MSc Biomedical Engineering

University of Patras

Professor Lena Kostaridou, PhD S. Skiadopoulos

Student: Elissaios Petrai

petrai AT ceid.upatra.gr

Medical Imaging Project

Fan-beam tomographic image reconstruction algorithm

June 2018

Table of Contents

Read_me_first

1. Make sure that all files, as well as the initial image are in the same folder, then run main.
2. After running main, the code starts running in sections. This means that as soon as a section is executed, it displays a message on the command line and waits for the user to press the enter button in order to proceed to the next section. The user is required to pay attention to the messages that appear on the command line.

   This way was purely selected for better computer memory management and for practical reasons.

THEORY

With this project we aim to examine the effects of various parameters of tomographic image reconstruction from projections, which stimulate clinical conditions in modern Computed Tomography (CT) systems.

Projections are received using fan-beam geometry as depicted in the following image under specified rotarion angles. For this reason, fanbeam[1] and ifanbeam[2] MATLAB's functions are used.

However, it is worth making a brief reference to the theory presented in the slides that accompany the project, but also to the description [3,4] given on MATLAB's page.

More specifically, as shown in the figure below, fanbeam function computes projections of an image matrix along specified derections. The x-rays are emitted from a single source (vertex) and ended up with different degree of attenuation as they pass through the physical sample, to the sensor, forming a fan shape.

Therefore, as a projection we define a set of line (ie x rays) along the x,y axis, which are represented from a function f(x,y).

By taking many projections from different angles by rotating the source around the center of the image, we could reconstruct and represent the desired image.

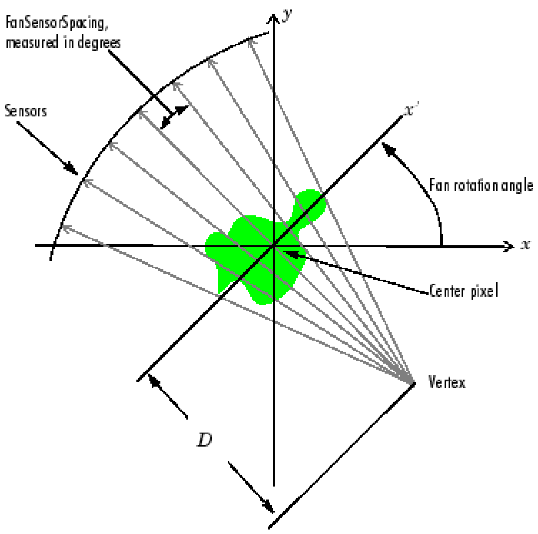


Figure 1: Fan-beam Projection at a Fan Rotation Angle theta

Concluding our brief theory, it is necessary to mention the SNR, CNR, ESF and LSF indices used to describe quantitatievely or qualitatively an image. We relied on the theory through the context of the course, but also from the course book, as well as the paper [5]

To be precise, in our case we define them as follows:

$SNR = \frac{\mu}{\sigma}$, where "μ" is the mean value of the signal intensity and "σ" is the standart deviation of the intensity in the background.

The CNR is therefore derived from the following formula

$$CNR = \frac{mean2(grey\,level_{bone}) - mean2(grey\,level_{background})}{stds(grey\,level_{backgounrd})} = |SNR_A - SNR_B|$$

For the computation of blurring we use ESF (Edge Spread Function) and LSF which is the derivative of the former. It essentially measures the change in brightness between two areas of interest through a line that we define.

Initial Sample Code

```
close all;
clear all

%% phantom image
P = phantom(512);
figure (1),imshow(P,[])

%% Compute fan-beam projection data of the phantom image
R = 380; %% R > = image size / 2
%% parameters
FanSensorGeometry1='arc'; % 'arc' and 'line'
FanSensorSpacing1=0.25; %, the arc of sensors is fixed = 87.7 degrees, Spacing (Δr):
0.850, 0.550, 0.350, 0.250, 0.150
FanRotationIncrement1=1; % Increment (Δs): 10
method_inter='linear'; % Interpolation: 'nearest', 'linear', 'cubic', 'spline'
filter='Hann'; % Filter: 'Ram-Lak','Shepp-Logan','Cosine','Hamming','Hann‘
filter_cutoff=0.85; %% cut-off frequency: 95%, 85%, 65%, 45%, 25% of window

[F3, sensor_pos, fan_rot_angle] =
fanbeam(P,R,'FanRotationIncrement',FanRotationIncrement1,'FanSensorGeometry',FanSensorG
eometry1,'FanSensorSpacing',FanSensorSpacing1);

%% Plot the projection data
figure (2), imagesc(fan_rot_angle, sensor_pos, F3)
colormap(hot); colorbar
xlabel('Fan Rotation Angle (degrees)')
ylabel('Fan Sensor Position (degrees)')

% % Reconstruct the image from the fan-beam projection data using ifanbeam. In each
reconstruction
output_size = max(size(P));
FanCoverage1='cycle'; %'minimal', 'cycle', no difference exists

Ifan3=ifanbeam(F3,R,'FanRotationIncrement',FanRotationIncrement1,
'FanSensorGeometry',FanSensorGeometry1,
'FanSensorSpacing',FanSensorSpacing1,'Filter',filter,'FrequencyScaling',filter_cutoff,'
Interpolation',method_inter,'OutputSize',output_size);
figure (3), imshow(Ifan3)
yi=[84,309];
xi=[262,262];
c = improfile(Ifan3,xi,yi)
figure(4),plot(c)

%%original
c1 = improfile(P,xi,yi)
figure(5),plot(c1)
```

TASKS

Task 1º: Initial image

Figure 2 shows the original image selected (randomly) for the project.

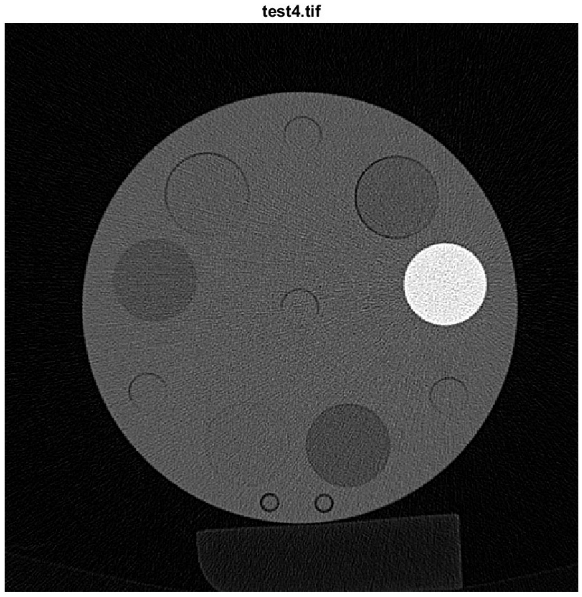


Figure 2: Selected Image test4.tif

Task 2º: ROIs

In *Figure 2* are depicted Regions of Interest (ROIs) and profiles selected from the original image. Specifically, bone, plastic water and a point in the background were selected for our project.

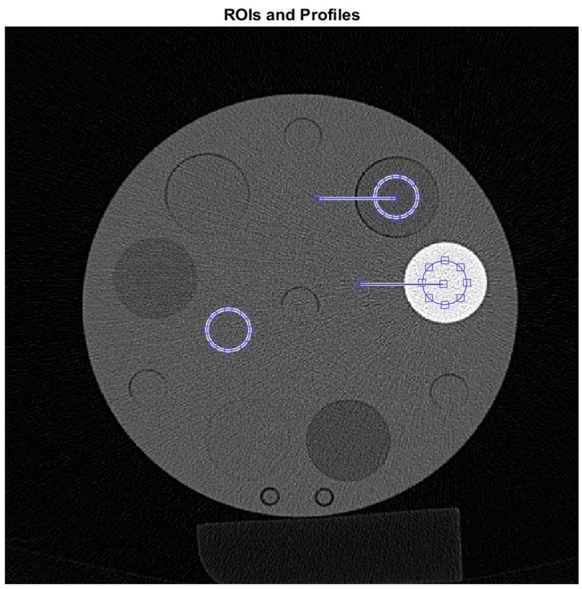
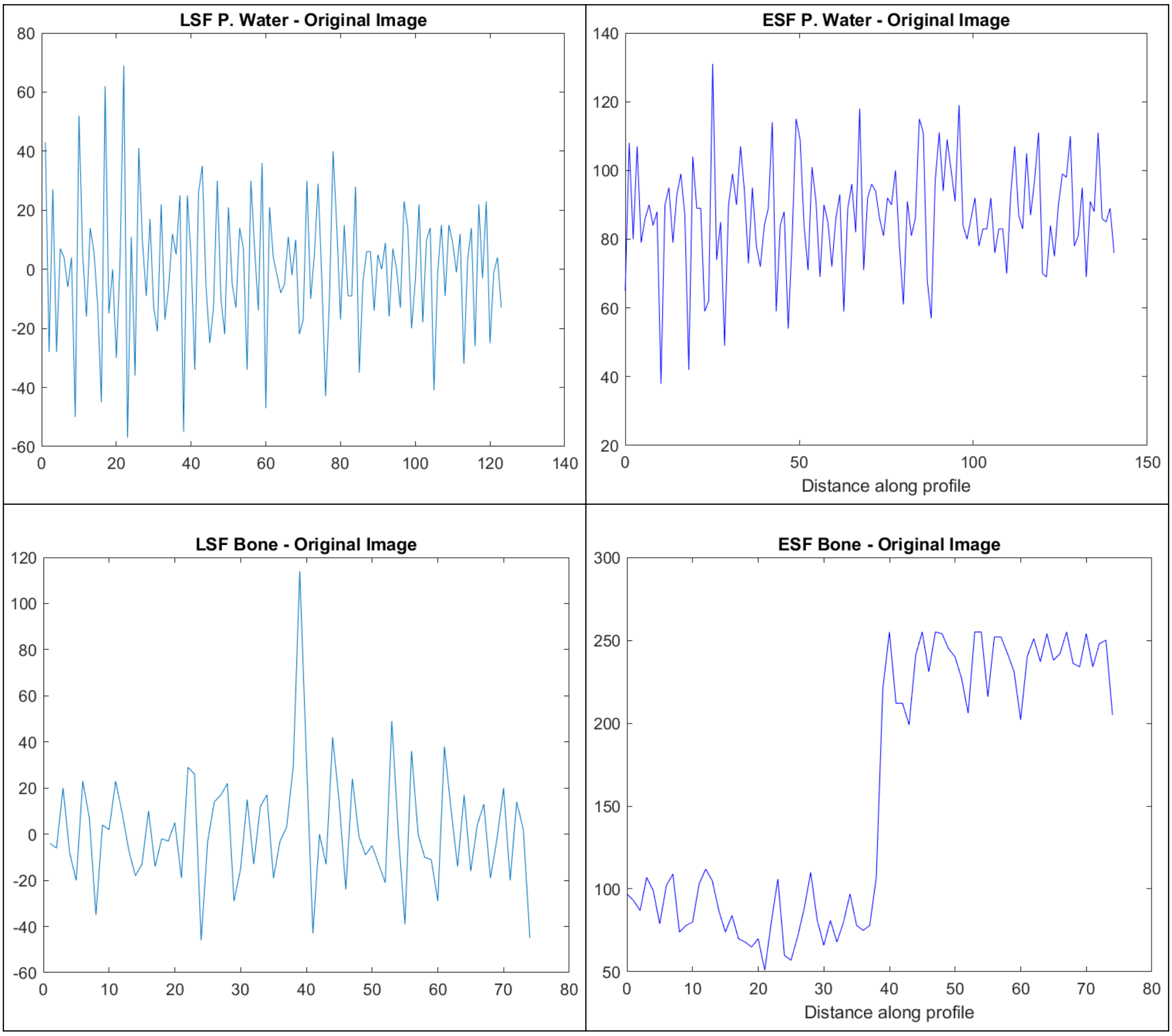


Figure 3: ROIs and Profiles

Next we quote the LSF and ESF of the original image

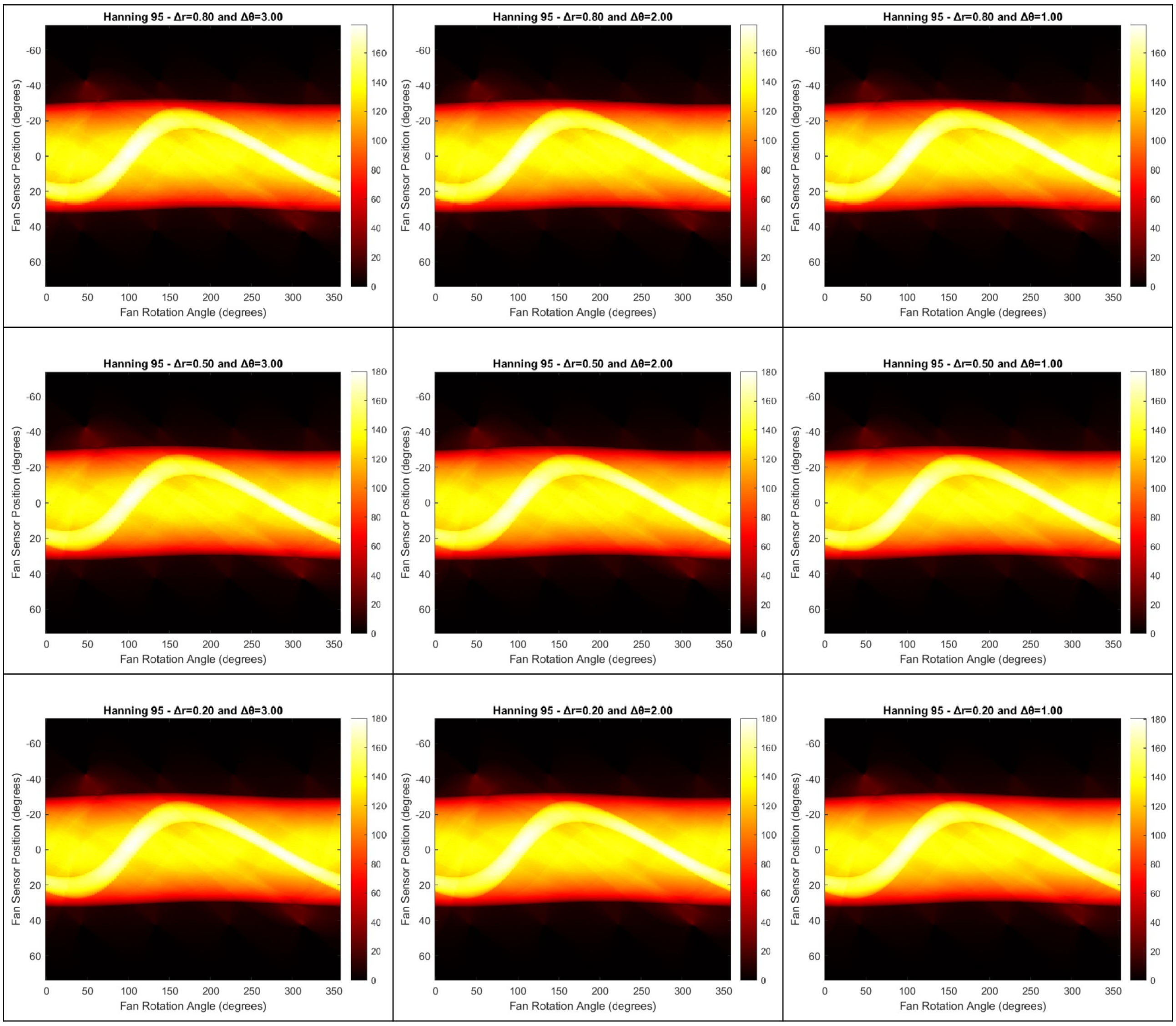
Table 1: LSF and ESF of the original image



Task 3º: Sinograms

For a *hanning filter* with a cut-off frequency 95%, we display sinograms for combination of *angular sampling (ie Sensor Spacing Δr)* and *projection samplaning (ie Rotation Increment Δθ)*. Specifically for the *Sensor Spacing* values Δr = 0.8, 0.5, 0.2 and *Rotation Increment* values Δθ = 3, 2, 1. The details of each sinogram are indicated in the title.

Table 2: Sinograms

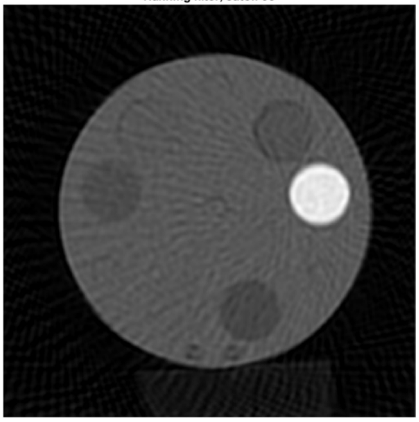
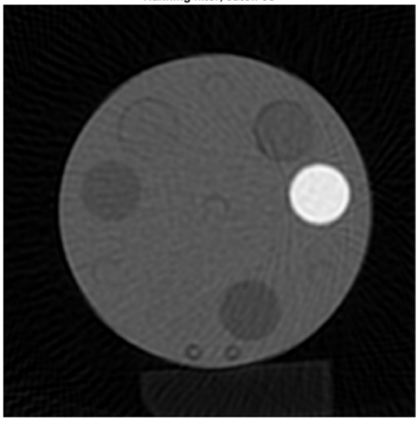
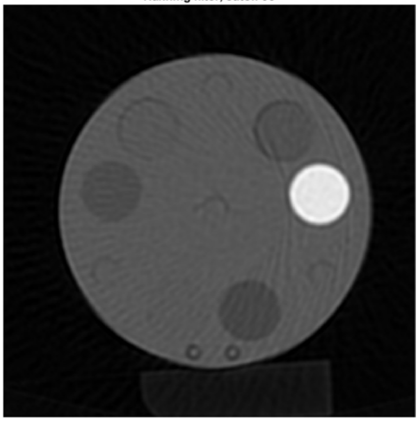
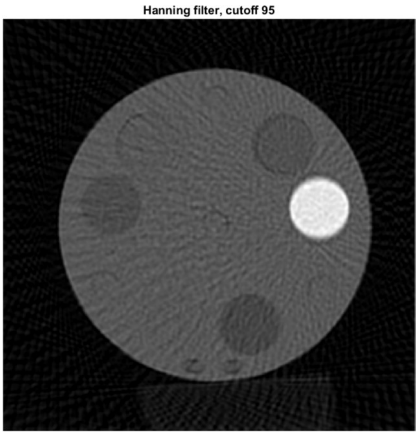
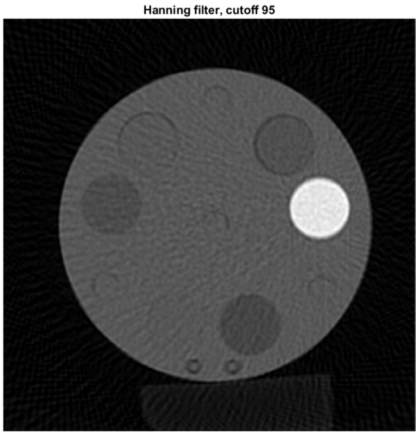
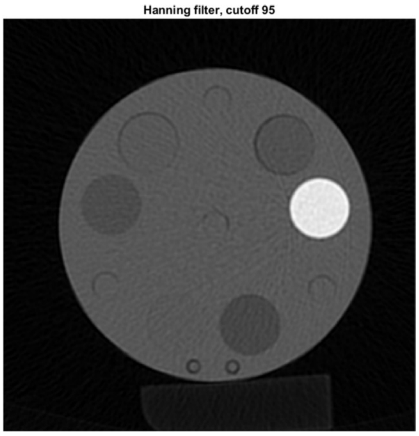
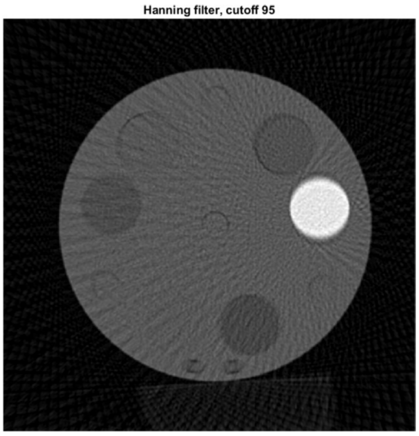
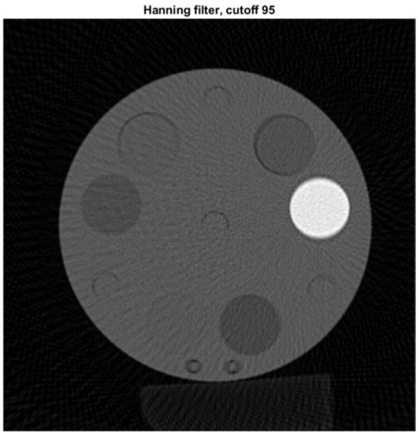
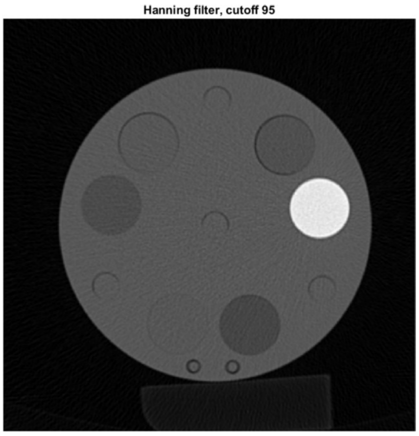


Next we move on to the *Task 4* where the reconstruction of these projections takes place.

Task 4°: Reconstructed Images (Hanning filter – Frequency Scaling 95%)

From the following reconstructed images, it is obvious that the optimal combination of angular sampling and projection sampling, for this particular filter, results for the values *Sensor Spacing* ($\Delta r=0.2$) and *Rotation Increment* ($\Delta\theta=1^0$ *– that corresponds to 360 projections)*.

Table 3: Reconstructed images

$\Delta r=0.8 - \Delta\theta=3$ Hanning filter, cutoff 95 	$\Delta r=0.8 - \Delta\theta=2$ Hanning filter, cutoff 95 	$\Delta r=0.8 - \Delta\theta=1$ Hanning filter, cutoff 95 
$\Delta r=0.5 - \Delta\theta=3$ Hanning filter, cutoff 95 	$\Delta r=0.5 - \Delta\theta=2$ Hanning filter, cutoff 95 	$\Delta r=0.5 - \Delta\theta=1$ Hanning filter, cutoff 95 
$\Delta r=0.2 - \Delta\theta=3$ Hanning filter, cutoff 95 	$\Delta r=0.2 - \Delta\theta=2$ Hanning filter, cutoff 95 	$\Delta r=0.2 - \Delta\theta=1$ Hanning filter, cutoff 95 

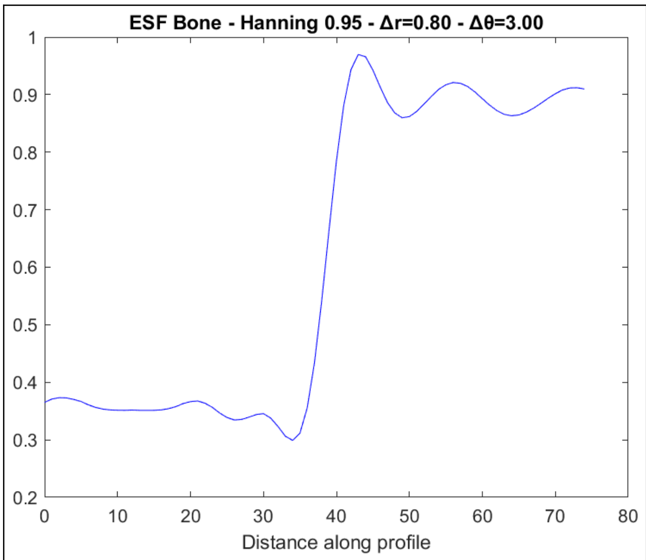
Next for the reconstructed images we present a table with the relative values of SNR and CNR in the ROIs we have selected.

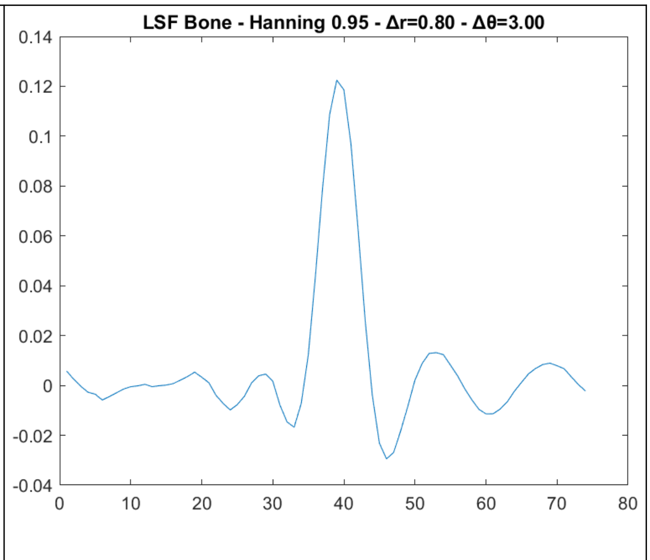
Table 4: SNR, CNR values for the given combinations Δr, Δθ

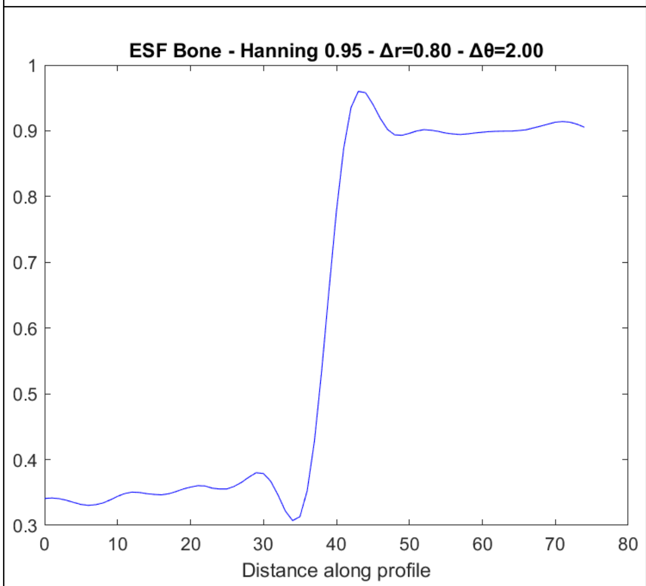
	SNR_Background	SNR_Bone	SNR_P.Water	CNR_Bone	CNR_P.Water
Δr=0.8 , Δs=3	20.214536	52.679355	18.378987	32.464819	1.835549
Δr=0.8 , Δs=2	30.509307	78.748784	27.476502	48.239477	3.032805
Δr=0.8 , Δs=1	49.017865	127.038148	44.323729	78.020283	4.694136
Δr=0.5 , Δs=3	14.137208	36.993759	12.911718	22.856551	1.225490
Δr=0.5 , Δs=2	16.357889	42.513979	14.777836	26.156089	1.580054
Δr=0.5 , Δs=1	32.168506	83.815083	29.123437	51.646577	3.045069
Δr=0.2 , Δs=3	14.212643	37.069528	12.899014	22.856885	1.313628
Δr=0.2 , Δs=2	17.346479	44.987698	15.646095	27.641219	1.700384
Δr=0.2 , Δs=1	26.826363	69.680422	24.246093	42.854059	2.580269

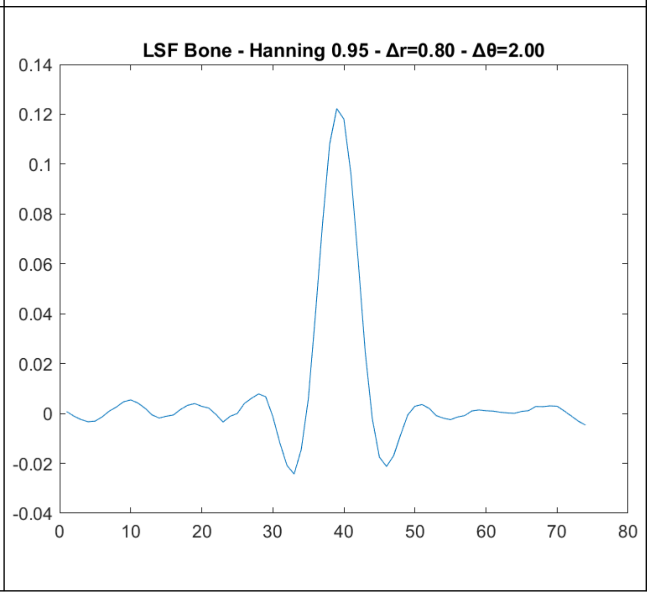
ESF and LSF for Bone

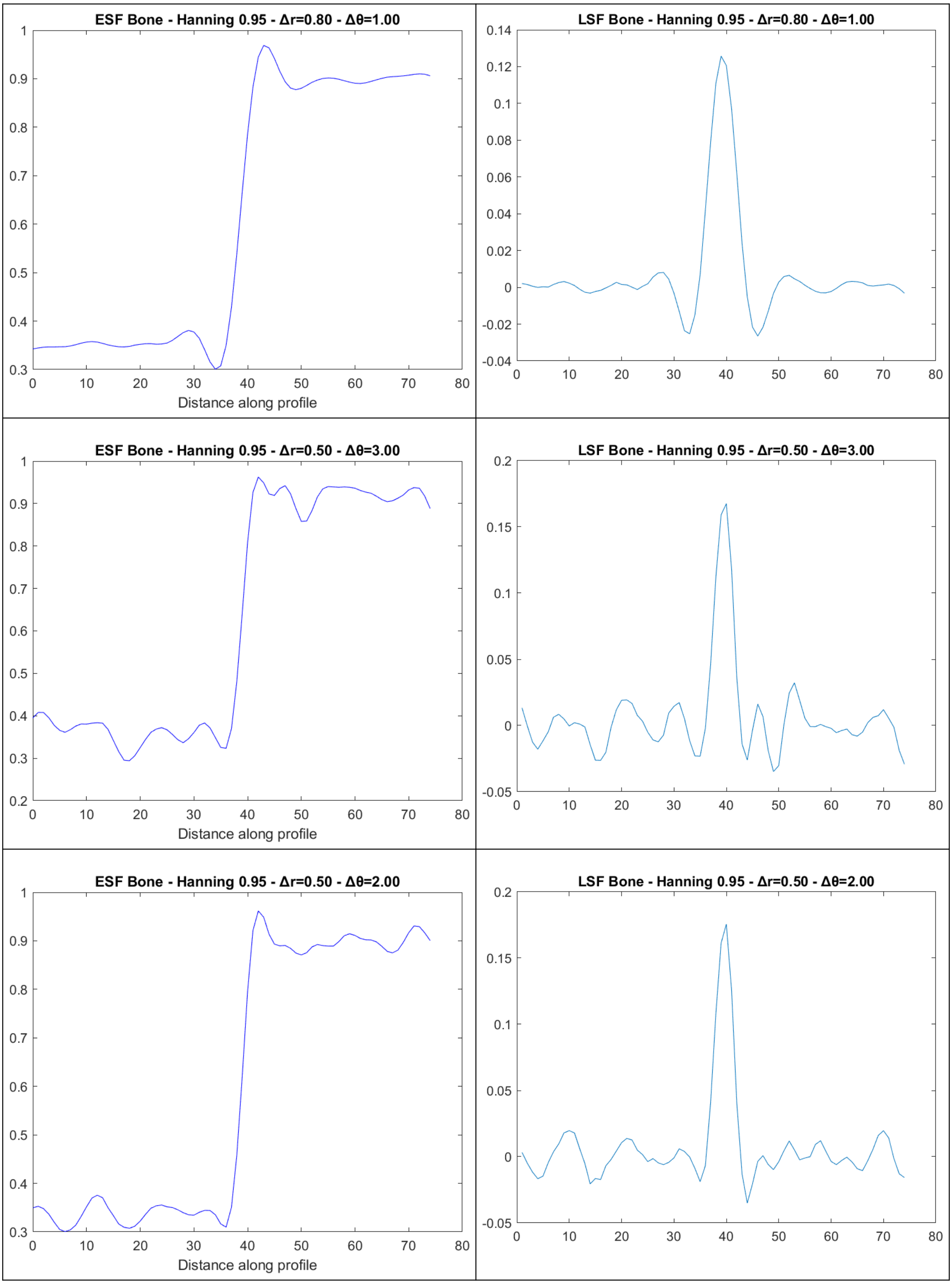
Table 5: ESF, LSF graphs for bone Profile

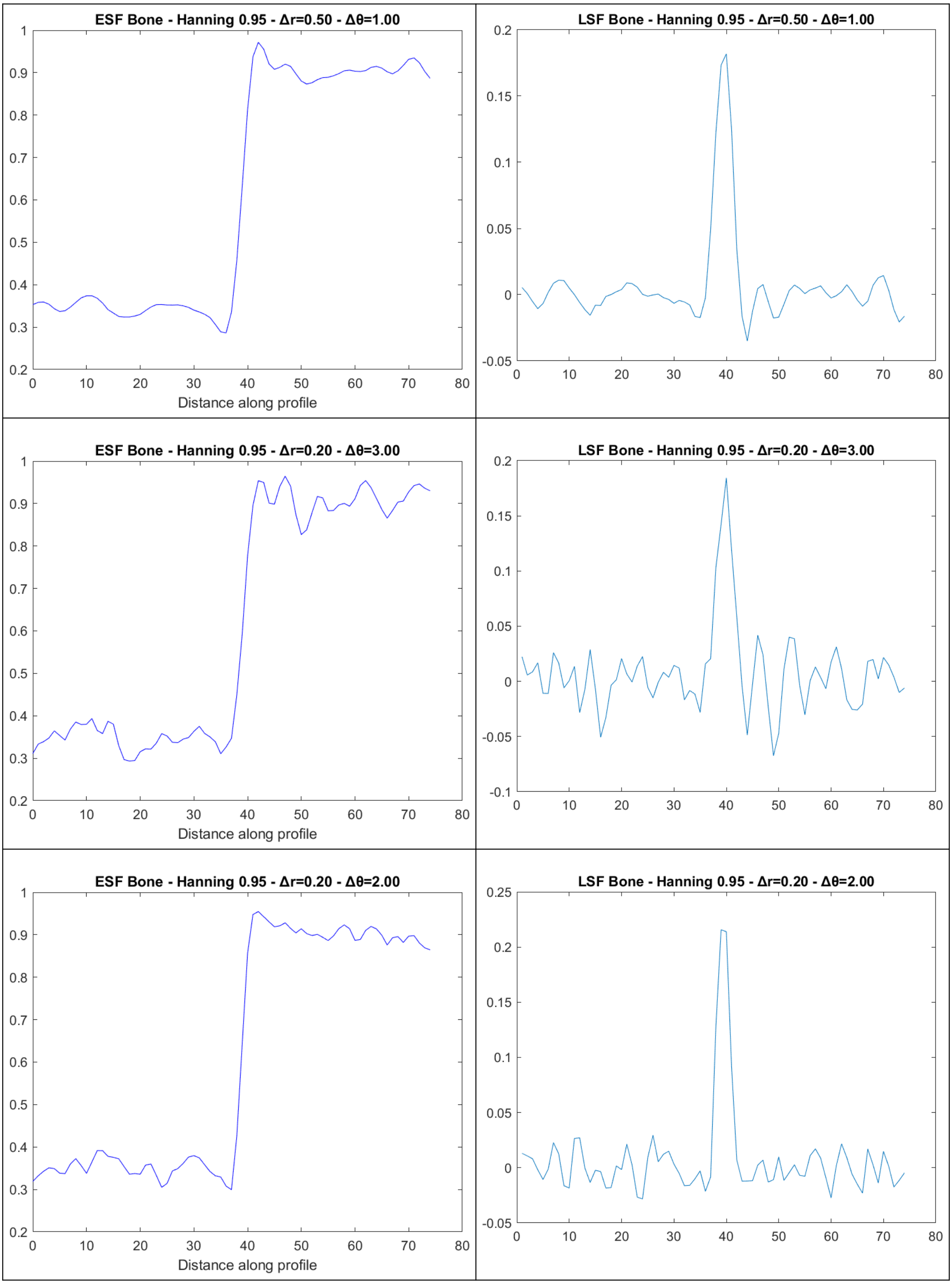


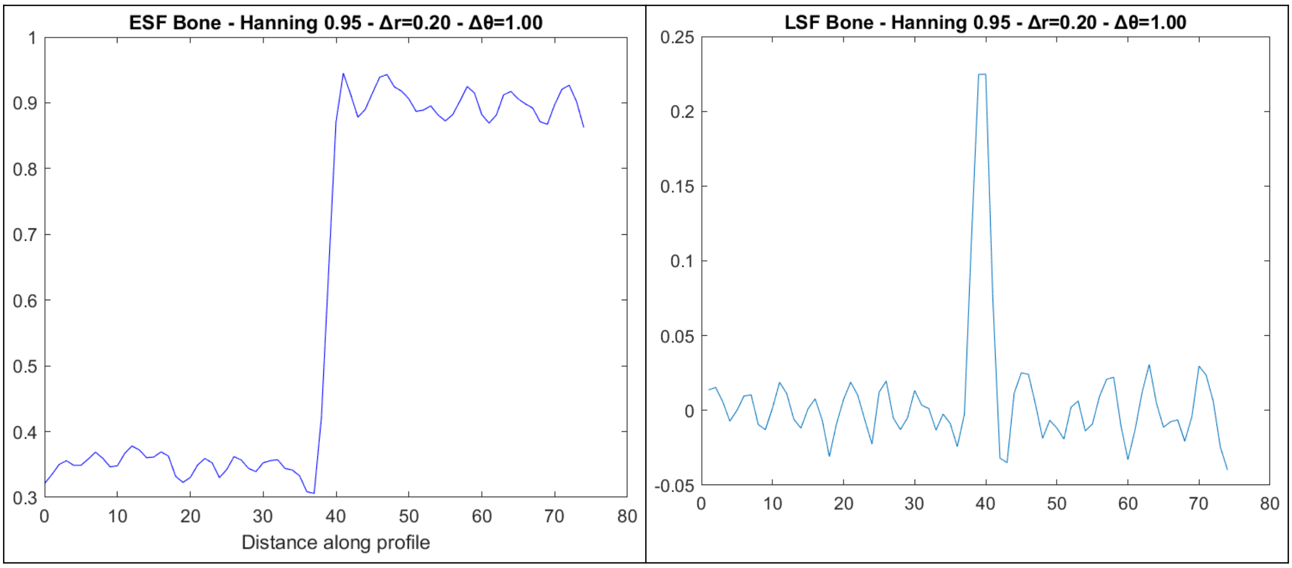






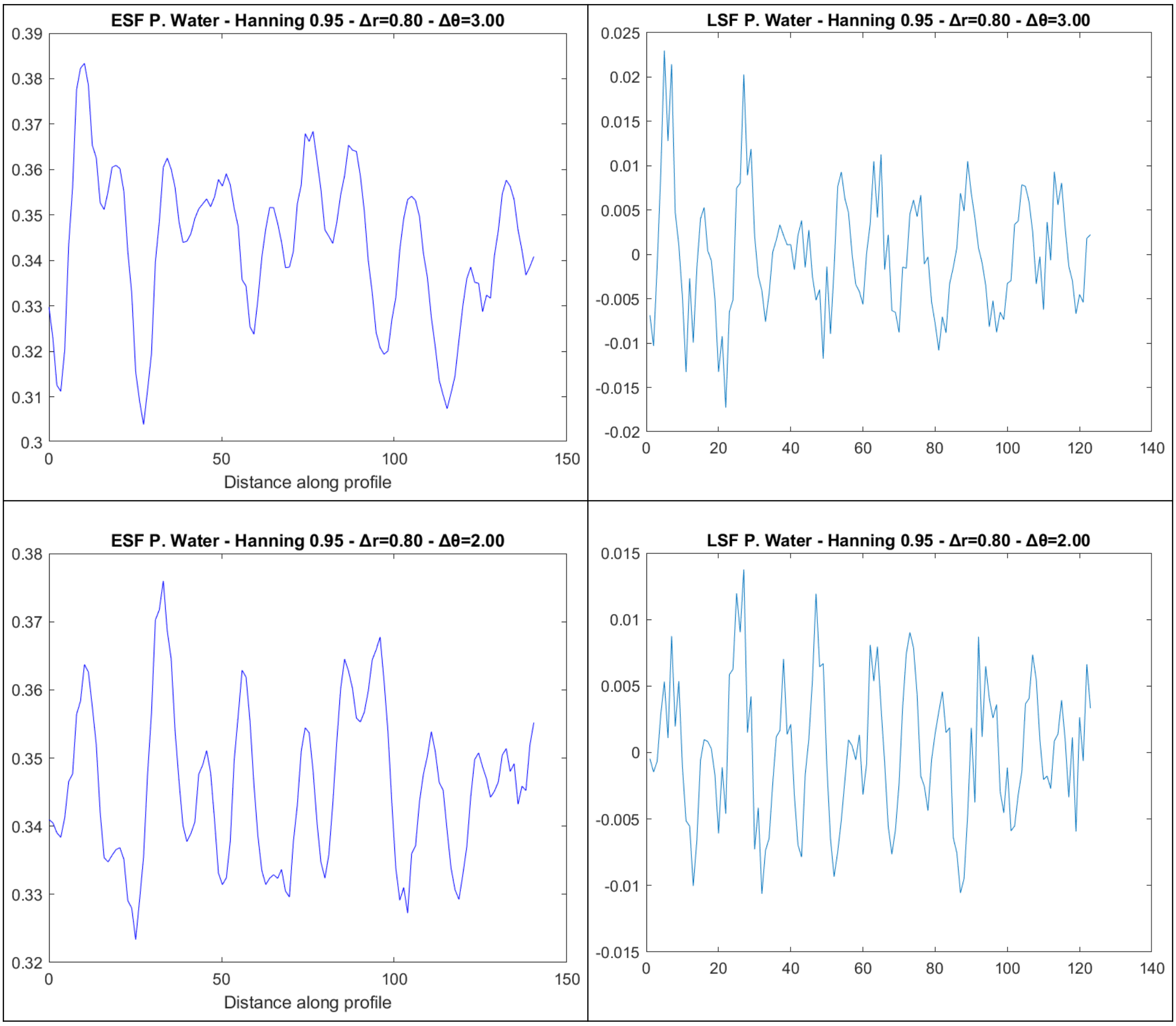

ESF Bone - Hanning 0.95 - Δr=0.80 - Δθ=1.00
Distance along profile
LSF Bone - Hanning 0.95 - Δr=0.80 - Δθ=1.00
ESF Bone - Hanning 0.95 - Δr=0.50 - Δθ=3.00
Distance along profile
LSF Bone - Hanning 0.95 - Δr=0.50 - Δθ=3.00
ESF Bone - Hanning 0.95 - Δr=0.50 - Δθ=2.00
Distance along profile
LSF Bone - Hanning 0.95 - Δr=0.50 - Δθ=2.00

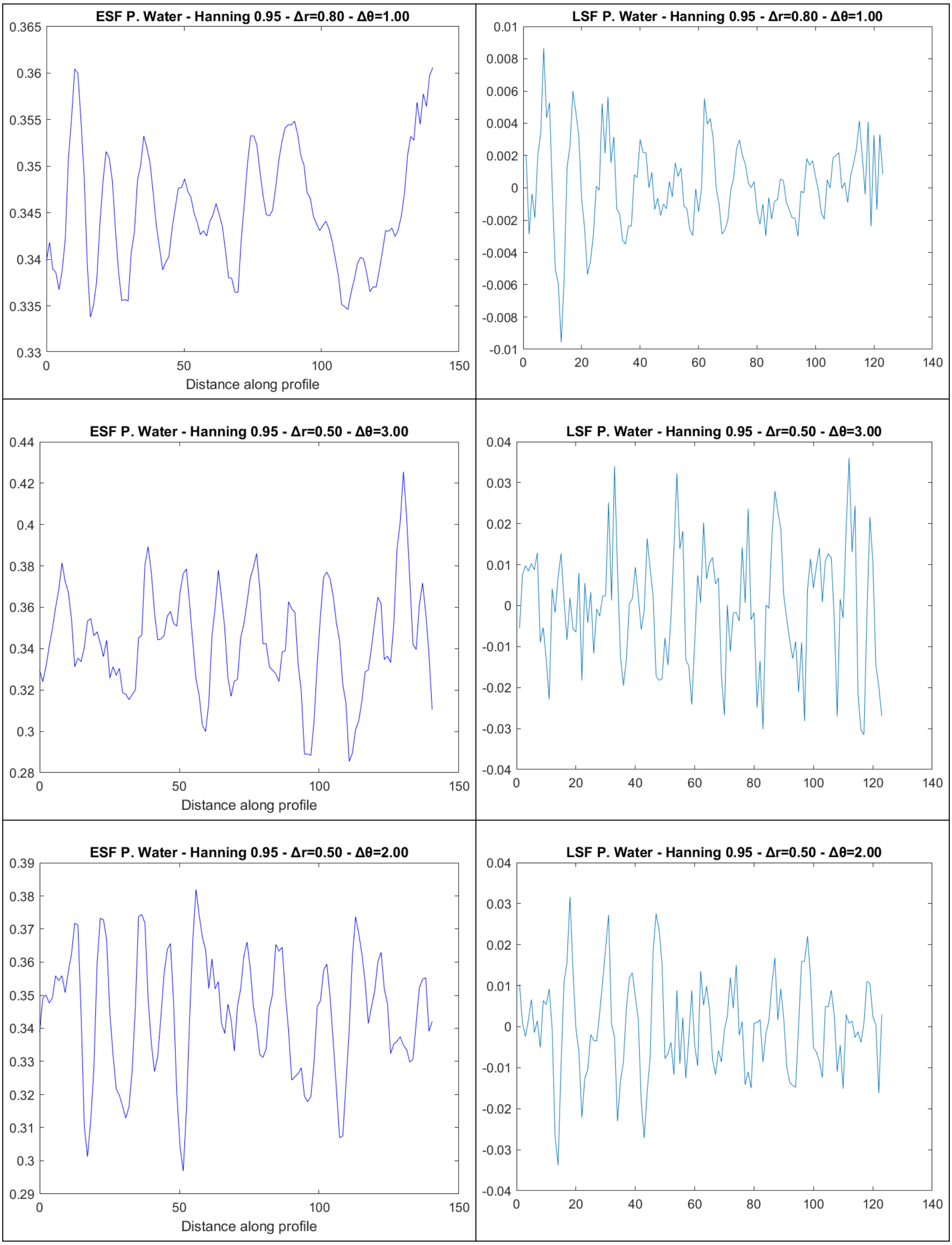

ESF Bone - Hanning 0.95 - Δr=0.50 - Δθ=1.00
Distance along profile
LSF Bone - Hanning 0.95 - Δr=0.50 - Δθ=1.00
ESF Bone - Hanning 0.95 - Δr=0.20 - Δθ=3.00
Distance along profile
LSF Bone - Hanning 0.95 - Δr=0.20 - Δθ=3.00
ESF Bone - Hanning 0.95 - Δr=0.20 - Δθ=2.00
Distance along profile
LSF Bone - Hanning 0.95 - Δr=0.20 - Δθ=2.00

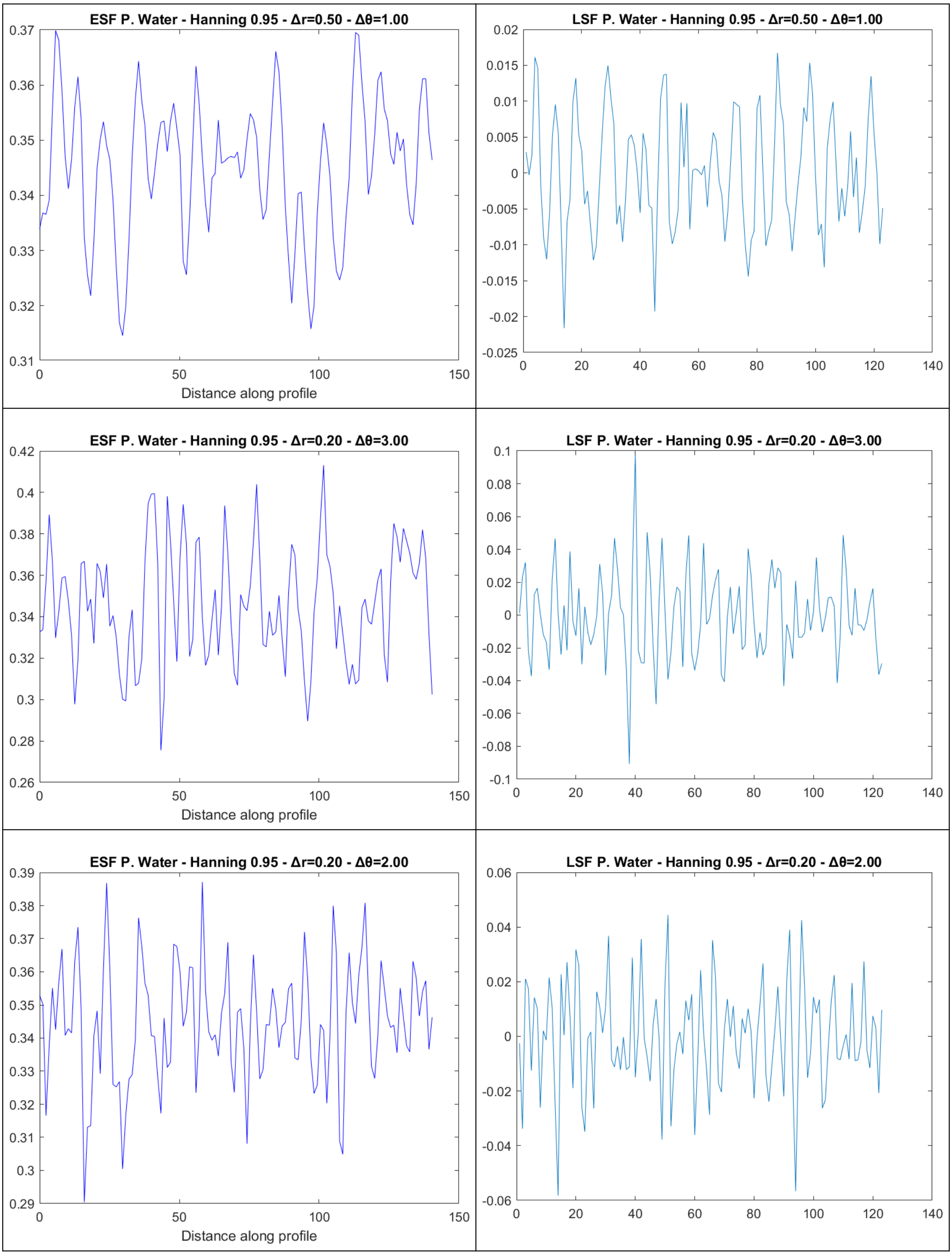


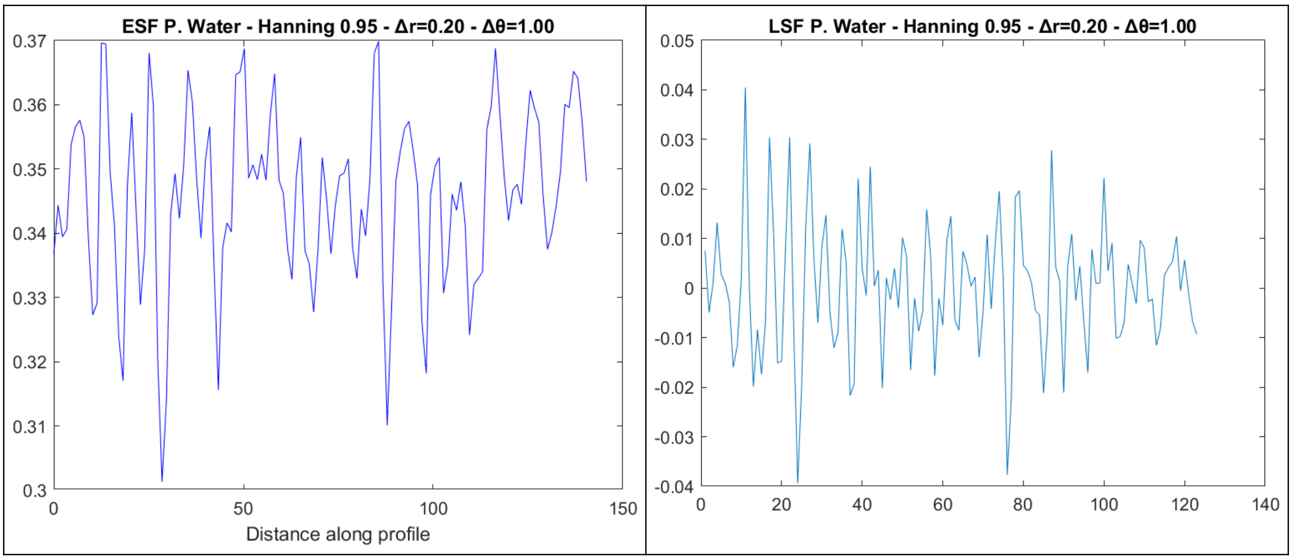
ESF and LSF for Plastic Water

Table 6: ESF, LSF graphs for Plastic Water Profile




ESF P. Water - Hanning 0.95 - Δr=0.80 - Δθ=1.00
Distance along profile
LSF P. Water - Hanning 0.95 - Δr=0.80 - Δθ=1.00
ESF P. Water - Hanning 0.95 - Δr=0.50 - Δθ=3.00
Distance along profile
LSF P. Water - Hanning 0.95 - Δr=0.50 - Δθ=3.00
ESF P. Water - Hanning 0.95 - Δr=0.50 - Δθ=2.00
Distance along profile
LSF P. Water - Hanning 0.95 - Δr=0.50 - Δθ=2.00


ESF P. Water - Hanning 0.95 - Δr=0.50 - Δθ=1.00
Distance along profile
LSF P. Water - Hanning 0.95 - Δr=0.50 - Δθ=1.00
ESF P. Water - Hanning 0.95 - Δr=0.20 - Δθ=3.00
Distance along profile
LSF P. Water - Hanning 0.95 - Δr=0.20 - Δθ=3.00
ESF P. Water - Hanning 0.95 - Δr=0.20 - Δθ=2.00
Distance along profile
LSF P. Water - Hanning 0.95 - Δr=0.20 - Δθ=2.00



In the above figures we see how the curves for these combinations are affected. Practically we observe that for the plastic water area, narrower LSF, implies a better and more clear/distinct image.

Task 5^o: Reconstructed Images by Applying Different Filters for Frequency Scaling 95%

From "Task 4" it emerged that the optimal combination was for the values Δr=0.2 and Δθ=1^0.

Therefore, for these values we will test and depict the reconstructed images for the filters "Ram-Lak", "Cosine" and "Hamming" with a frequency Scaling of 100%.

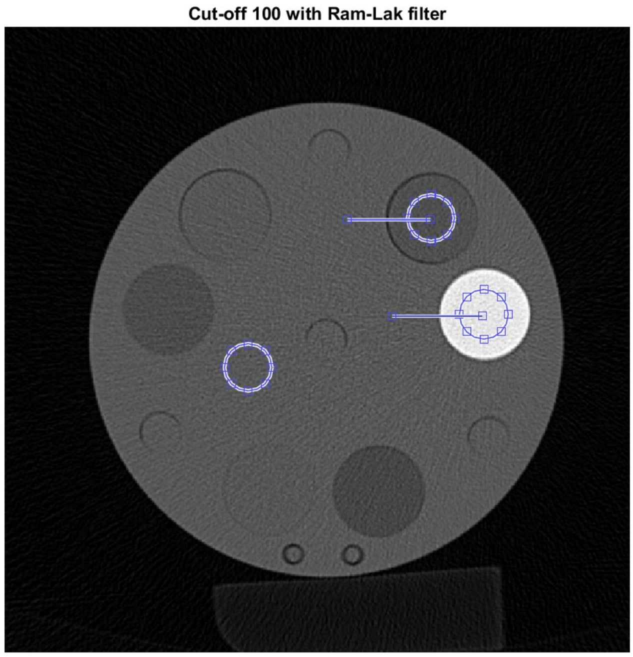
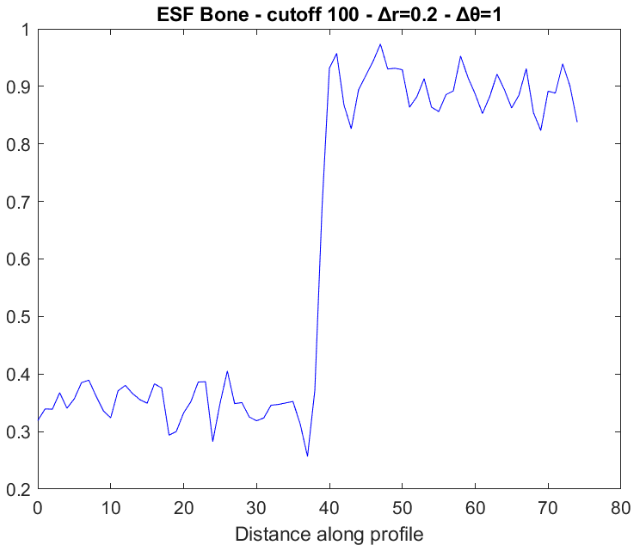
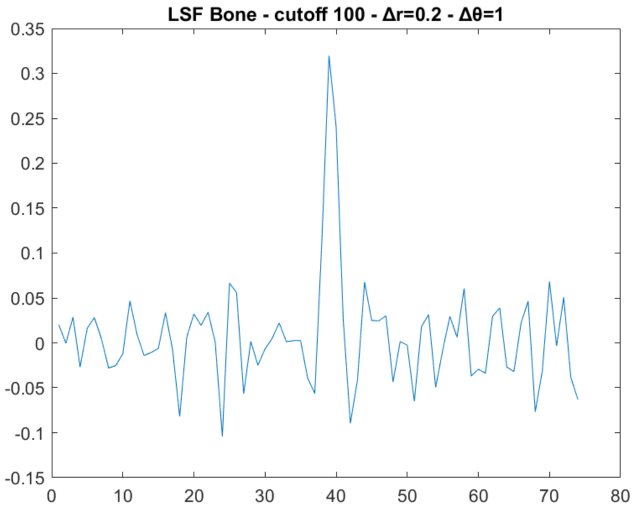
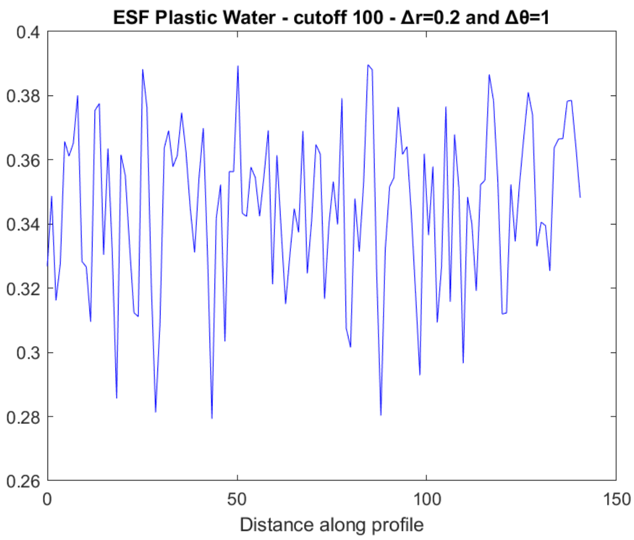
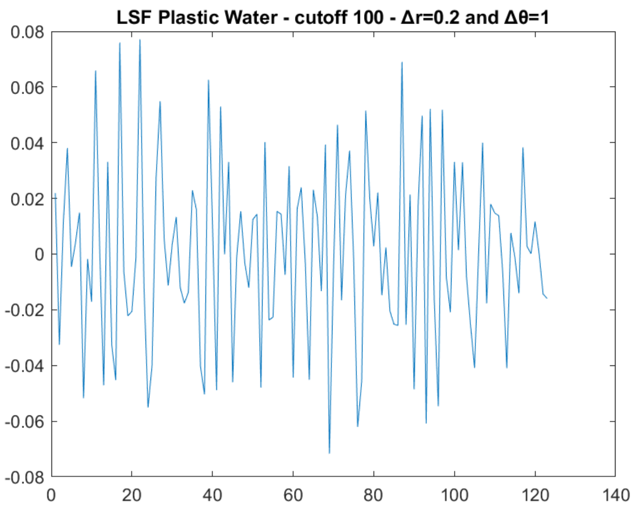
First we present (collectively) a table with the resulting SNR and CNR values:

Table 7: SNR, CNR values for Low-Rak, Cosine and Hamming filters

Δr=0.2 , Δs=1	**SNR_Backgound**	**SNR_Bone**	**SNR_P.Water**	**CNR_Bone**	**CNR_P.Water**
Low-Rak	13.075513	33.957889	11.811651	20.882375	1.263862
Cosine	19.114943	49.645513	17.272938	30.530570	1.842005
Hamming	23.790153	61.791897	21.500247	38.001744	2.289906

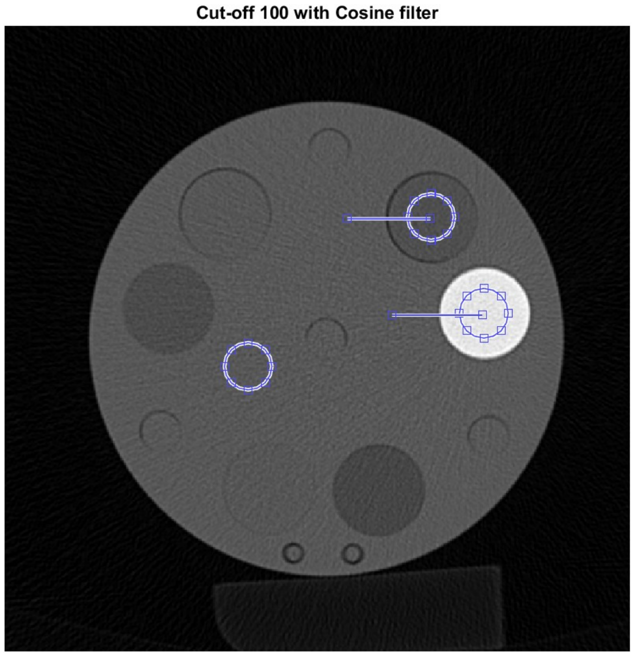
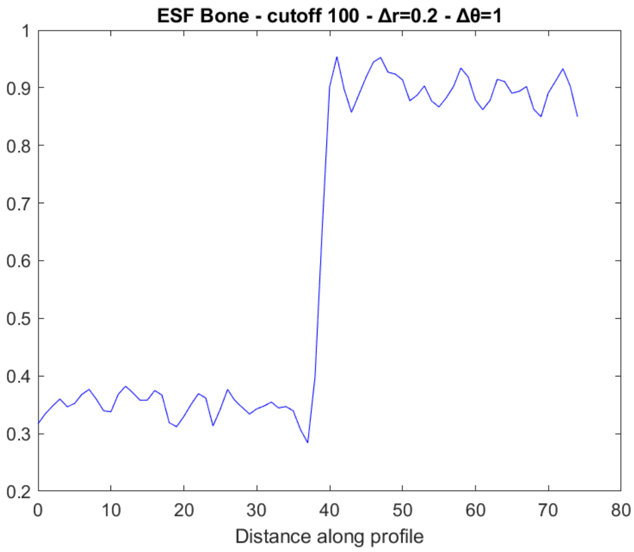
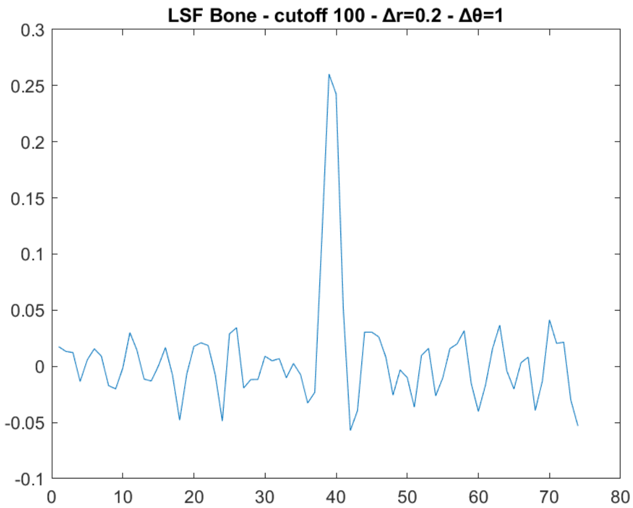
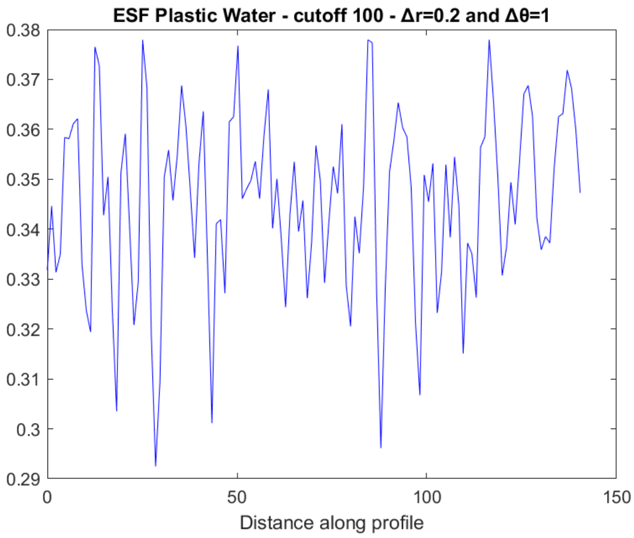
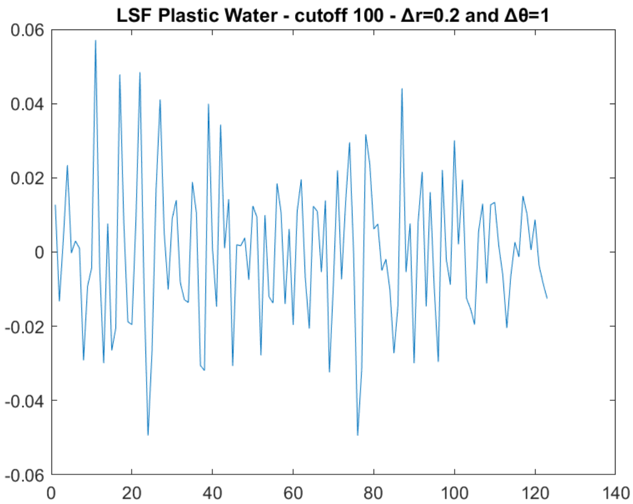
Filter: Ram-Lak

Table 8: Reconstructed image and ESF, LSF graphs for Ram-Lak filter

SNR_Background=13.075513 SNR_Bone=33.957889 SNR_P.Water=11.811651 CNR_Bone=20.882375 CNR_P.Water=1.263862	
	
	

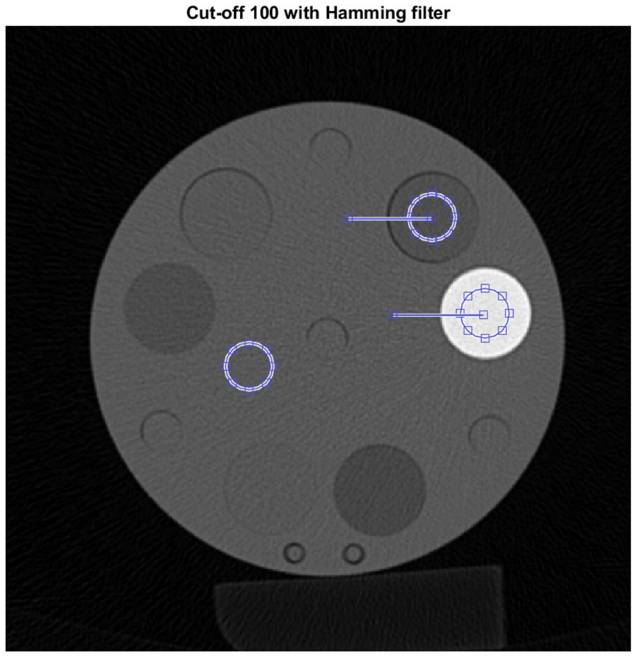
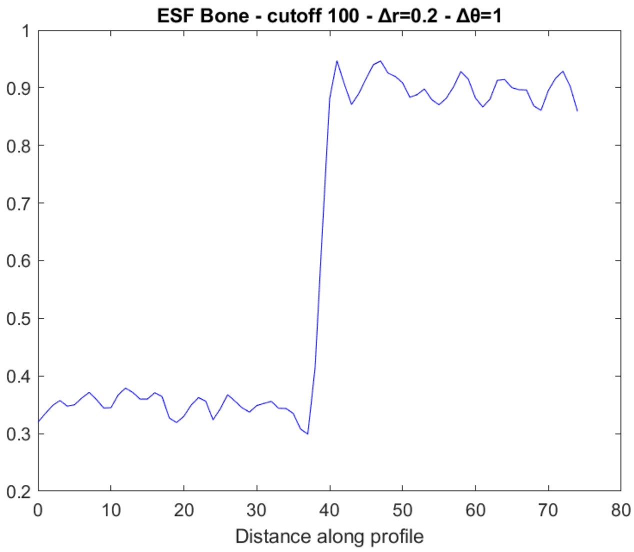
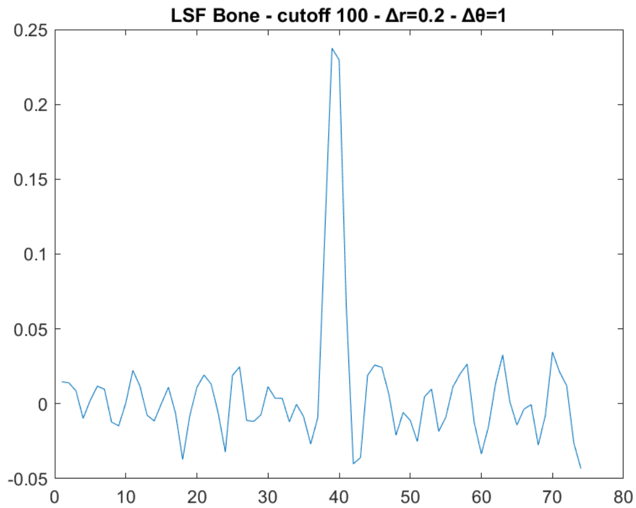
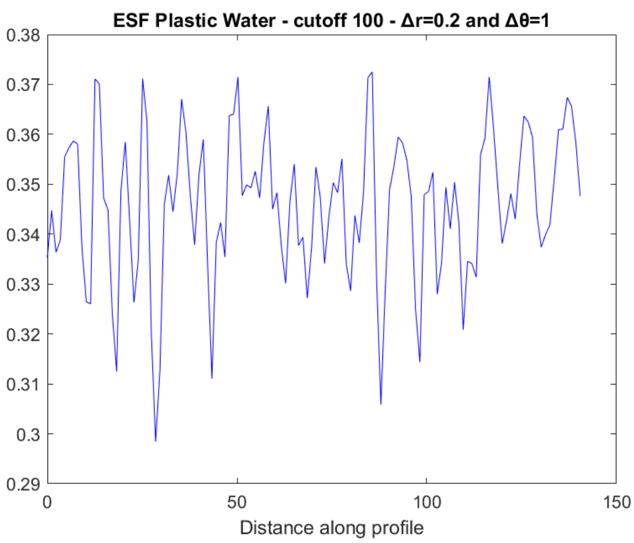
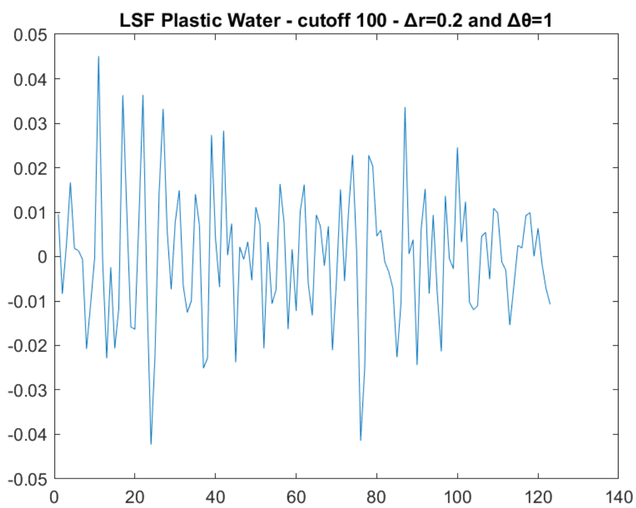
Filter: Cosine

Table 9: Reconstructed image and ESF, LSF graphs for Cosine filter

SNR_Background=19.114943 SNR_Bone=49.645513 SNR_P.Water=17.272938 CNR_Bone=30.530570 CNR_P.Water=1.842005	
	
	

Filter: Hamming

Table10: Reconstructed image and ESF, LSF graphs for Ram-Lak filter

SNR_Background=23.790153 SNR_Bone=61.791897 SNR_P.Water=21.500247 CNR_Bone=38.001744 CNR_P.Water=2.289906	
	
	

The differences that arise for the above images, through the human eye are difficult to recognize.

Although the Cosine filter seemed slightly more efficient based on the image quality to the human eye, however we relied on quantitative numbers and chose to continue with the hamming filter. This is because as we said, the SNR is defined as the signal strenght to the noise strenght. Therefore, a higher number of SNRs (theoretically) means a better picture. In practice, however, this is the case when this value is also accompanied by a high value of the CNR.

Task 6^o: Filter Application for 100%, 90% and 80% Frequency Scaling

In the previous question where we applied the Ram-Lak, Cosine and Hamming filters, although visually the differences were not as strong as we already mentioned, however due to SNR and CNR values the Hamming filter was considered optimal between them. Therefore, we will reapply this filter for frequencies scaling 100%, 90% and 80%.

For the Hamming filter and $\Delta r=0.2$, $\Delta\theta=1$ the following CNR and SNR values obtained.

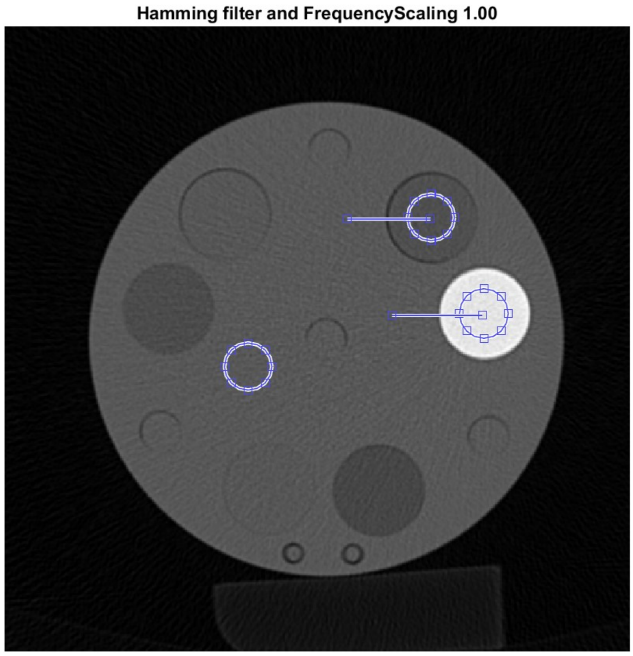
Table 11: SNR, CNR values for the corresponding Cutting Frequencies

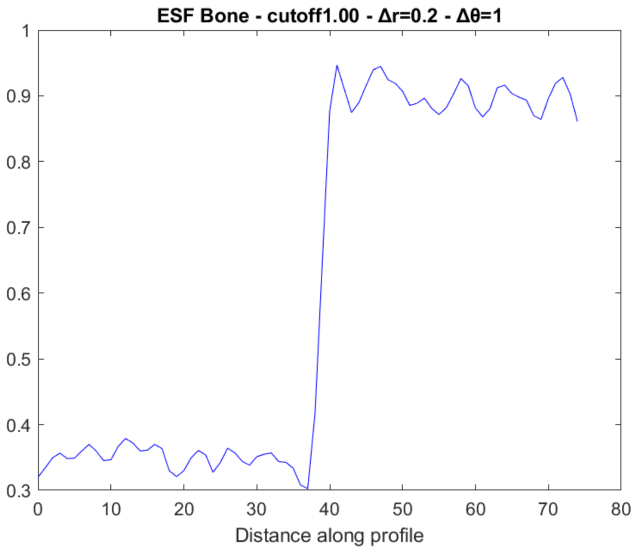
Δr=0.2 , Δs=1	**SNR_Background**	**SNR_Bone**	**SNR_P.Water**	**CNR_Bone**	**CNR_P.Water**
100%	25.267689	65.630817	22.836664	40.363128	2.431025
90%	28.684143	74.507365	25.925971	45.823222	2.758172
80%	33.592518	87.261263	30.363935	53.668745	3.228583

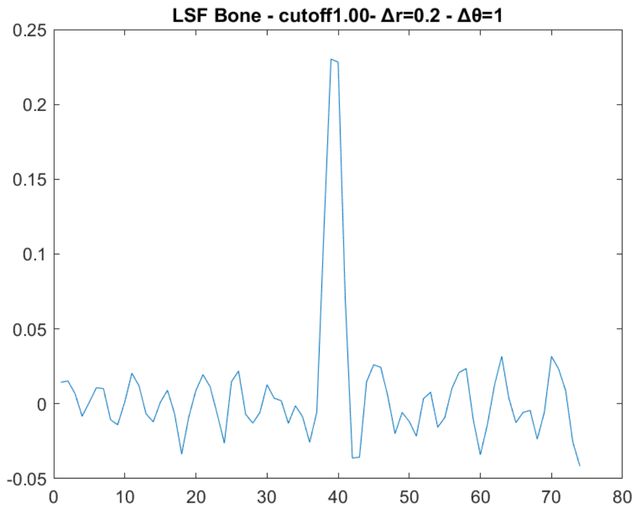
We also quote the reconstructed images along with the ESF and LSF.

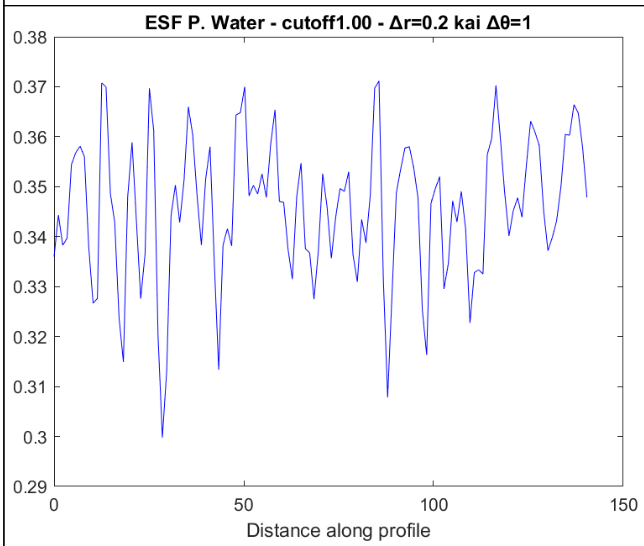
Cut-off Frequency 100%

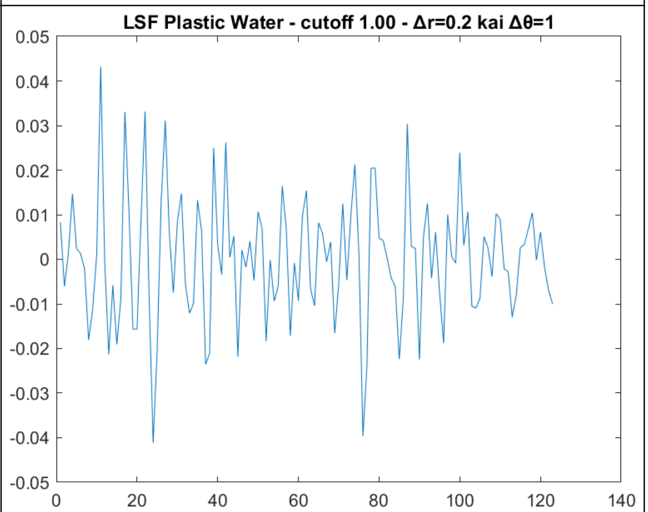
Table12:

<table>
<tr>
<td>Reconstructed Image and ESF, LSF graphs for Ram-Lak filter with cut-off frequency 100%.</td>
<td>


</td>
</tr>
<tr>
<td>


</td>
<td>


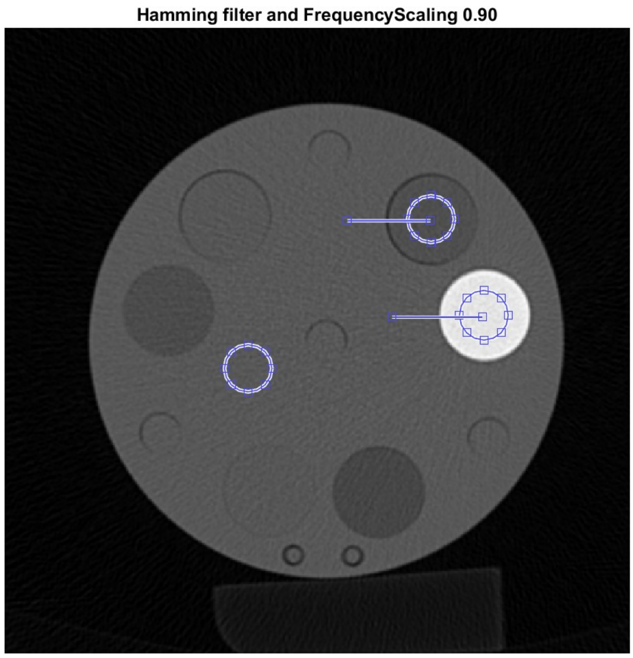
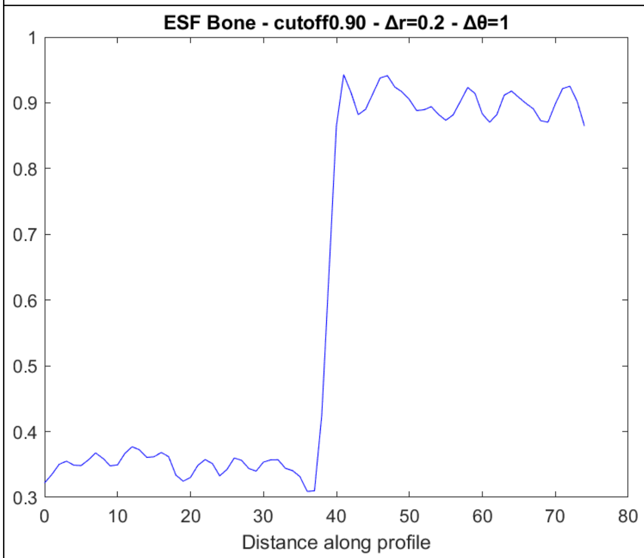
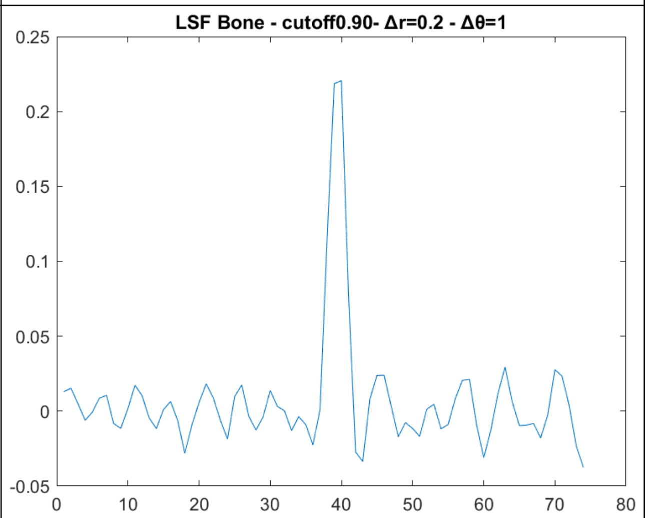
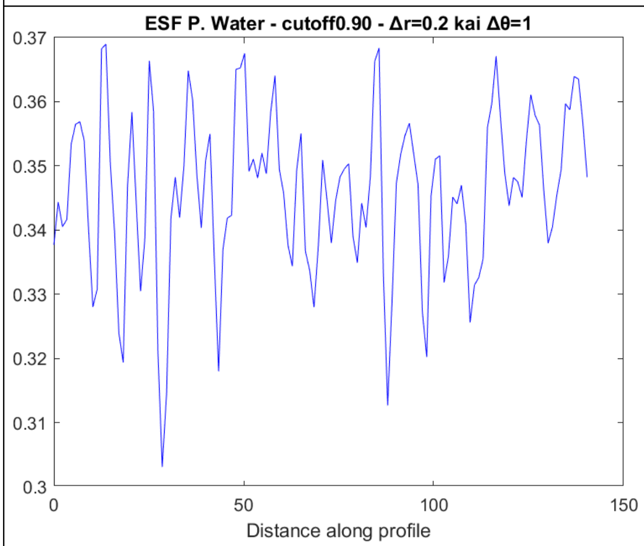
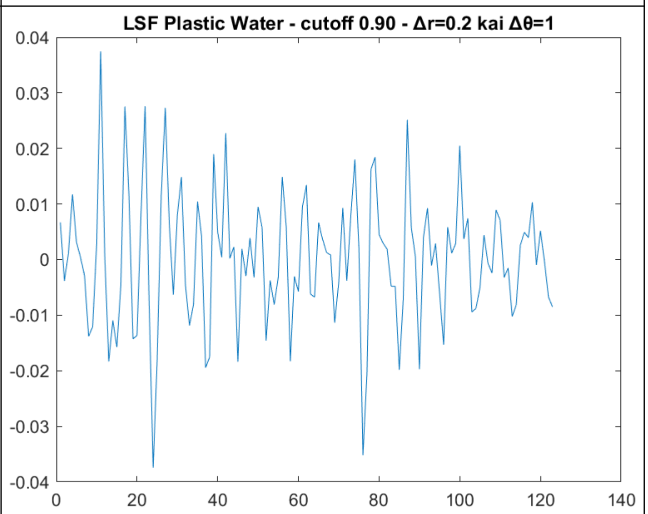
</td>
</tr>
<tr>
<td>


</td>
<td>


</td>
</tr>
</table>

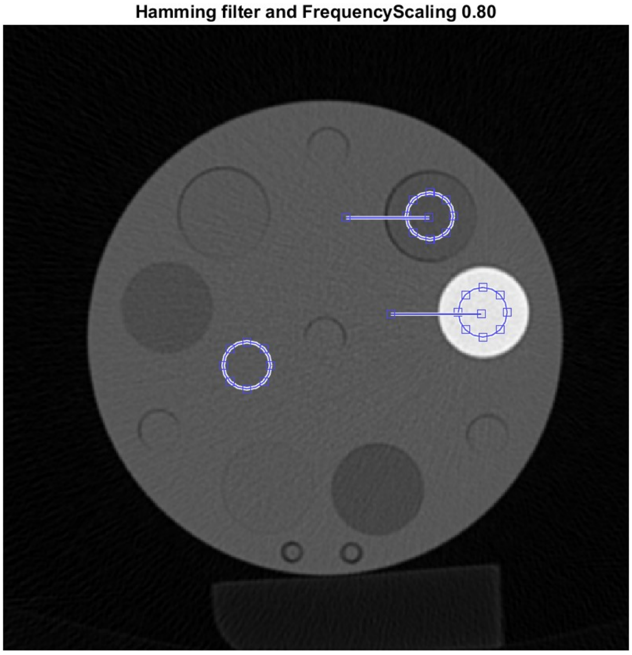
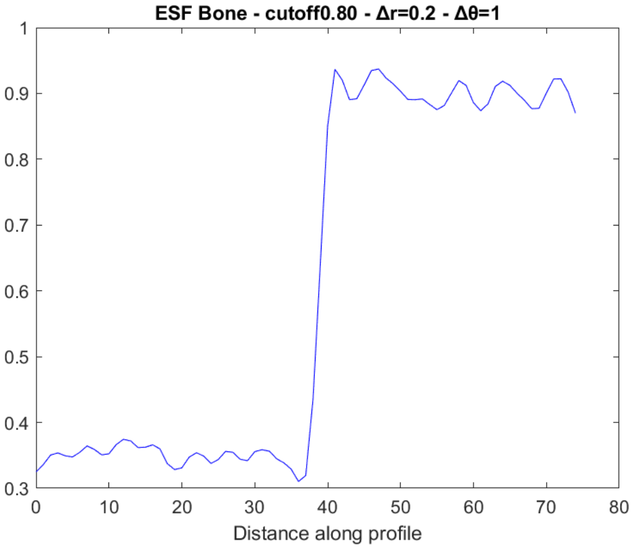
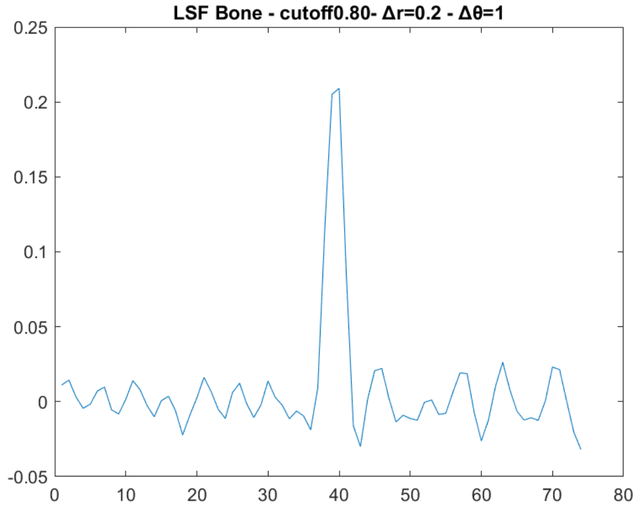
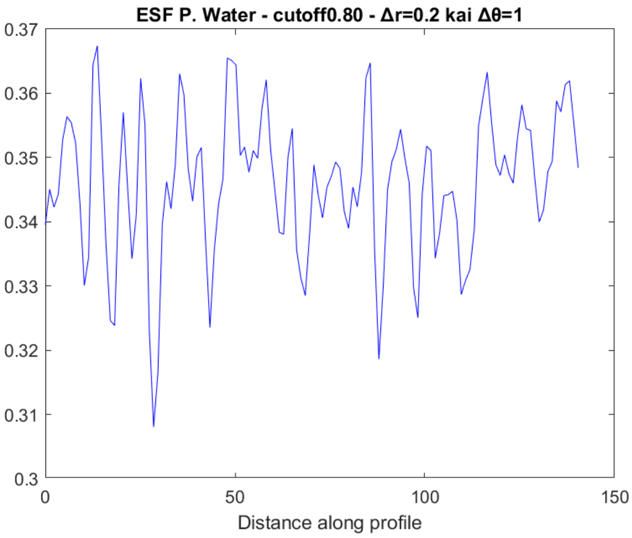
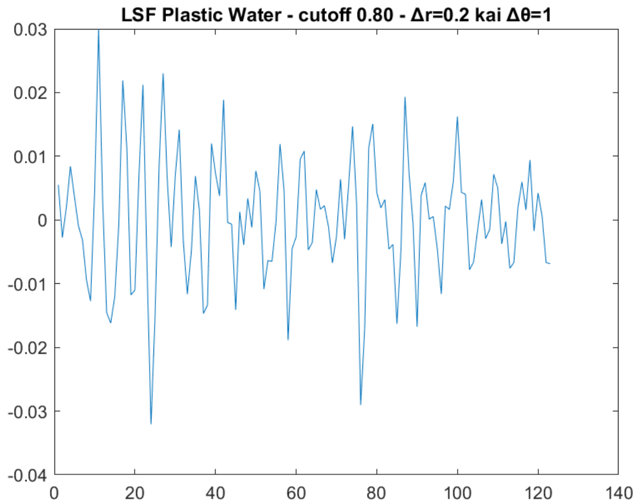
Cut-off Frequency 90%

Table 13:

Reconstructed Image and ESF, LSF graphs for Ram-Lak filter with cut-off frequency 90%.	
	
	

Cut-off Frequency 80%

Table14:

Reconstructed Image and ESF, LSF graphs for Ram-Lak filter with cut-off frequency 80%.	
	
	

QUESTIONS

1st Question

In this question is asked to present the corresponding graphs in order to answer the question how the image quality indicators differ (SNR, CNR, blurring of structures) as a function of angular sampling (ie Sensor Spacing Δr), projection sampling (ie Rotation Increment Δθ), projection modification filter, as well as the cut-off frequency of the filter.

Thus, in the same order, the corresponding graphs (based on the values from *Table 4)* are listed:

As a Function of Angular Sampling Δr

Three graphs are presented for each value of Δθ

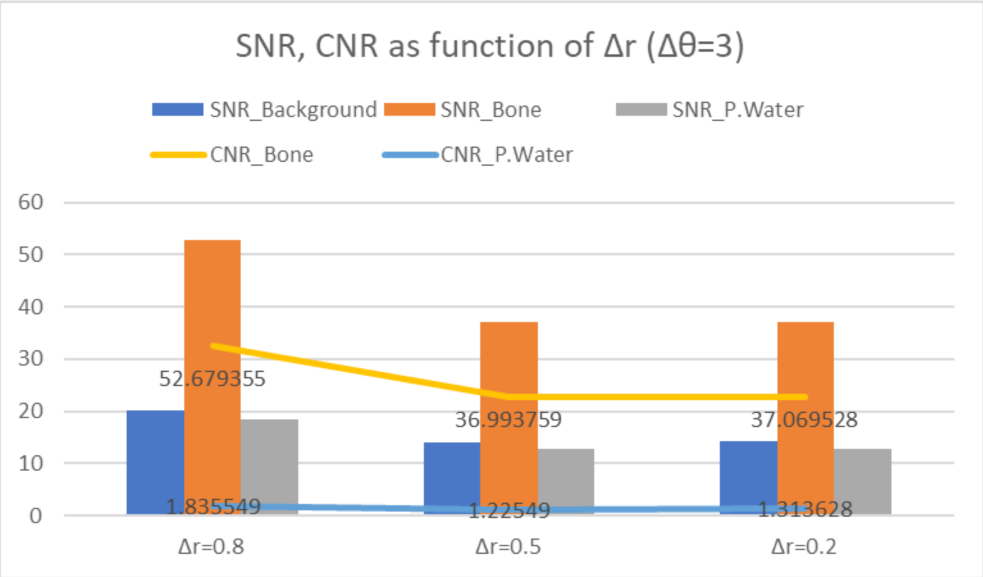


Figure 4: SNR, CNR as function of Δr, for Δθ=3

In this diagram, the result is controversial as for shorter distance between the sensors, we would expect less noise and therefore a higher SNR value.

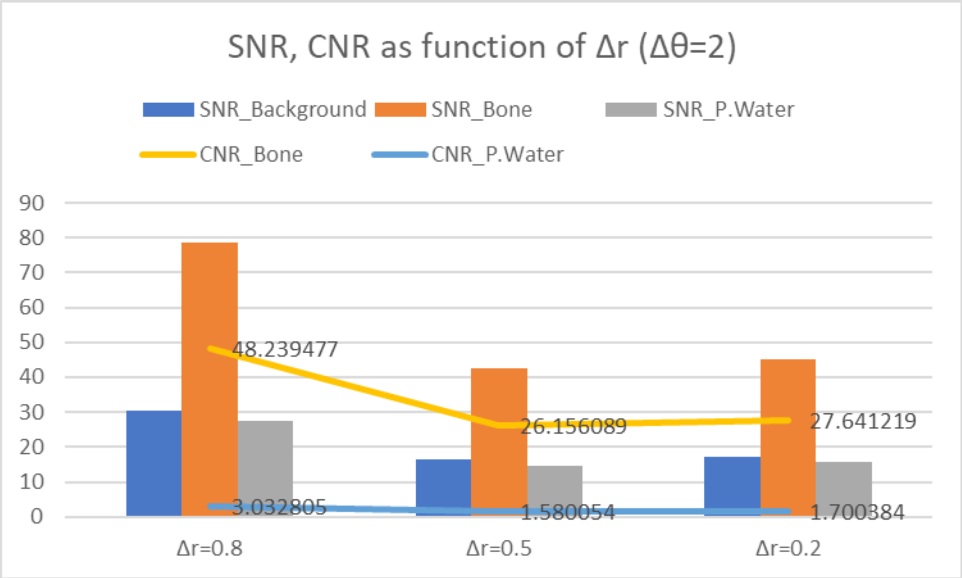


Figure 5: SNR, CNR as function of Δr, for Δθ=2

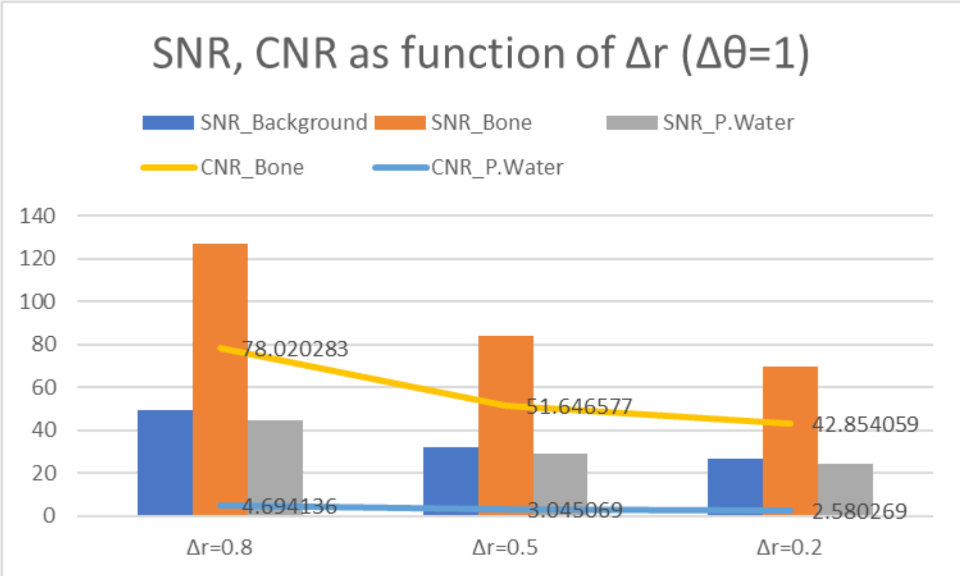


Figure 6: SNR, CNR as function of Δr, for Δθ=1

As a Function of Sampling Projections Δs

By the same token, we got at this point three different graphs, one for each value of Δr:

From the diagrams below it is clear that increasing the number of projections (from 120 to 360) significantly improves image quality.

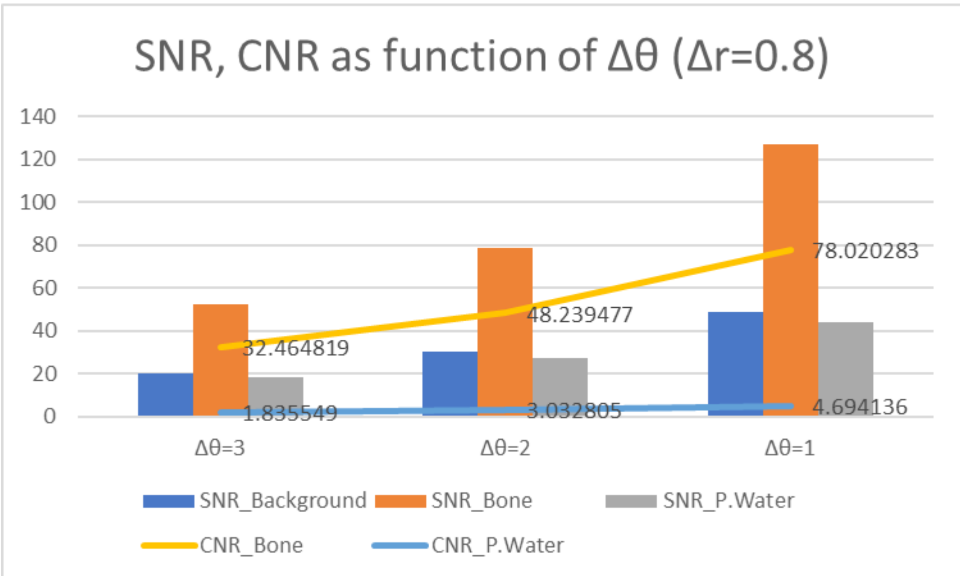


Figure 7:SNR, CNR as function of Δθ, for Δr=0.8

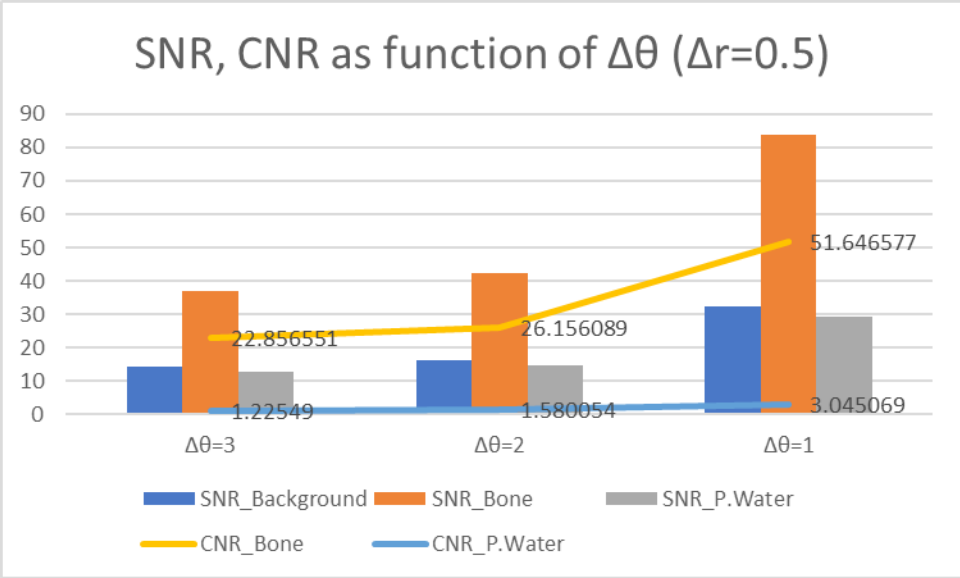


Figure 8: SNR, CNR as function of Δθ, for Δr=0.5

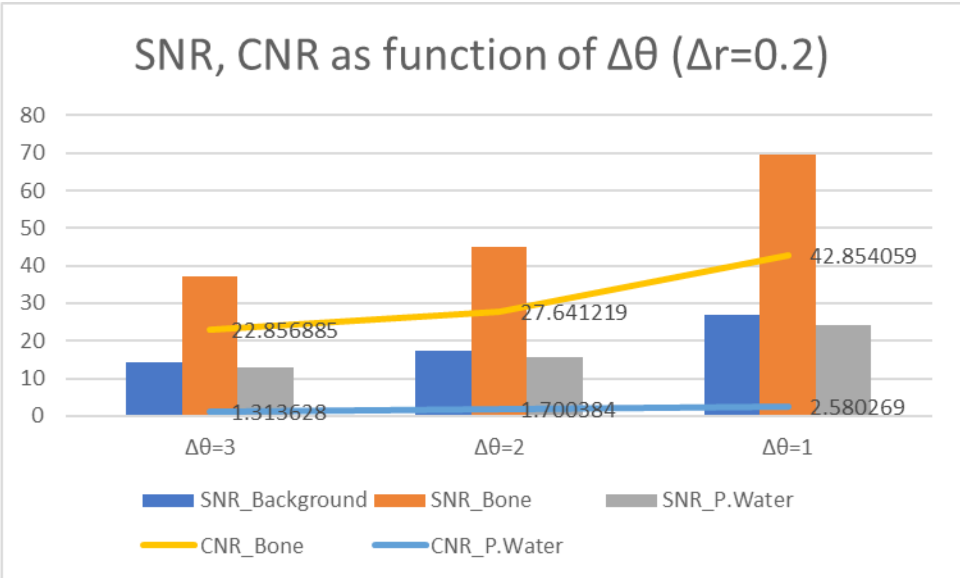


Figure 9: SNR, CNR as function of Δθ, for Δr=0.2

Figure 9 represents in practice the optimal reconstruction image we obtained for the combinations of angular sampling and projection sampling.

As a Function of Projection Modification Filters

The following graph simulates the change in values as a function of the display modification filters (see *Table 7):*

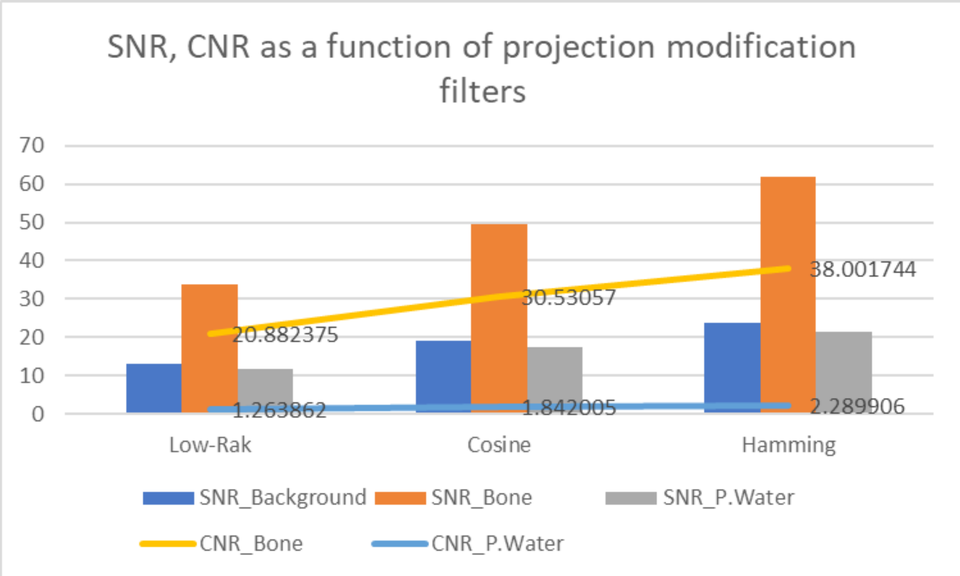


Figure 10: SNR, CNR as a function of projection modification filters

It is obvious that for the Hamming filter higher SNR and CNR values result in all cases. So we actually have a better picture than with the other two filters.

As a Function of Filter Cut-off Frequency

Finally, the latest graph is presented showing the changes as a function of the cutoff frequencies used for the Hamming filter.

The diagram *Figure 11* confirms that for a lower cut-off frequency we have a clearer picture. In practice the noise is at high levels, so it makes sense. On the other hand, however, a very low cut-off frequency makes the image blurry. Consistently the ideal limit is somewhere between the values we experimented with.

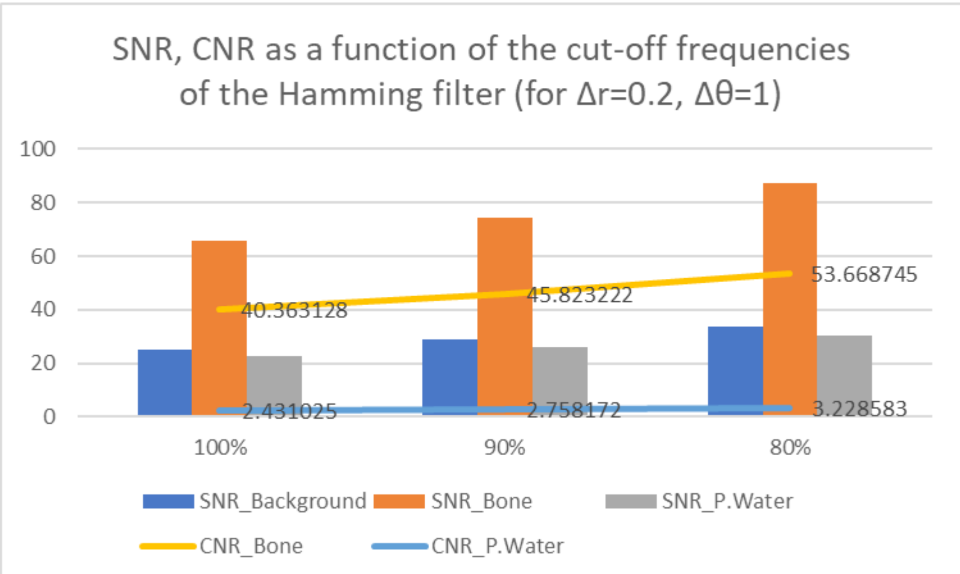


Figure 11: SNR, CNR as a function of the cut-off frequencies of the Hamming Filter

2nd Question

From *Figure 10*, it appears quantitatively that the Hamming filter is more efficient than the other two, as it gives higher SNR and CNR values in combination. Confirmed the reconstructed image was highly satisfactory.

When it comes to choosing the cut-off frequency for the hamming filter, we notice that the lower the frequency response, the higher the CNR and SNR values. Which makes sense because we've already mentioned that noise is mainly at high frequencies. So by reducing the cut-off frequency, we cut more and more high frequencies, so more and more noise. However, here the question arises that although noise is "cut", however the image begins to blur reducing the cut-off frequency, thus having a negative effect on spatial resolution.

Again, the differences in the human eye for the specific image and the specific cut-off frequencies were not very visible. Therefore we would choose the middle option 90%. In any case, within this framework we could set the limit range for cut-off frequency, so that the image does not become blurred.

3rd Question

In practice, we would consider that a numerical value (SNR, CNR) in combination with an image result, would be more desirable than an ESF, LSF chart. Intuitively, framing them in a ratio (numerical values and graphs), we think more intense fluctuations exist for the SNR, CNR.

The LSF index expresses the traceability of structures.

References

1. https://www.mathworks.com/help/images/ref/fanbeam.html
2. https://www.mathworks.com/help/images/ref/ifanbeam.html
3. https://www.mathworks.com/help/images/fan-beam-projection-data.html
4. https://www.mathworks.com/help/images/examples/reconstructing-an-image-from-projection-data.html
5. Gulliksrud, Stokke, and Trægde Martinsen, “How to Measure CT Image Quality.”