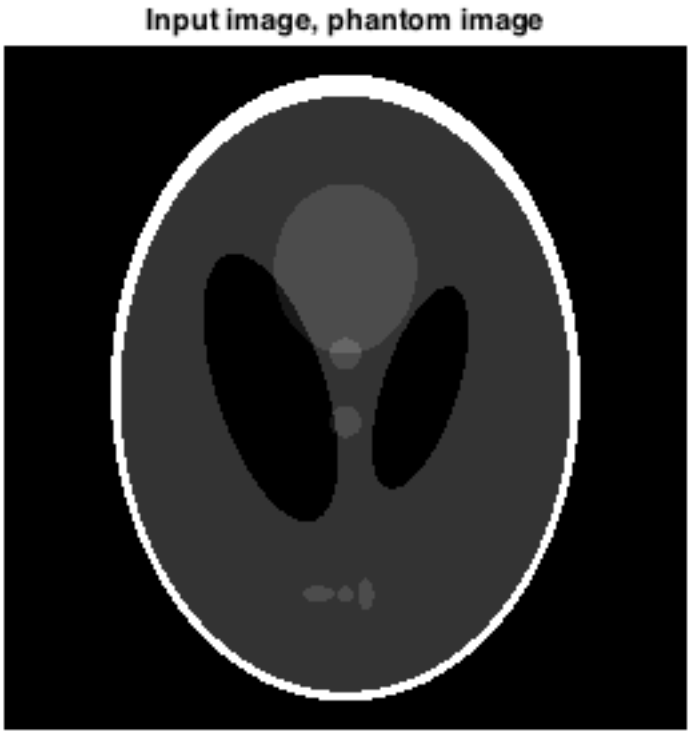
X-Ray Computed Tomography : Filtered Backprojection

Table of Contents

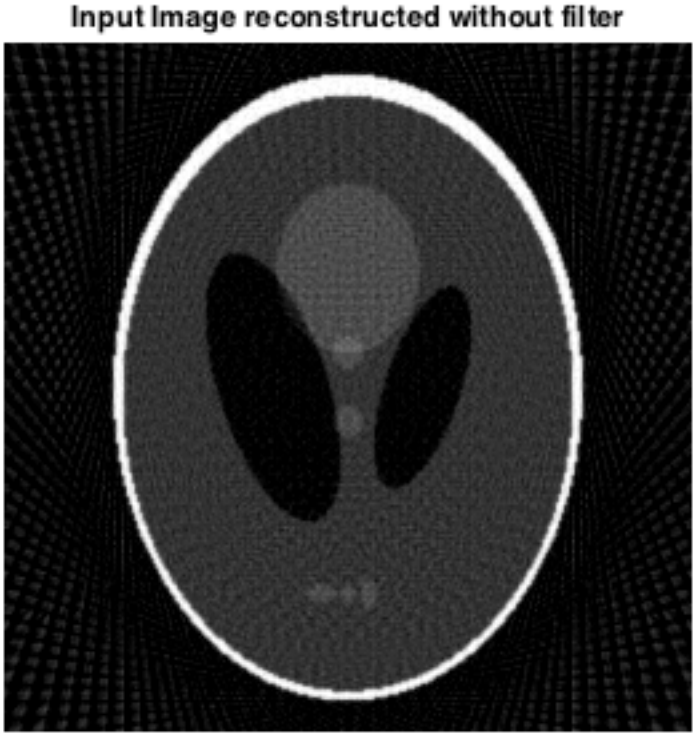
Question 2a

```
input_image = phantom(256);
theta = 0:3:177;
theta_count = length(theta);
[A, xp] = radon(input_image, theta);
figure(); imshow(input_image); title('Input image, phantom image')
```



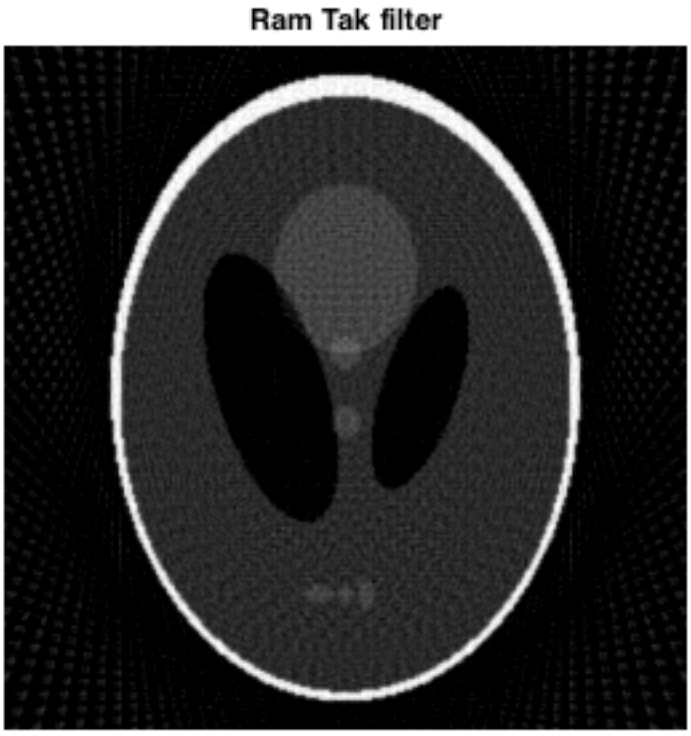
We take the inverse radon transform of the input image, first without using any additional filters. We can clearly see that there is some sort of noise/distortion introduced in the image after reconstruction. The image is given below.

```
B = iradon(A, theta);
figure(); imshow(B); title('Input Image reconstructed without filter');
```



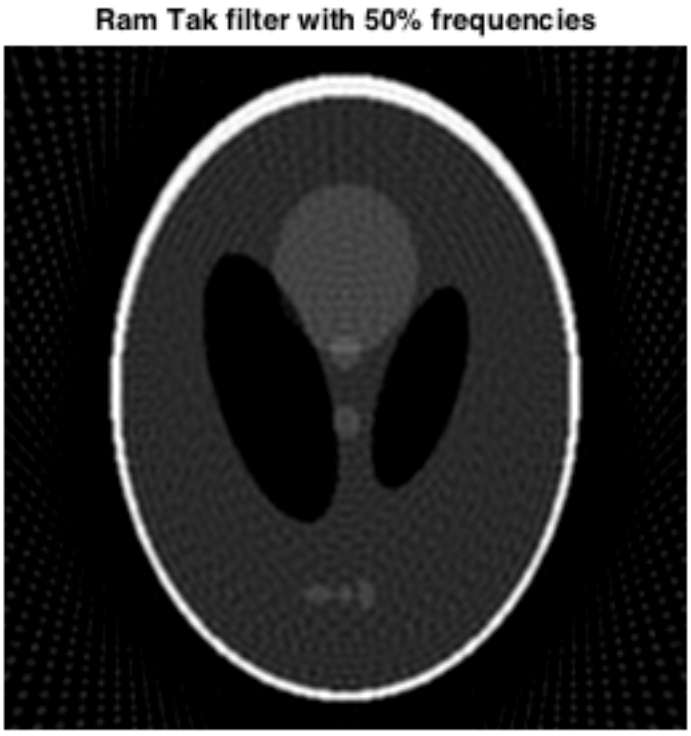
Ram Tak filter

```
my_filter = myFilter(1, 1);
output_image = ApplyFilter(input_image, my_filter);
figure(); imshow(output_image); title('Ram Tak filter');
```



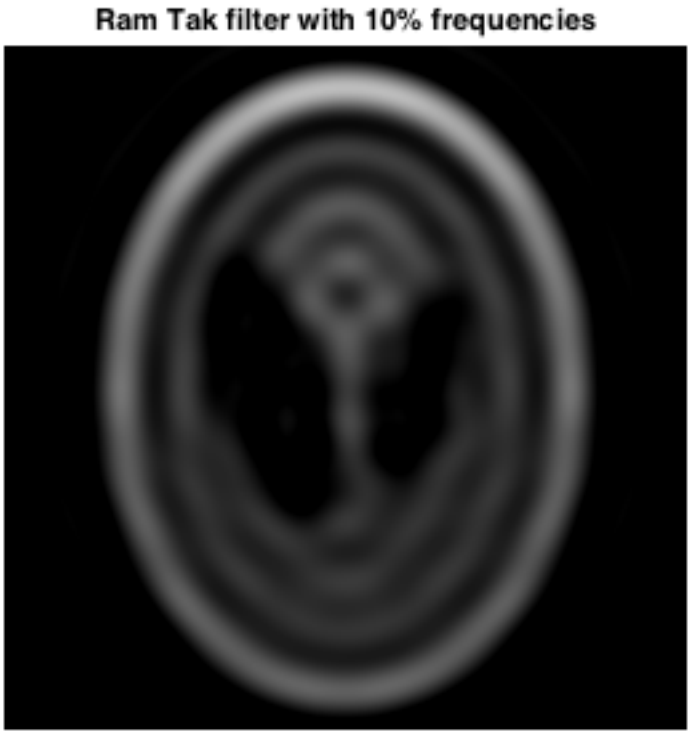
Ram Tak filter with 50% of max frequency

```
my_filter = myFilter(1, 0.5);
output_image = ApplyFilter(input_image, my_filter);
figure(); imshow(output_image); title('Ram Tak filter with 50% frequencies');
```



Ram Tak filter with smoothed image(10% of max frequency)

```
my_filter = myFilter(1, 0.1);
output_image = ApplyFilter(input_image, my_filter);
figure(); imshow(output_image); title('Ram Tak filter with 10% frequencies');
```



Shepp-Logan filter

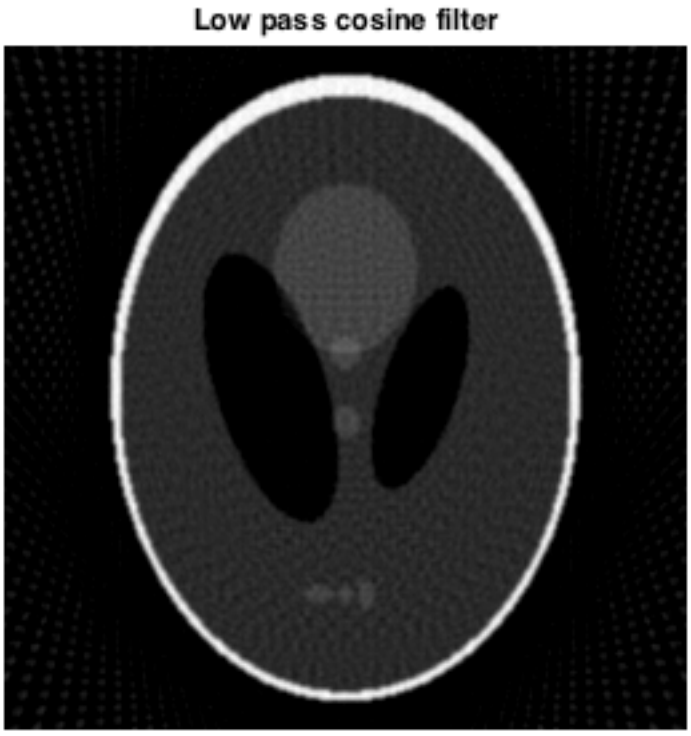
```
my_filter = myFilter(2, 1);
output_image = iradon(A, theta, 'linear', 'Shepp-Logan', 1.0, size(image,1));
figure(); imshow(output_image); title('Shepp-Logan filter');
```

Ram Tak filter with 10% frequencies

Shepp-Logan filter

Low pass cosine filter

```
my_filter = myFilter(3, 1);
output_image = ApplyFilter(input_image, my_filter);
figure(); imshow(output_image); title('Low pass cosine filter');
```

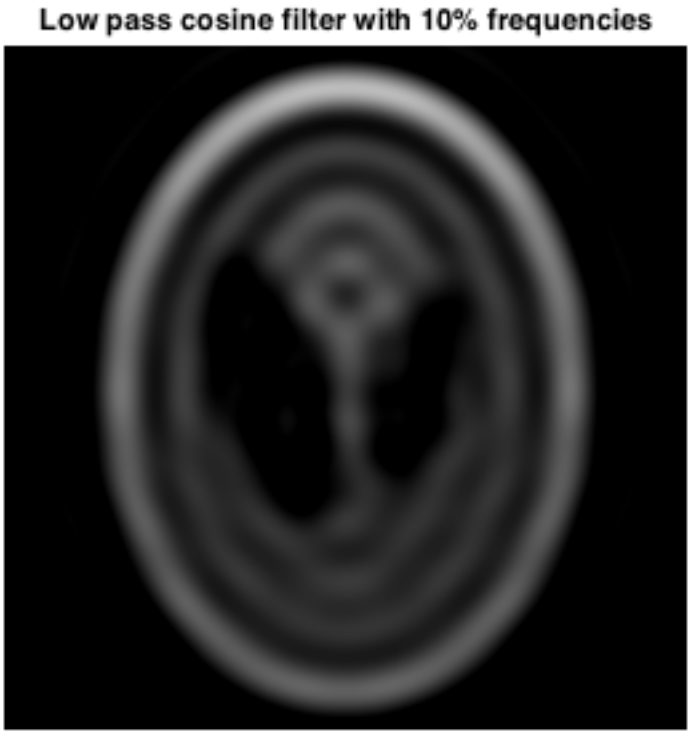


Low pass cosine filter with 50% of max frequency

```
my_filter = myFilter(3, 0.5);
output_image = ApplyFilter(input_image, my_filter);
figure(); imshow(output_image); title('Low pass cosine filter with 50% frequencies
```

Low pass cosine filter with smoothed image(10% of max frequency)

```
my_filter = myFilter(3, 0.1);
output_image = ApplyFilter(input_image, my_filter);
figure(); imshow(output_image); title('Low pass cosine filter with 10% frequencies
```



Observations

- We clearly see that the input image reconstructed without filtering has introduced some noise. Many filters have been subsequently used in attempt to smoothen the image.
- First we use the Ram Tak filter. We see some improvement in that noise, though the improvement is very low.
- When we take just 50% of the frequencies, we see that the image is smoothed(it is visible better at the white edge, where it is easy that the edge is not that sharp).
- Since the image blurring isnt that visible in the 50% image, we also show the 10% image, where we have used only the lowest 10% of the frequencies to reconstruct the image. The image is highly softened, as would be visible from the reconstruction.
- When we compare between multiple filters, cosine filter causes the most smoothing(removes noise the best) and Ram Tak filter removes the least noise(can be easily seen by comparing the ram tak and cosine filters.

Published with MATLAB® R2014b