftshift before applying this mask. Here is how to use it:

```
n = size(X,1); % X is a square matrix with dimension nxn.
Ls = 50; sigma = 0.05;
SpMsk= fftshift(MRImask(n,Ls));
idx = find(SpMsk~=0);
len = nnz(SpMsk); % number of nonzeros in SpMsk;
K = fft2(X).*SpMsk; % simulate partial Fourier data K
% add complex valued noise to the locations with data sampled
K(idx) = K(idx)+sigma*(randn(len,1)+sqrt(-1)*randn(len,1));
```

2. Simulate a random sampling mask M that has value 1 inside a small square (say 20×20) around the center of the image domain, and randomly choose $c\%$ elsewhere to be 1 where $c = 20, 10, 5$. Make sure to fftshift this mask M before dot product with Fourier (k-space) data.

Note that the simulated data K has the same size with the original image X. It has 0 values at non-sampled frequency. Let b be the vector version of the sampled data, then it should collects K at sampled locations.

You need Matlab wavelet toolbox to do the wavelet computation. Let me know if you don't have access to Matlab wavelet toolbox. Following shows how to use Matlab's wavelet transform and inverse transform. Download Wavedb1Phi.m.

```
% Psi is a function handle that computes
% discrete wavelet transform (Haar).

Psi = @(x) Wavedb1Phi(x,0);
```

```
% PsiT is a function handle that computes
% inverse discrete wavelet transform (Haar).

PsiT = @(x) Wavedb1Phi(x,1);
```

Parameter selection. The parameter α, depends on the number of measurements, the value of σ (noise standard deviation) as well as the sparsity of x under the Haar wavelet transform. It needs to be tuned. Try several α values (e.g., $\alpha = 10^d, d = -2, -1, 0, 1, 2$). List a table to compare SNR and relative error as α varies. Choose and display the result corresponding to the largest SNR or the smallest relative error.