Homework 3- Some Noises

Berkant Güneş 20160807013

Original image:

%%upload image ---MAİN-----

```
clc;

close all;

[fname, path] = uigetfile('*.jpg', 'Select an Image');

fname = strcat(path,fname);

im = imread(fname);

im2 = im2double(imread(fname));



%gaussian_noise(im);       %add gaussian noise

%saltPepper_noise(im);    %add salt and pepper noise

%medianFilter(im,4);       %removed gaussian and salt&pepper noise

%periodic_noise_and_removed(im);   %periodic noise and removed periodic noise
```

Salt and Pepper noise:

```
function out = saltPepper_noise(im)

J = im;

p3= 0.05;

x = rand(size(J));%n*n boyutun 0-1 arası elemanları olan matrix üretir

d = find(x < p3/2);%find() sıfır olmayan indisleri verir

J(d) = 0; %Minimum valued

d= find(x >= p3/2 & x < p3);

J(d) = 255; %Maximum(saturated) valueimshow(J)


imwrite(J, 'salt&pepper.jpg', 'quality', 95);%%salt and pepper noise image save


figure;

subplot(121)

imshow(im)

title('Original Image')

subplot(122)

imshow(J)

title('Salt&Pepper Noise')

end
```

Output1:

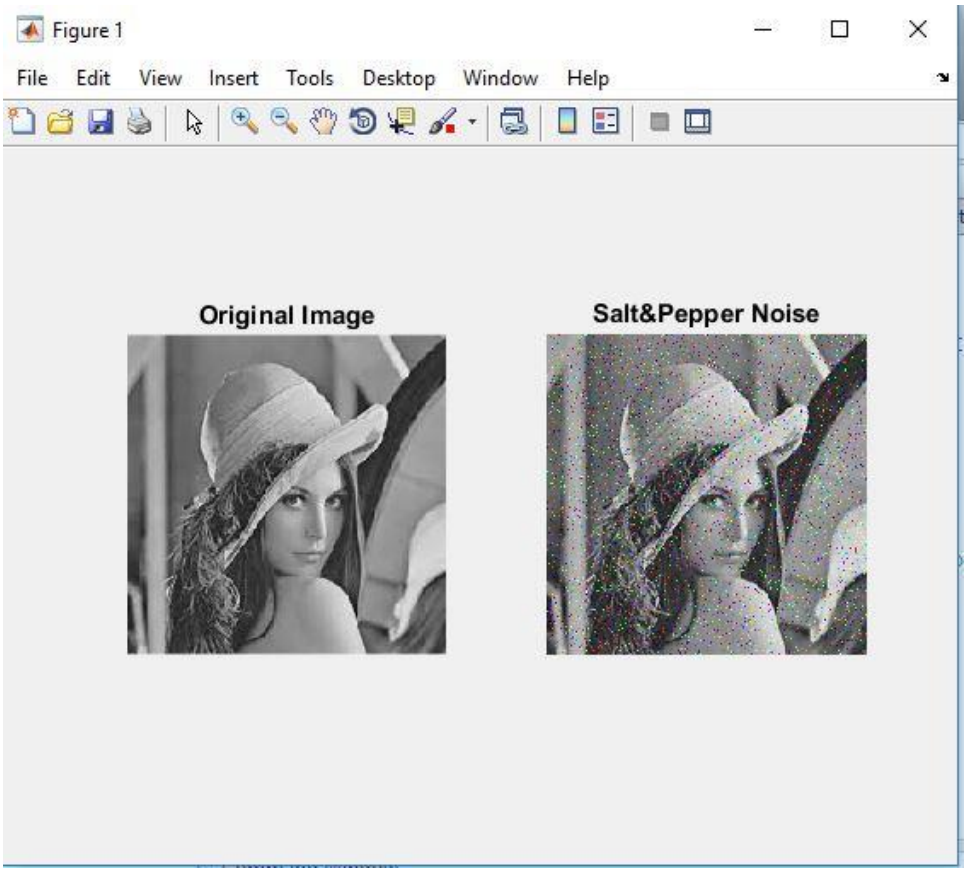
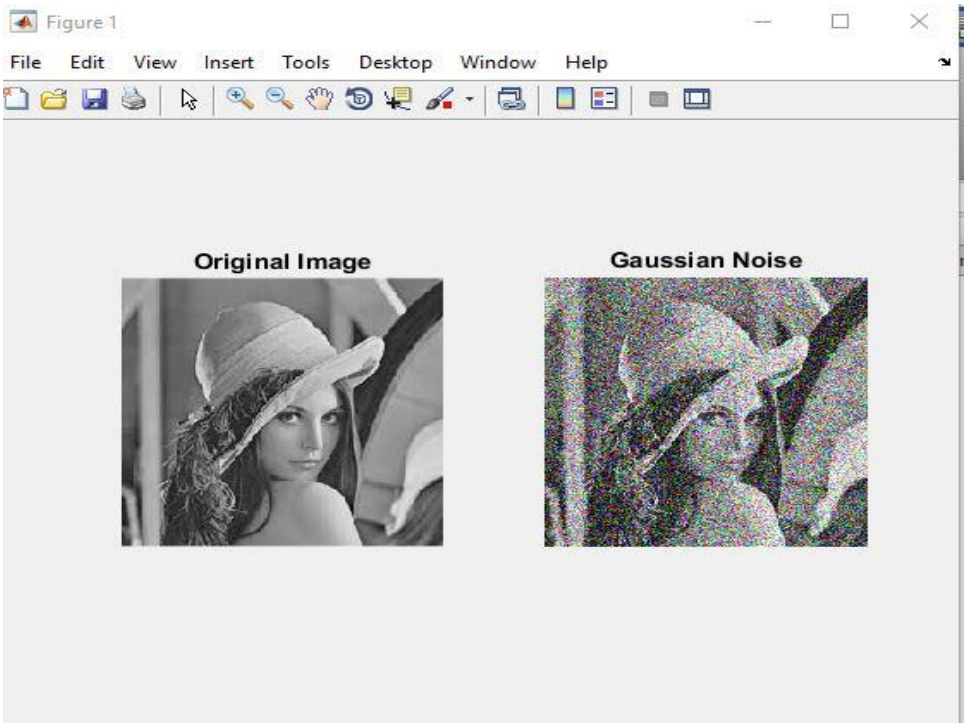


Figure 1
File Edit View Insert Tools Desktop Window Help
Original Image
Salt&Pepper Noise

Gaussian noise:

```
function out = gaussian_noise(im)

 J = im;

p3= 0;p4 = 0.05;

J = im2double(J);

b = J + sqrt(p4)*randn(size(J)) + p3;


imwrite(b, 'gaussian.jpg', 'quality', 95);%%gaussian noise image save


figure;

subplot(121)

imshow(im)

title('Original Image')

subplot(122)

imshow(b)

title('Gaussian Noise')

 end
```

Output2:

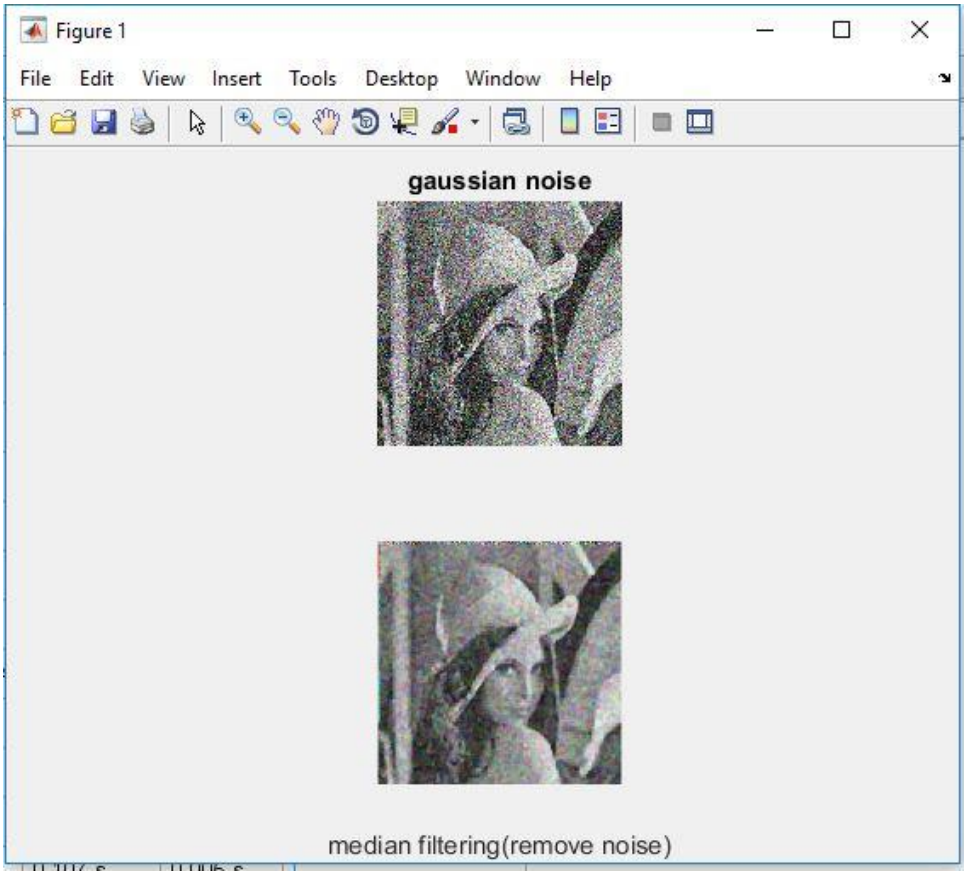


Removed Salt & Pepper noise and Gaussian noise with median filter technique:

```
%-removes salt and pepper noise from image.

%-removes gaussian noise from image.

%With median filter, gaussian and salt and pepper noise are removed


function out = medianFilter(Im,n)

[height,width] = size(Im);

Im1 = double(Im);

out = Im1;

center = round((n+1)/2);

for i = 1:height-n+1

   for j = 1:width-n+1

      temp = Im1(i:i+n-1,j:j+n-1);

      ele = temp(1,:);

      for count = 2:n

         ele = [ele,temp(count,:)];
```

```
        end
        med = median(ele);
        out(i+center,j+center) = med;
    end
end

out = uint8(out);
figure
subplot(2,1,1)
imshow(Im)
subplot(2,1,2)
imshow(out)
xlabel('median filtering(remove noise)')
end
```

Output3:

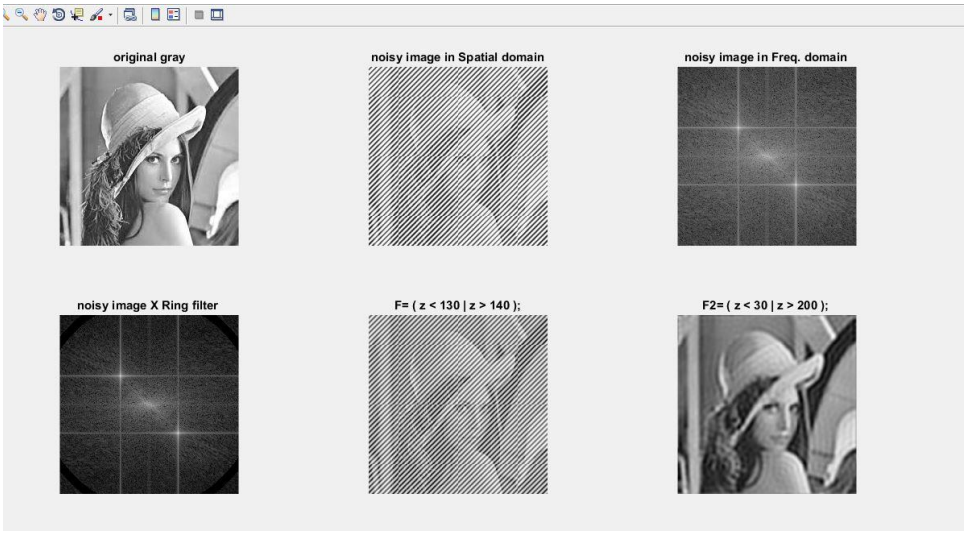


Periodic noise and removed periodic noise:

```
function [out] = periodic_noise_and_removed(im)
A = rgb2gray(im);
[w,q] = size (A);
[x, y] = meshgrid(1:q, 1:w);
s1 = 1 + sin ( x + y );
A2 = im2double(A) + s1 ;
tgpf = fftshift ( fft2 (A2) );
subplot(2,3,1); imshow(mat2gray (A*1.2)) ;title ('original gray') ;

subplot(2,3,2); imshow((A2/2));title('noisy image in Spatial domain');

subplot(2,3,3);imshow (mat2gray( log ( abs(tgpf) ) ) ) ; title('noisy image in Freq. domain ') ;
```

```
k = sqrt ( ( x - q/2 ).^2 + ( y - w/2 ).^2 );

F= ( k < 130 | k > 140 );

resf = tgpf .* F ;

resi = ifft2 ( resf );

subplot(2,3,4);

imshow (mat2gray( log (1+ abs(resf) ) ) );

title ('noisy image X Ring filter ') ;

subplot (2,3,5);

imshow (mat2gray ( log (1+ abs(resi) ) ) ) ;

title('F= ( z < 130 | z > 140 );') ;


F2= ( k < 30 | k > 200 );

resf2 = tgpf .* F2 ;

resi2 = ifft2 ( resf2 );

subplot(2,3,6);

imshow (mat2gray ( log (1+ abs(resi2) ) ) ) ;

title('F2= ( z < 30 | z > 200 ); ') ;


end
```

Output4:


original gray
noisy image in Spatial domain
noisy image in Freq. domain
noisy image X Ring filter
F= (z < 130 | z > 140);
F2= (z < 30 | z > 200);