BioImaging CT Project

Submitted by: Sanket Dharwadkar [B00574254]

Introduction:

As learned from this project and through the course, computed tomography is basically X-ray images stacked together so as to get the depth information as the third dimension that constitutes a diagnostic image. The X-ray projection of the object at each angle of the CT gantry rotation produces a sinogram where the Y axis shows the angle in degrees while the X axis shows the spatial distance. For the following project, we have been given such a sinogram and we are required to construct its corresponding phantom object and determining the X-ray attenuation and possibly the object density throughout the object. During the project I worked and discussed with Nico Ciccheti and Madhura Shanbhag regarding the problems that we faced.

|  |  |
| --- | --- |
| C:\Users\sanket dharwadkar\Documents\MATLAB\CT Proj images\Figure 1.bmp  Figure 1. Disk phantom | C:\Users\sanket dharwadkar\Documents\MATLAB\CT Proj images\Figure 2.jpg  Figure 2. Disk phantom sinogram |

Figure 2 shows the sinogram which is achieved by grasping the attenuation of X-rays through the Disk Phantom (Figure 1) in the diagnostic scenario. In the code template provided, we have basically added the densities (shown as a function of greyscale) along the depth of the object thereby resulting in spatially discrete values of attenuation. We can see that the phantom has a very dark circle at its center and three other circles of varying grey scales and size around it. Since the phantom is constructed using a code and not using actual back-projections, we do not see any noise that is generally seen to been picked up by the X-ray detectors.

Question B: 0th moment

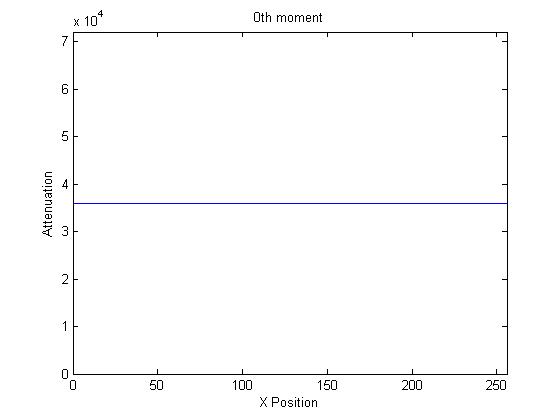


Figure 3. 0th moment of sinogram

The 0th moment, as shown in Figure 3 is the response of the CT detectors when X-rays first hit them at each angle. This would be the sum of attenuation amplitude (Y axis) at each angle (X axis) and therefore will be the sum of each of the 256 angular projected columns of the sinogram.

Comment: The striking result of it being a flat line is because of the fact that no matter which angle the projection is being taken from, the sum of the attenuation intensity will be constant since the object features are constant and not dynamic.

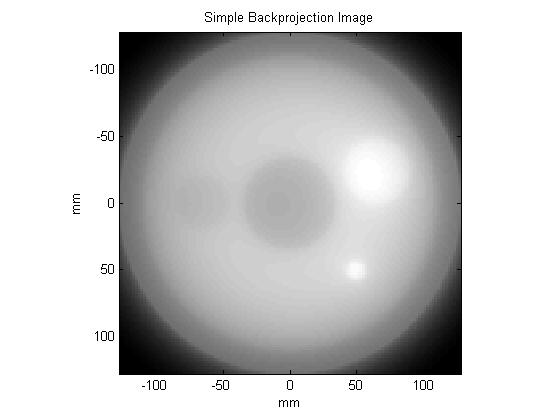


Figure 4. Simple back-projected image

Figure 4 shows a simple back-projection which is computed by overlaying projections on top of each other which create a concentration gradient for all the components of the image. A single column will contain the attenuation information for a single angular projection. There are 256 angular projections in total which correspond to the 180 degree shown on the sinogram. We therefore select one column at a time, smear the attenuation magnitude information over 128 rows and then rotate it to the angle which corresponds to in degrees (256th projection corresponds to 180o, so nth projection will correspond to n\*180/256 degrees).

Comments: The specific ‘greyness’ is added again and again every time a new projection smear is added to the old one. This makes the dark spots to show relatively dark and the light spots will show relatively light on the back-projected image. There is a background ‘haze’ that can be seen which adds a light shade to the entire image. This is because projections are smeared all over the respective lines and not only on the specific feature location. So for example, if a bright spot is located ‘x’ centimeters deep, the brightness is smeared all over the dimension line thereby making the entire line bright.

Question D1 and D2: Filter Sinogram and Filtered Back-Projections

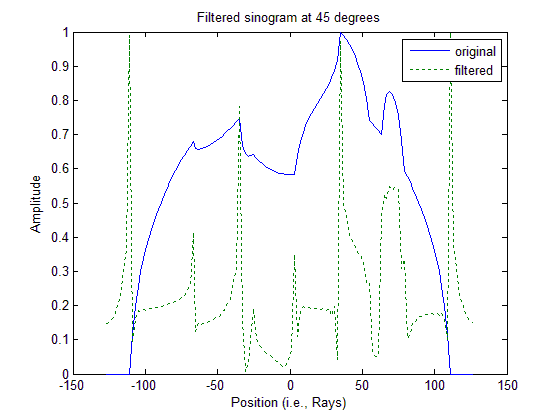


Figure 5. Filtered vs Original sinogram at 450

For carrying out filtered back-projection of the sinogram, we need to construct the filter that we will be using in frequency domain. This will help us easily multiply it to the Fourier transformed sinogram instead of convolution. The filter response is then multiplied with a Fourier transformed and shifted sinogram. The result to this is a Ram-Lak filtered sinogram, in frequency domain and all zero frequencies centered. To get the original frequency distribution, we inverse Fourier shift it and then inverse Fourier transform it to later get the spatial domain sinogram. Figure 5 shows a comparison of original sinogram projection and its filtered counterpart.

Comments: This is the step where I stumbled a lot with the code. I learned here that the dimensions of each and every matrix need to be same so as to carry out mathematical operations successfully. For that reason, I had to skip zero-padding because that was causing the sinogram to not match the ‘sinogramfilt’ dimensions. In the filter response at 45o we can see there are amplitude spikes when an edge is detected.

It was required to display the filtered sinogram at 45o for which provisions were already made in the template (256\*45/180 = 64th projection or column of the sinograms were selected to be displayed). We can see that the filter made tremendously significant changes in the amplitudes of projection attenuations (which are needed to remove the background haze and amplify feature appearance).

|  |  |
| --- | --- |
| C:\Users\sanket dharwadkar\Documents\MATLAB\CT Proj images\Figure 6.jpg  Figure 6. Filtered sinogram | C:\Users\sanket dharwadkar\Documents\MATLAB\CT Proj images\Figure 7.jpg  Figure 7. Filtered back-projected image |

Figure 6 shows the filtered sinogram that we use to back-project just like we did earlier, only this time I found a new back-projection result shown as Figure 7. The background haze is significantly reduced and the image looks very much similar to our original phantom.

Comments: As compared to our simple back-projected image, it is therefore seen that the Ram-Lak filter (high pass filter) has been able to remove low frequency noise (haze), improve contrast, thereby improving the total signal-to-noise ratio. The resolution also seems to have increased but mainly due to the increased sharpness and improved contrast.

Question E: Compare to MATLAB functions

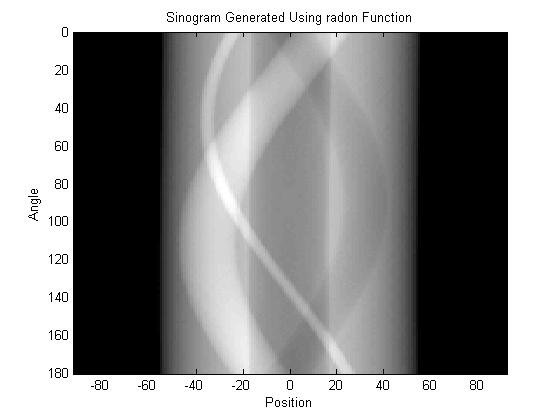


Figure 8. Sinogram generated using radon function

Using radon function on original phantom matrix to display sinogram, it is seen that we can skip all the sinogram generating template code.

|  |  |
| --- | --- |
| C:\Users\sanket dharwadkar\Documents\MATLAB\CT Proj images\Figure 9.jpg  Figure 9. Iradon using Ram-Lak | C:\Users\sanket dharwadkar\Documents\MATLAB\CT Proj images\Figure 10.jpg  Figure 10. Iradon using Hamming |

I used the iradon function with two filters: Ram-Lak and Hamming.

Comments: Ram-Lak filter being a high pass filter as compared to the mid-frequency pass Hamming filter, it was expected and also seen that the Ram-Lak filtered image (Figure 9) has sharper features than the Hamming filtered image (Figure 10). This is because high frequency signals (blocked by Hamming) are responsible for sharper edges and lower frequency signals (blocked by Ram-Lak) are responsible for smoothening image features. Therefore, to have better sharpness and better resolution we should use the Ram-Lak filter.

Question F: Subsampled sinogram back-projection

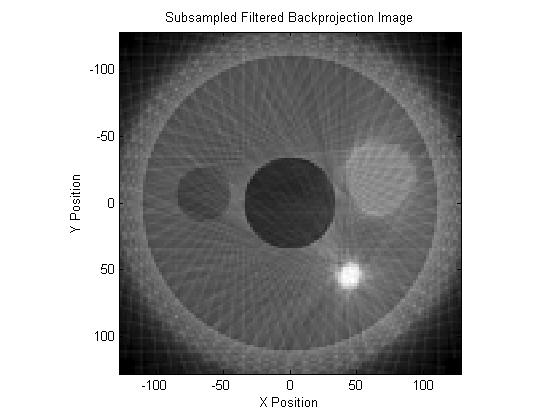


Figure 11. Subsampled filtered back-projection

Figure 11 shows the subsampled sinogram which gives us lower number of back-projections to work with. The resulting image is unable to provide us with the same amount of detail as the previous back-projection. There are also projection streak lines that are visible due to the superimposing of lower number of projections thereby worsening contrast, degrading SNR and resolution (more projections add greyscale gradient which improves contrast and also help with identifying edges of contained features).

Question G, H and I: Mystery object

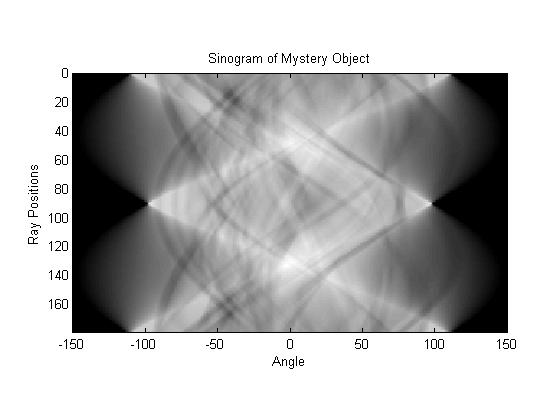


Figure 12. Mystery object sinogram

|  |  |
| --- | --- |
| C:\Users\sanket dharwadkar\Documents\MATLAB\CT Proj images\Figure 13.jpg  Figure 13. Simple back-projection of mystery object | C:\Users\sanket dharwadkar\Documents\MATLAB\CT Proj images\Figure 14.jpg  Figure 14. Filtered back-projection of mystery object |

It was almost impossible to guess the object from the sinogram (Figure 12), but it became clear from the simple back-projection (Figure 13) that it was the Binghamton Bearcat. This object had different dimensions so care had to be taken while setting the limits for the angles in the ‘for’ loop and also the resultant filtered image size in iradon function (Figure 14).

Comments: I struggled a few times while setting theta values (theta=0:180 was not accepted, then I realized 0:180 has 181 values and not 180!) In this case the number of angular projections is same as the number of angles in degrees, so conversion wasn’t really required while using ‘imrotate’. Because we are using the Ram-Lak filter, we are enhancing the high frequency edge components and removing the low frequency haze in the simple back-projection. This increases the mystery image’s contrast, resolution and SNR.

Conclusion:

We observe that radon and iradon functions are pretty powerful in constructing back-projections from a given sinogram. These functions go through the convolution of each given angular projection with a chosen filter response. This feature can be very helpful for quick diagnosis of features without going through the lengthy mathematical computations.

We learned that using the Ram-Lak filter will help us when we are looking for sharp changes in the object image since it removes any smoothening within the image. This can be very useful for a surgeon in pinpointing the location of a tumor before surgery. Such a filter improves contrast, resolution and SNR of the reconstructed image as compared to a low pass filter which enhances the background haze and noise.