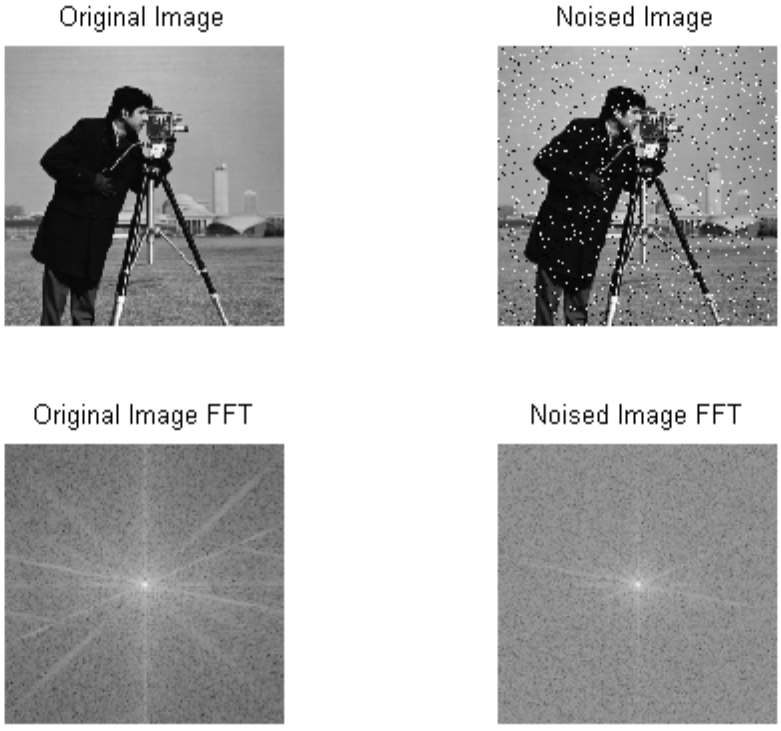
Table of Contents

Loading the image and computing the FT

```
cm = imread('cameraman.tif');
cm_noised = imnoise(cm,'salt & pepper');

cm_fft = fftshift(fft2(cm));
cm_noised_fft = fftshift(fft2(cm_noised));
```

Plotting the Spectrum along with the image

```
figure;
subplot(2,2,1);
imshow(cm);
title('Original Image');

subplot(2,2,2);
imshow(cm_noised);
title('Noised Image');

subplot(2,2,3);
imshow(uint8(255/max(max(log(abs(cm_fft))))*log(abs(cm_fft))));
title('Original Image FFT');

subplot(2,2,4);
imshow(uint8(255/max(max(log(abs(cm_noised_fft))))*log(abs(cm_noised_fft))))
title('Noised Image FFT');
```

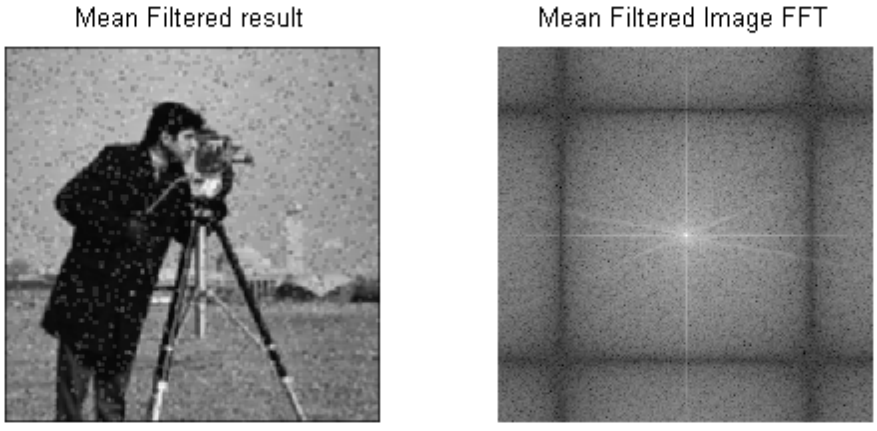


Filtering in Time Domain

```
% Mean Filtering

mean = (1/9)*ones(3,3);
cm_noised_mean = uint8(conv2(cm_noised, mean));
cm_noised_mean_fft = fftshift(fft2(cm_noised_mean));

figure;

subplot(1,2,1);
imshow(cm_noised_mean);
title('Mean Filtered result');

subplot(1,2,2);
imshow(uint8(255/max(max(log(abs(cm_noised_mean_fft))))*log(abs(cm_noised_mean_fft
title('Mean Filtered Image FFT');

% Median Filtering

cm_noised_median = medfilt2(cm_noised);
cm_noised_median_fft = fftshift(fft2(cm_noised_median));

figure;
```

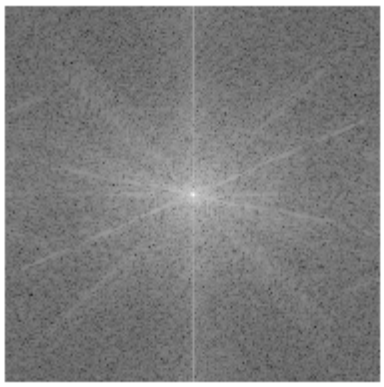
```
subplot(1,2,1);
imshow(cm_noised_median);
title('Median Filtered result');

subplot(1,2,2);
imshow(uint8(255/max(max(log(abs(cm_noised_median_fft))))*log(abs(cm_noised_median
title('Median Filtered Image FFT');

% Mode Filtering

cm_noised_mode = colfilt(cm_noised, [5 5], 'sliding', @mode);
cm_noised_mode_fft = fftshift(fft2(cm_noised_mode));

figure;

subplot(1,2,1);
imshow(cm_noised_mode);
title('Mode Filtered result');

subplot(1,2,2);
imshow(uint8(255/max(max(log(abs(cm_noised_mode_fft))))*log(abs(cm_noised_mode_fft
title('Mode Filtered Image FFT');
```

```
Warning: CONV2 on values of class UINT8 is

obsolete.
        Use CONV2(DOUBLE(A),DOUBLE(B)) or
        CONV2(SINGLE(A),SINGLE(B)) instead.
```



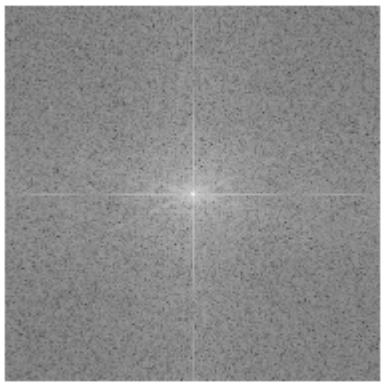
Median Filtered result

Median Filtered Image FFT



Mode Filtered result

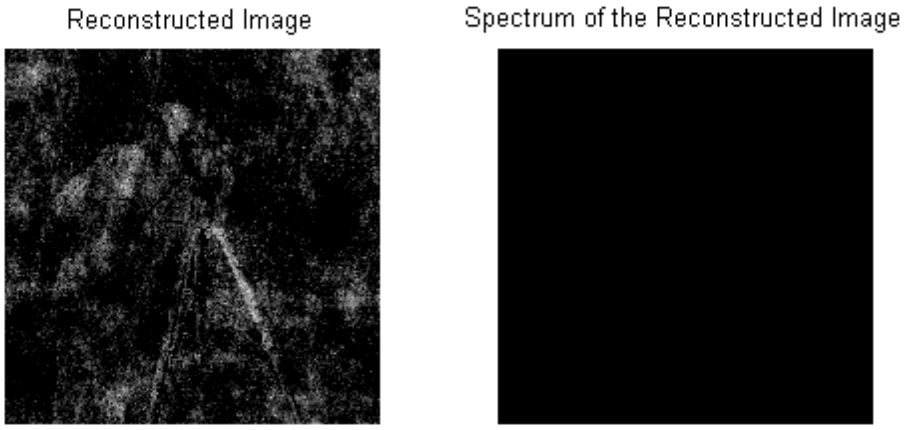
Mode Filtered Image FFT



Frequency domain filtering of Salt and Pepper(Impulse) Noise

```
freq_spec = cm_noised_fft;
freq_spec_med = medfilt2(abs(freq_spec));
mask = (abs(freq_spec)-freq_spec_med)>0.5;
freq_spec(mask) = 0;
reconstructed =  (ifft2(ifftshift(freq_spec)));
reconstructed = uint8(255/max(max(reconstructed))*reconstructed);

figure;

subplot(1,2,1);
imshow(reconstructed);
title('Reconstructed Image');

subplot(1,2,2);
imshow(uint8(255/max(max(log(abs(freq_spec)))*log(abs(freq_spec)))));
title('Spectrum of the Reconstructed Image');
```

Warning: Displaying real part of complex input.



Conclusion

With the Magnitude Spectra of image with and without the salt and pepper noise, we can see few patterns in the spectra of clean image which can no more be seen in the noisy spectra.

Median Filter is best for Salt and Pepper noise, and the mean filter works well for Gaussian noise.

In frequency domain we dont quite get the same results as spatial domain, for salt and pepper noise, spatial domain median filter is much better.

Published with MATLAB® R2014a