Assignment VI

Biomedical Imaging & Analysis (ECE J1-791) - Fall 2014

Sun Yat-sen University – Carnegie Mellon University (SYSU-CMU)
SYSU-CMU Joint Institute of Engineering
Instructor: Prahlad G Menon, PhD

Instructions

Please show your solutions to each problem in full, writing explanations neatly along with mathematical justifications, as needed. For computer programs, please remember to turn in your code through the course's blackboard session, as well as any plots / figures that are requested. This assignment is due on **Thursday, 30 Oct 2014** via Blackboard, including an a Report with explanations associated with each question in the assignment, as well as any associated code and result files. If you have collaborated with another student on solving this homework assignment please state so (e.g. "I helped John with question 1" etc.).

LEARNING GOALS:

- *Projections and K-Space:*
  - K-space data of Image Projections.
  - Radon Space v/s K-space and the Fourier Transform.

1. **(40 points, total) Verify what 'back-projection' means by performing a 'virtual CT scan':**
   a. **(20 points)** Start with the 2D k-space of the Shepp-Logan Phantom from your K-Space Assignment #3 to extract 'projection lines' across the center of k-space and populate a matrix r(k, :) where 'k' represents the rotational step number, starting from $-\pi$ to 0^0. The resulting 'r' matrix is the Fourier Transform of the 'sinogram' of image-space. Verify that your result makes sense by comparing the plots of the appropriately center-shifted inverse Fourier transform of your resulting matrix, 'r', against the sonogram generated from the Radon() function in Matlab (eg: R = radon(I,0:M);) applied onto the fully reconstructed image-space, I, of the phantom K-Space data.
   HINT: Instead of creating lines through k-space with different angles, you can rotate the k-space matrix and take lines through the center of k-space as 'columns' each time you rotate it.
   Eg: `RotatedkSpace = imrotate(kSpaceMatrix,-180/steps,'bicubic','crop');`

b. **(20 points)** Next, fill the blanks and perform 'literal back-projection' from your using the following function which does back-projection of each diagonal line through k-space, one by one. Try obtaining results using: i) 50, ii) 100 and iii) 180 rotation steps for the CT scanner.

```
function reco=BackProjection_Simple(r,steps)
reco=zeros(128); %Create 128 x 128 accumulator matrix in image space

for k=1:steps
   q=real( <inverse fourier> r(k, :) );
% SWEEPING BACK the projection line in real-space
   reco = reco + ones(length(q),1)*q;

% Instead of adding to lines with different angles, we rotate the
matrix and add to its columns
   reco=imrotate(reco,-180/steps,'bicubic','crop');

end
reco=imrotate(reco,180,'bicubic','crop');
```

Insert representative plots / images of the reconstructed image for each exercise and submit your code.

2. **(40 points) Using K-Space to directly recover f(x, y) from a set of M projections** $p_{\theta 1}$, $p_{\theta 2}$ … $p_{\theta M}$, created using the Radon transform of the full reconstructed image-space of the 2D Shepp-Logan phantom k-Space data given in Assignment 3:
   a. **(20 points)** This time, display the reconstruction of f(x, y) using M = 5, M = 10, and M = 50 (in each case evenly distributed between [0, π]) projections using ramp-filter, *filt*, to filter your back-projection to reduce the dominant effect of the center of k-space and additionally **write your own code to do this operation by appropriately creating an "accumulator matrix of k-space" of all the projections first before recovering the image-space using an inverse Fourier Transform.** Note how this is a different strategy than the method used in Q1! Please report screenshots of the k-space that you accumulate and the resulting images, in each case. Comment on the differences between the reconstructions using both filtered back projection and the unfiltered approach used in Q1.
   Again, write your own code to do this operation in k-space (including the filtering operation) and do not use the iRadon function in Matlab!
   b. **(10 points)** Now, attempt to use the iRadon function in Matlab to recover the projections with a ramp filter. Are these results different than those obtained from the approach in Q2 Part (a)..?
   c. **(10 points)** In the case of using 10 projections only, how does the iRadon result improve if you linearly interpolate the projections in Radon space by subdividing the sonogram 10 times before inverting

the Radon transform..? Explain what you see in the resulting reconstructed image space.

Insert representative plots for each exercise and submit your code.

3. **(20 points) Examining the effect of an artifact in a CT scan:**
Use the Matlab function 'phantom' to generate the image-space version of the Shepp-Logan phantom image: I = phantom(128) . Next, add a rectangular region of very high density to the center of the image i.e. a region of 40x40 pixels having intensity 100 times the mean intensity of the image. This circular image region would represent the effect of a 'metal fragment' – a high-density region – in your imaged sample. Now, use the Radon() function to create sinograms using M = 50, M = 100, and M = 180 projections, and finally generate reconstructions of the Radon space for each case using the iRadon() function in Matlab. Report your observations regarding the effect of the circular artifact on your reconstruction results in each case, with supporting snapshots of your results. Please submit your code, as well.