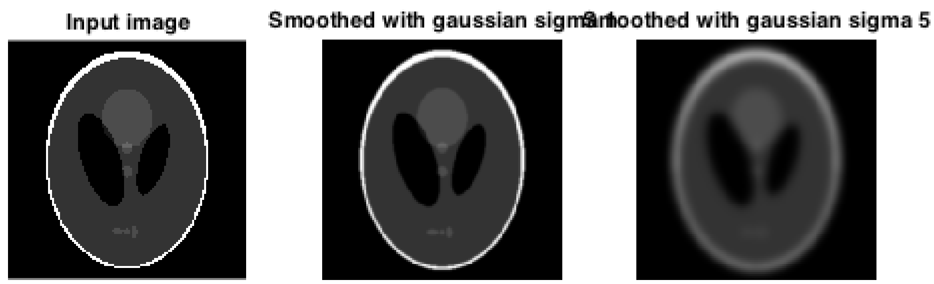
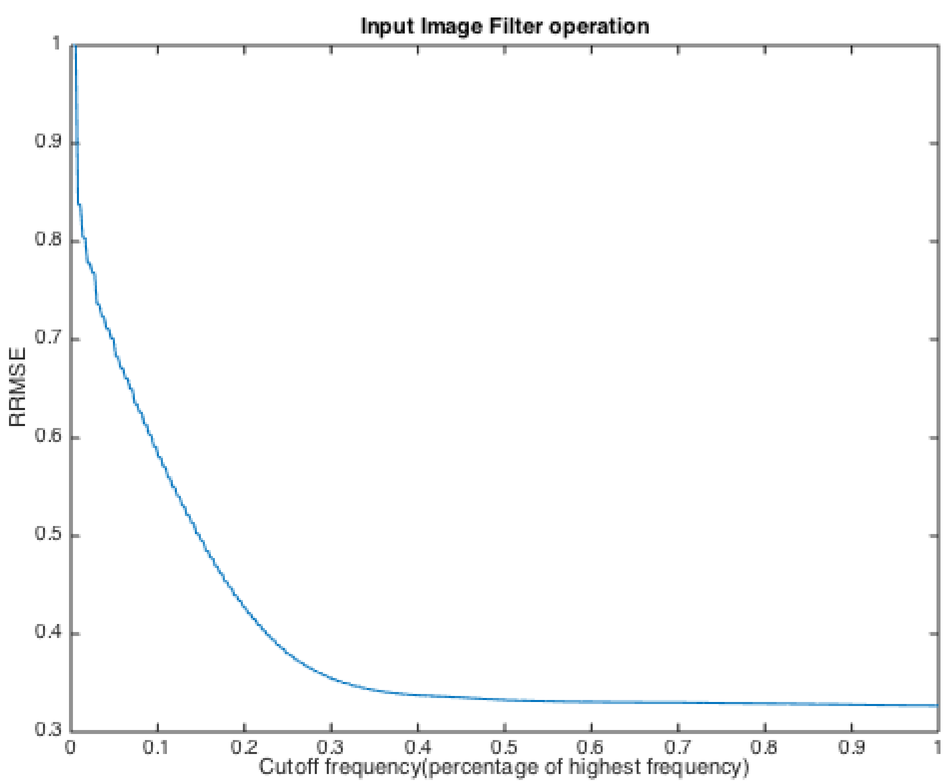
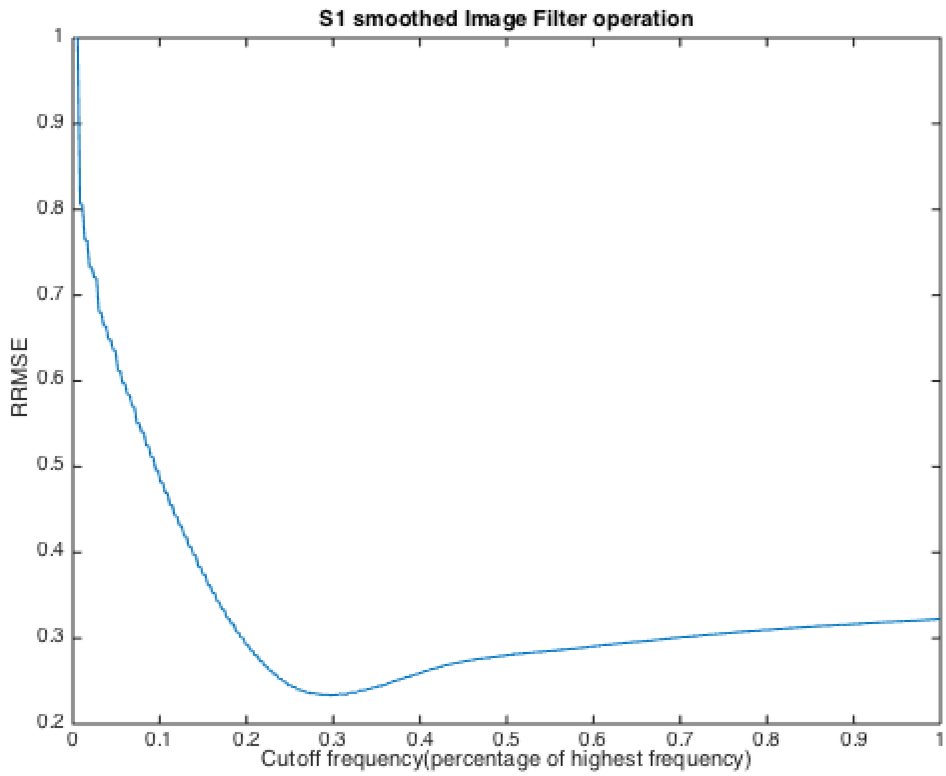
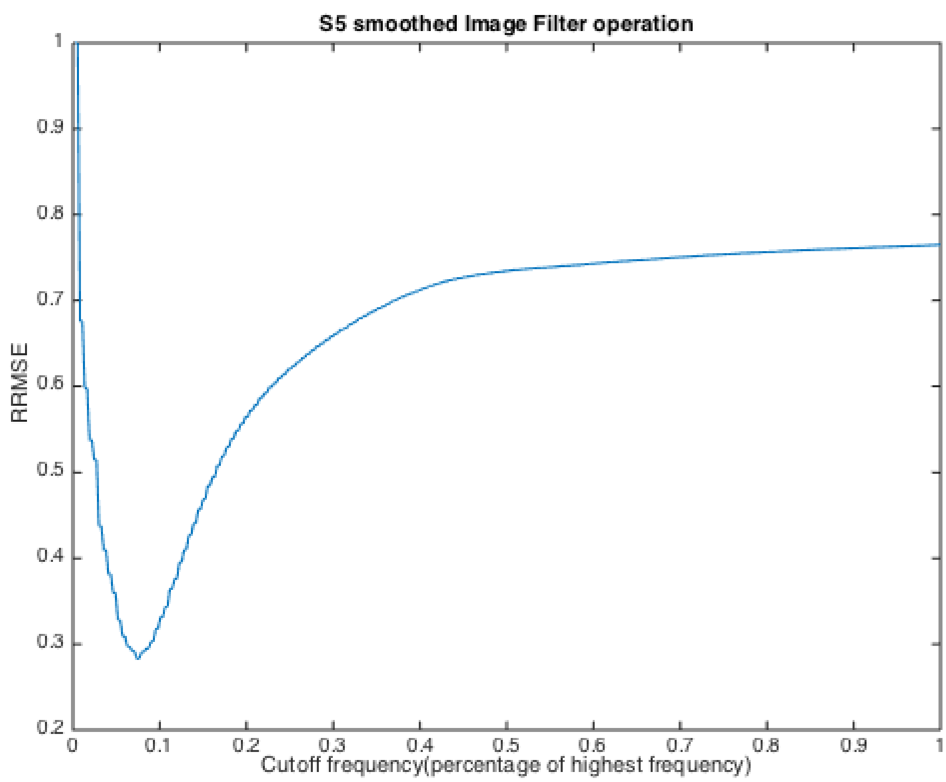
Question 2c

```
S0 = phantom(256);
mask = fspecial ('gaussian', 11, 1);
S1 = conv2 (S0, mask, 'same');
mask = fspecial ('gaussian', 51, 5);
S5 = conv2 (S0, mask, 'same');
% Note S1 is blurred with Gaussian of sigma 1, S5 is blurred with sigma 5

figure();
subplot(1,3,1), imshow(S0), title('Input image')
subplot(1,3,2), imshow(S1), title('Smoothed with gaussian sigma 1')
subplot(1,3,3), imshow(S5), title('Smoothed with gaussian sigma 5')

input_array = [S0, S1, S5];
theta = 0:3:177;
for i = 1:3
    if (i==1)
        image = S0;
        title_name = 'Input Image Filter operation';
        a_number = 0;
    elseif (i==2)
        image = S1;
        title_name = 'S1 smoothed Image Filter operation';
        a_number = 1;
    elseif (i==3)
        image = S5;
        title_name = 'S5 smoothed Image Filter operation';
        a_number = 5;
    end
    [A] = radon(image, theta);
    RRMSE_array = [];
    for l = 1:367;
        my_filter = myFilter(1, l/367);
        output_image = ApplyFilter(S0, my_filter);
        RRMSE = sqrt(sum(sum((output_image - image).^2)))/sqrt(sum(sum(image.^2)))
        RRMSE_array = [RRMSE_array, RRMSE];
    end
    figure();
    plot((1:367)/367, RRMSE_array); title(title_name); xlabel('Cutoff frequency(pe
end
```


Input image
Smoothed with gaussian sigma 1
Smoothed with gaussian sigma 5


Input Image Filter operation
RRMSE
Cutoff frequency(percentage of highest frequency)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


S1 smoothed Image Filter operation
RRMSE
Cutoff frequency(percentage of highest frequency)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


S5 smoothed Image Filter operation
RRMSE
Cutoff frequency(percentage of highest frequency)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

Observation

- For the input image with no smoothing, the graph is obvious. The input image has a lot of edges and hence a lot of high frequeny components as a part of the signal. If we consider the high frequency components in the reconstruction, then we will get better results, lesser result. Hence the error has reduced continuously as the frequency components have increased.
- For the S1 case, we have smoothed the signal. Hence the input signal/image does not have higher frequency components. But obviously when we start from including very few frequencies, the error is huge. Hence the error falls to some minimum value, approximately close to the highest 'input signal' frequency component still present in the input signal. After this frequency, the higher frequencies would be noise, and now we are modelling noise as well. Hence our error increases as we include more and more frequencies.
- Results are similar to the S1 case. But here we see that the minima point is lower as compared to the previous case, which make sense since this signal has been smoothed more. So there are lower frequency components in the input image, hence we reach the minima very early in the frequency axis(x axis).

Published with MATLAB® R2014b