
Problem Set 5

problem2

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                               summary                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% gaussian filter works best for gaussian noise (of course..)
% wavelet filter is not very good for all these three filter.
% median filter seems good to salt and pepper
% bilateral filter doesn't work for salt and pepper.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% implementation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% original picture named f, gaussian noisy picture named fng,
% saltPepper
% nosiy picture named fsp;
f = double(building);
sigma1 = sigma(5,f);
sigma2 = sigma(20,f);

fng1 = f + sigma1*randn(size(f));
fng2 = f + sigma2*randn(size(f));

fsp020 = imnoise(im2double(building),'salt & pepper',0.20);

figure; imagesc(f);title('the original picture')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%      ISOTROPIC GAUSSIAN      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% name them hgau;
[fx,fy] = size(fng1);
fgau = cell(1,200);
errng1 = zeros(200,1);
k = 0;

for i = 0.1:0.1:20
    k = k+1;
    hgau = gau(i);
    fgau{1,k} = conv2(fng1,hgau,'same');
    errng1(k,1) = MSE(fgau{1,k},f);
end
indexng1 = find(errng1 == min(errng1(:)))
figure;
subplot(2,1,1);imagesc(fng1);title('gaussian 5dB noise')
subplot(2,1,2);imagesc(fgau{1,13});title('gaussian var = 2.6')
```

```

% for SNR=5dB, the best variance of the gaussian filter will be 2.6.

errng2 = zeros(200,1);
k = 0;

for i = 0.1:0.1:20
    k = k+1;
    hgau = gau(i);
    fgau{1,k} = conv2(fng2,hgau,'same');
    errng2(k,1) = MSE(fgau{1,k},f);
end
indexng2 = find(errng2 == min(errng2(:)))
figure;
subplot(2,1,1);imagesc(fng2);title('gaussian 20dB noise')
subplot(2,1,2);imagesc(fgau{1,2});title('gaussian var = 0.4')

% for SNR=20dB, the best variance of the gaussian filter will be 0.4.

errng3 = zeros(200,1);
k = 0;

for i = 0.1:0.1:20
    k = k+1;
    hgau = gau(i);
    fgau{1,k} = conv2(fsp020,hgau,'same');
    errng3(k,1) = MSE(fgau{1,k},f);
end
indexng3 = find(errng3 == min(errng3(:)))
figure;
subplot(2,1,1);imagesc(fng2);title('salt and pepper noise')
subplot(2,1,2);imagesc(fgau{1,1});title('gaussian var = 0.2')

% for Salt and pepper noise, the best variance of the gaussian filter
will be 0.2.
% gaussian filter works not very good.

% ranges have been iterately shrinked to this narrowness.
% the picture will be well denoised meanwhile not too smoothing or too
blur

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%      Scale by scale wavelet thresholding      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% wavelet analyzed subcells named fw;
N = 2;

[fw1,fw2,fw3,fw4] = analysis(fng1, h0, h1, N);
%[fw1,fw2,fw3,fw4] = analysis(fng2, h0, h1, N);
%[fw1,fw2,fw3,fw4] = analysis(fsp020, h0, h1, N);
[fo1,fo2,fo3,fo4] = analysis(f, h0, h1, N);
err = zeros(10,1);
index = zeros(N,1);

```

```

j = 1;
for i = N:-1:1
    k= 0;
    for value = 1:j:1000
        k = k+1;
        fw2{1,i}(abs(fw2{1,i})<value) = 0;
        fw3{1,i}(abs(fw3{1,i})<value) = 0;
        fw4{1,i}(abs(fw4{1,i})<value) = 0;

        fresult =
synthesis(fw1{1,i},fw2{1,i},fw3{1,i},fw4{1,i},g0,g1);

        if(i == 1)
            err(k,1) = MSE (fresult,f);
        else
            err(k,1) = MSE(fresult,fo1{1,i-1});
        end
    end
    index(i,:) = min(find(err == min(err(:)))));
end

value = 1 + j.*[index-ones(N,1)]
%value = 300*ones(2,1);
[fw1,fw2,fw3,fw4] = analysis(fng1, h0, h1, N);
for i = N:-1:1

    %fw1{1,i}(fw1{1,i}<value(i,1)) = 0;
    fw2{1,i}(abs(fw2{1,i})<value(i,1)) = 0;
    fw3{1,i}(abs(fw3{1,i})<value(i,1)) = 0;
    fw4{1,i}(abs(fw4{1,i})<value(i,1)) = 0;
end

fw = synthesis(fw1,fw2,fw3,fw4,g0,g1);
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(fw);title('wavelet
    for SNR=5dB')

% the thresholds for SNR=5dB, scale N = 2 are 430 and 347 for each
layers.

[fw1,fw2,fw3,fw4] = analysis(fng2, h0, h1, N);
[fo1,fo2,fo3,fo4] = analysis(f, h0, h1, N);
err = zeros(10,1);
index = zeros(N,1);

j = 1;
for i = N:-1:1
    k= 0;
    for value = 1:j:1000
        k = k+1;
        %fw1{1,i}(fw1{1,i}<value) = 0;
        fw2{1,i}(abs(fw2{1,i})<value) = 0;
        fw3{1,i}(abs(fw3{1,i})<value) = 0;

```

```

        fw4{1,i}(abs(fw4{1,i})<value) = 0;

        fresult =
        synthesis(fw1{1,i},fw2{1,i},fw3{1,i},fw4{1,i},g0,g1);

        if(i == 1)
            err(k,1) = MSE (fresult,f);
        else
            err(k,1) = MSE(fresult,fo1{1,i-1});
        end
    end
    index(i,:) = min(find(err == min(err(:)))));
end

value = 1 + j.*[index-ones(N,1)];
%value = 300*ones(2,1);
[fw1,fw2,fw3,fw4] = analysis(fng1, h0, h1, N);
for i = N:-1:1

    %fw1{1,i}(fw1{1,i}<value(i,1)) = 0;
    fw2{1,i}(abs(fw2{1,i})<value(i,1)) = 0;
    fw3{1,i}(abs(fw3{1,i})<value(i,1)) = 0;
    fw4{1,i}(abs(fw4{1,i})<value(i,1)) = 0;
end

fw = synthesis(fw1,fw2,fw3,fw4,g0,g1);
figure;
subplot(2,1,1);imagesc(fng2);subplot(2,1,2);imagesc(fw);title('wavelet
    for SNR=20dB')
% the thresholds for SNR=20dB, scale N = 2 are 42 and 22 for each
    layers.

[fw1,fw2,fw3,fw4] = analysis(fsp020, h0, h1, N);
[fo1,fo2,fo3,fo4] = analysis(f, h0, h1, N);
err = zeros(10,1);
index = zeros(N,1);

j = 0.001;
for i = N:-1:1
    k= 0;
    for value = 0.001:j:1
        k = k+1;
        %fw1{1,i}(fw1{1,i}<value) = 0;
        fw2{1,i}(abs(fw2{1,i})<value) = 0;
        fw3{1,i}(abs(fw3{1,i})<value) = 0;
        fw4{1,i}(abs(fw4{1,i})<value) = 0;

        fresult =
        synthesis(fw1{1,i},fw2{1,i},fw3{1,i},fw4{1,i},g0,g1);

        if(i == 1)
            err(k,1) = MSE (fresult,f);
        else

```

```

        err(k,1) = MSE(fresult,fol{1,i-1});
    end
end
index(i,:) = min(find(err == min(err(:)))));
end

value = 0.01 + j.*[index-ones(N,1)];
%value = 300*ones(2,1);
[fw1,fw2,fw3,fw4] = analysis(fng1, h0, h1, N);
for i = N:-1:1

    %fw1{1,i}(fw1{1,i}<value(i,1)) = 0;
    fw2{1,i}(abs(fw2{1,i})<value(i,1)) = 0;
    fw3{1,i}(abs(fw3{1,i})<value(i,1)) = 0;
    fw4{1,i}(abs(fw4{1,i})<value(i,1)) = 0;

end

fw = synthesis(fw1,fw2,fw3,fw4,g0,g1);
figure;
subplot(2,1,1);imagesc(fsp020);subplot(2,1,2);imagesc(fw);title('wavelet
    for Salt and pepper')
% the thresholds for salt and pepper, this time a little bit different
% since the image has been normalized, ranging form 0-1.
% scale N = 2 are 0.0180 and 0.0340 and 22 for each layers.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%      median filtering .
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%for SNR=5dB%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fun = @(x) median(x(:));
B1 = nlfilter(fng1,[3 3],fun);
err = MSE (B1,f)
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B1);title('median
    filter of SNR = 5dB neighbor[3*3]')
% error for the 3*3 median filter is 1.7114e+03.

fun = @(x) median(x(:));
B2 = nlfilter(fng1,[9 9],fun);
err = MSE (B2,f)
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B2);title('median
    filter of SNR = 5dB neighbor[9*9]')
% error for the 9*9 median filter is 1.2684e+03.

fun = @(x) median(x(:));
B3 = nlfilter(fng1,[15 15],fun);
err= MSE (B3,f)
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B3);title('median
    filter of SNR = 5dB neighbor[15*15]')
% error for the 15*15 median filter is 1.6358e+03.

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%for SNR=20dB%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fun = @(x) median(x(:));
B1 = nlfilter(fng2,[3 3],fun);
err = MSE (B1,f)
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B1);title('median
    filter of SNR = 20dB neighbor[3*3]')
% error for the 3*3 median filter is 256.6544.

fun = @(x) median(x(:));
B2 = nlfilter(fng2,[9 9],fun);
err = MSE (B2,f)
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B2);title('median
    filter of SNR = 20dB neighbor[9*9]')
% error for the 9*9 median filter is 891.6244

fun = @(x) median(x(:));
B3 = nlfilter(fng2,[15 15],fun);
err= MSE (B3,f)
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B3);title('median
    filter of SNR = 20dB neighbor[15*15]')
% error for the 15*15 median filter is 1.3382e+03.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%for salt and pepper noise%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fun = @(x) median(x(:));
B1 = nlfilter(fsp020,[3 3],fun);
err = MSE (B1,im2double(building))
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B1);title('median
    filter of salt and pepper neighbor[3*3]')
% error for the 3*3 median filter is 0.0061.

fun = @(x) median(x(:));
B2 = nlfilter(fsp020,[9 9],fun);
err = MSE (B2,im2double(building))
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B2);title('median
    filter of salt and pepper neighbor[9*9]')
% error for the 9*9 median filter is 0.0160.

fun = @(x) median(x(:));
B3 = nlfilter(fsp020,[15 15],fun);
err= MSE (B3,im2double(building))
figure;
subplot(2,1,1);imagesc(fng1);subplot(2,1,2);imagesc(B3);title('median
    filter of salt and pepper neighbor[15*15]')
% error for the 15*15 median filter is 0.0255.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    bilateral filtering .
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Implements bilateral filtering for grayscale images.

```

```

j = 1;
k = 0;
for i = 300:320
    for z = 1:5:20
        k = k+1;
        C = bilateral(fng1,3,i,z);
        err1(k,1) = MSE (C,f);
    end
end
indexbi1 = min(find(err1 == min(err1(:))))
% after a lot of test, I choose signal = 15, sigma2 = 13

C1 = bilateral(fng1,15,15,13);
figure;
subplot(2,1,1);imagesc(fng1);title(' 5dB noisy pic')
subplot(2,1,2);imagesc(C1);title('bilateral filter of 5dB noisy pic')


j = 1;
k = 0;
for i = 10:j:20
    for z = 1:50
        k = k+1;
        C = bilateral(fng2,3,i,z);
        err(k,1) = MSE (C,f);
    end
end
indexbi2 = min(find(err == min(err(:))))
% after a lot of test, we choose signal = 11, sigma2 = 17
C2 = bilateral(fng2,3,11,17);
figure;
subplot(2,1,1);imagesc(fng2);title('20dB noisy pic');
subplot(2,1,2);imagesc(C2);title('bilateral filter of 20dB noisy pic')


j = 5;
k = 0;
for i = 50:100
    k = k+1;
    C = bilateral(fsp020,11,i,i);
    err(k,1) = MSE (C,im2double(f));
end
indexbi3 = min(find(err == min(err(:))))
% after a bunch of test, there seems no best choice for salt and
% pepper noise..
C3 = bilateral(fsp020,11,1,5);
figure;
subplot(2,1,1);imagesc(fsp020);subplot(2,1,2);imagesc(C3);title('bilateral
    filter of 20dB noisy pic')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% bilateral filter works best for 20dB gaussian noise.
% for salt and pepper it seems to blur the whole picture.

```

```
% for 5dB gaussian noise, it almost doesnt work.
```

```
indexng1 =
```

```
13
```

```
indexng2 =
```

```
2
```

```
indexng3 =
```

```
1
```

```
value =
```

```
418
```

```
371
```

```
err =
```

```
1.7245e+03
```

```
err =
```

```
1.2922e+03
```

```
err =
```

```
1.6468e+03
```

```
err =
```

```
257.9017
```

```
err =
```

```
891.5782
```

```
err =
```

```
1.3401e+03
```

```
err =  
    0.0061
```

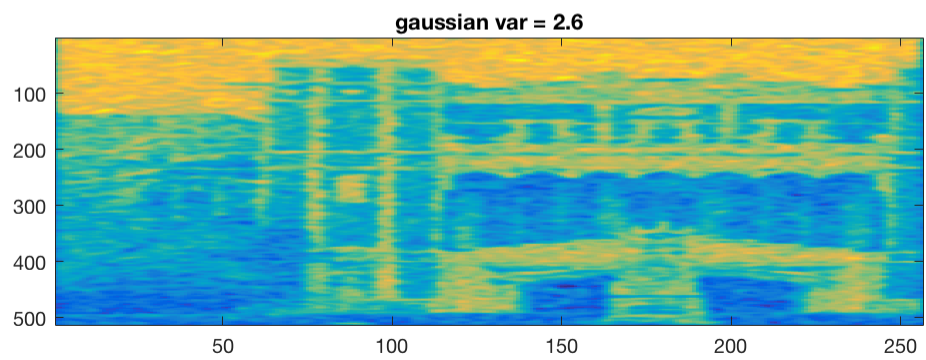
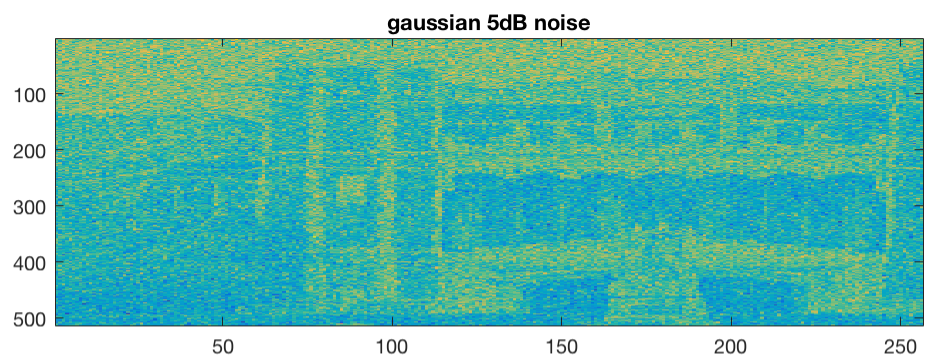
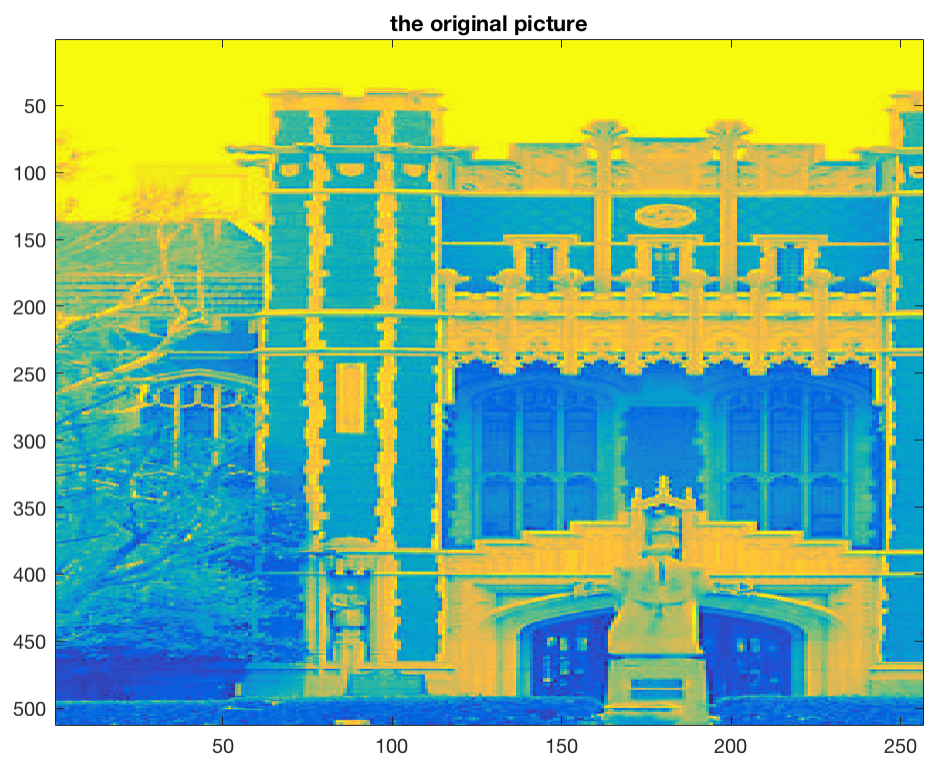
```
err =  
    0.0163
```

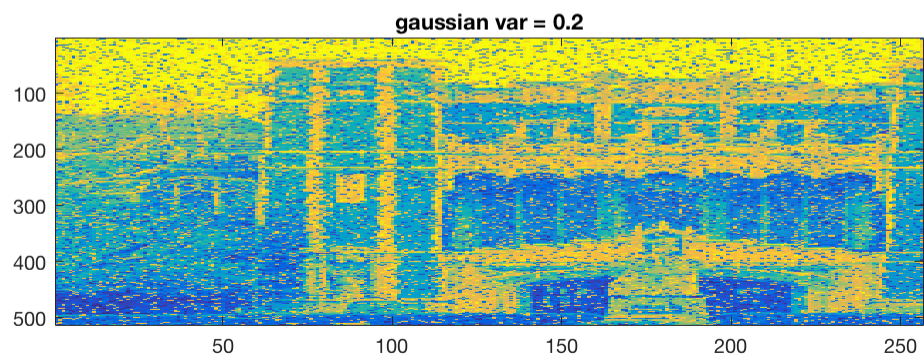
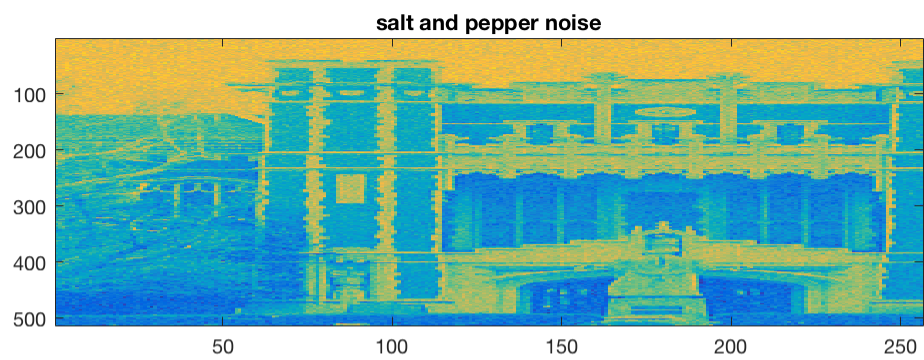
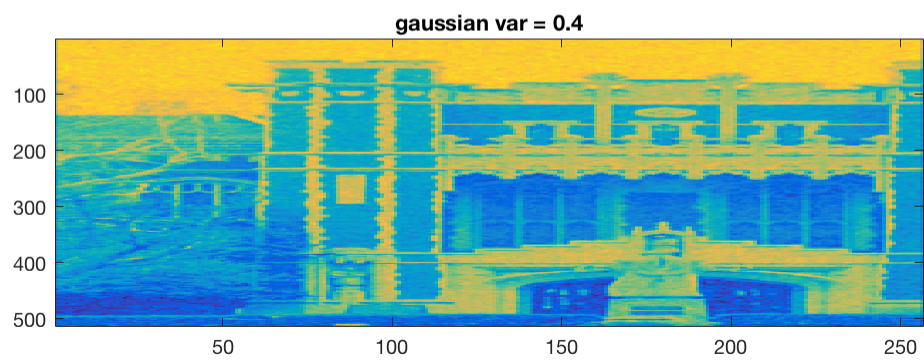
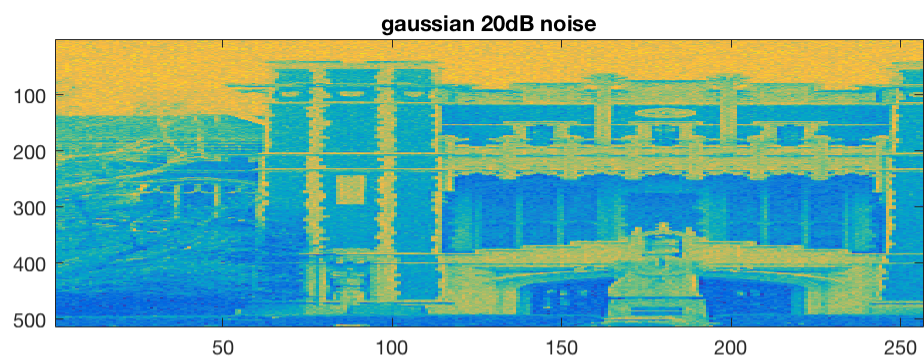
```
err =  
    0.0255
```

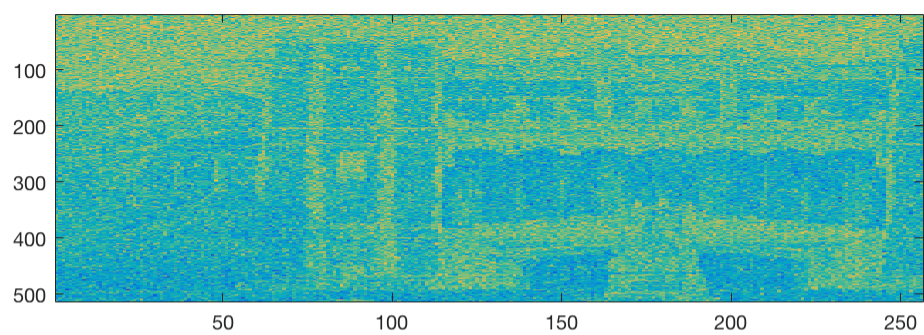
```
indexbi1 =  
    263
```

```
indexbi2 =  
    27
```

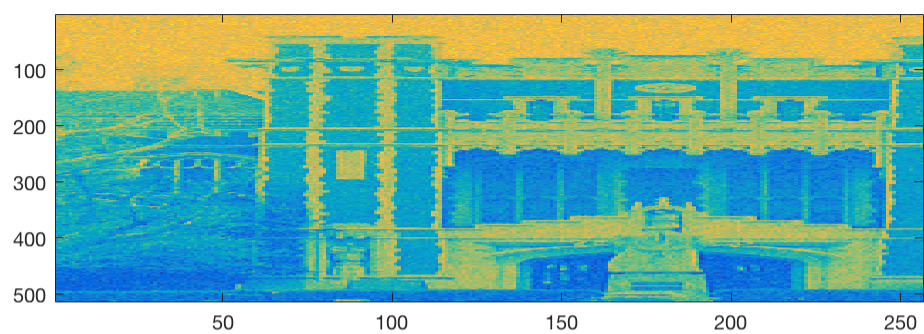
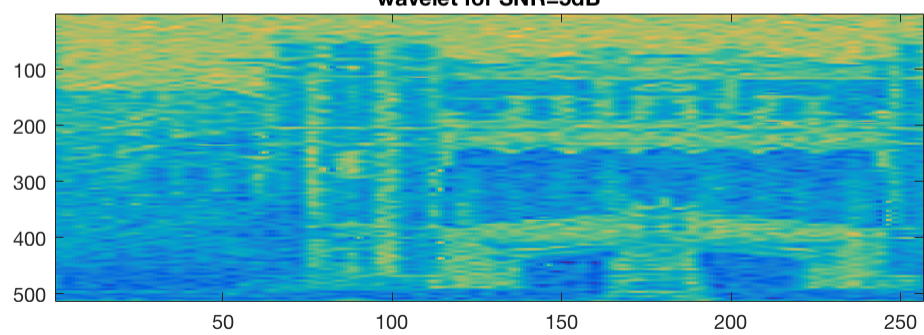
```
indexbi3 =  
    77
```



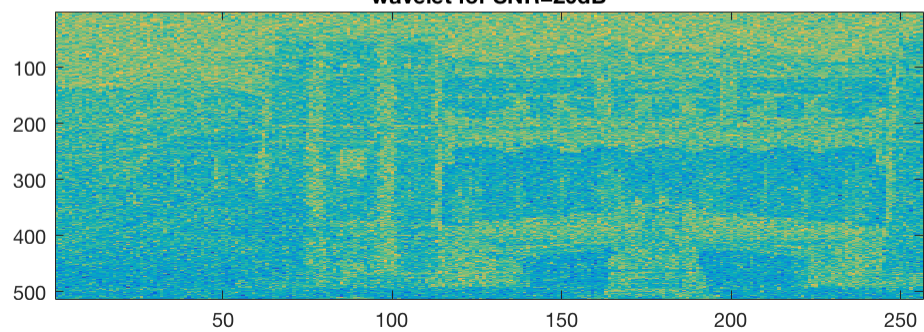


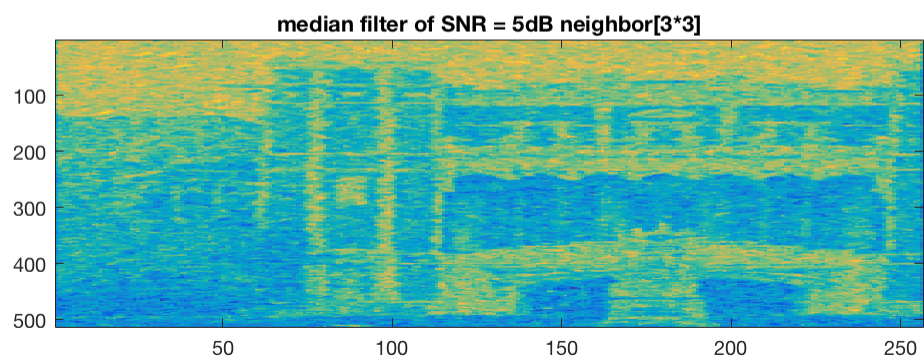
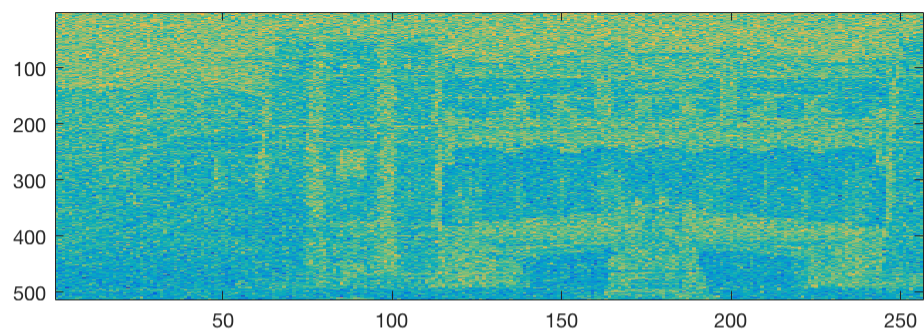
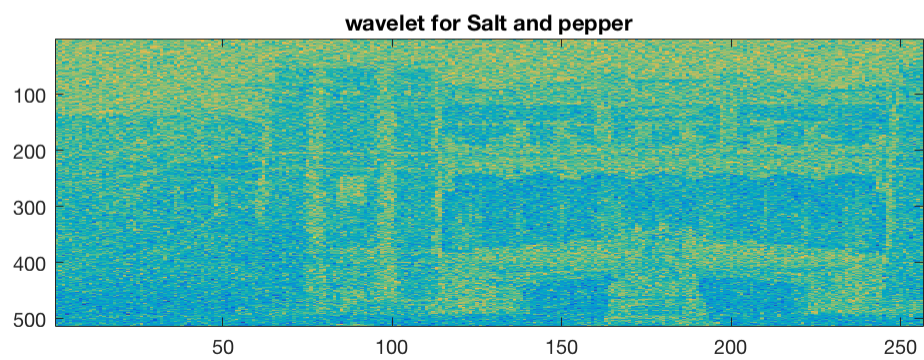
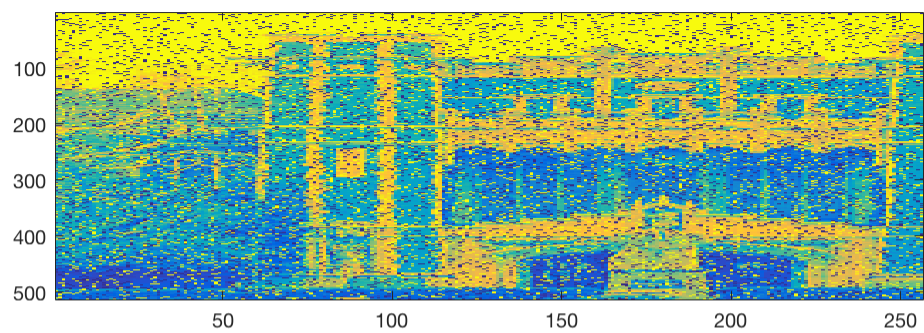


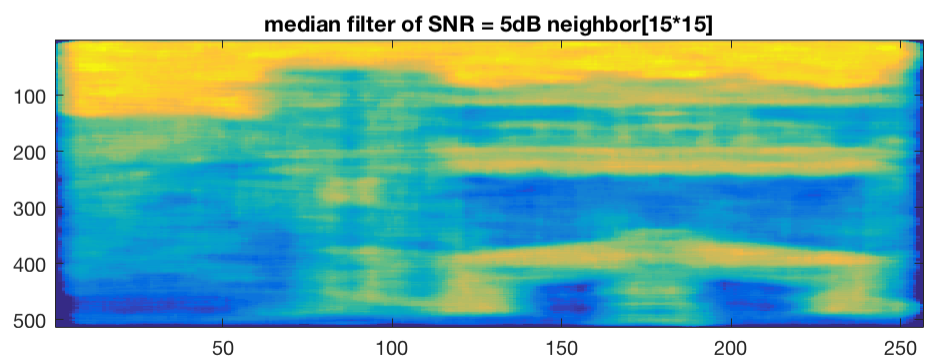
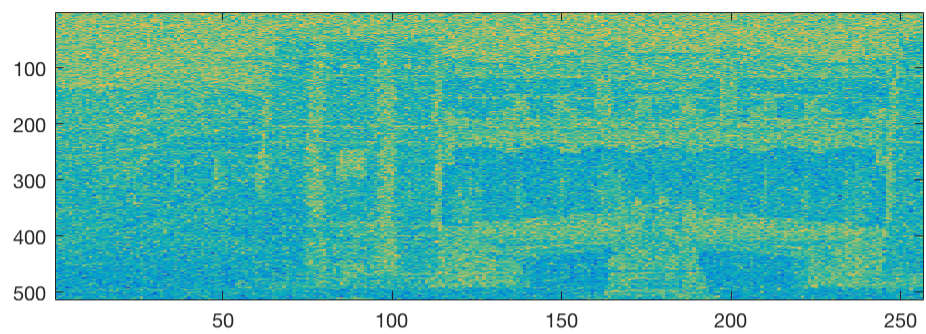
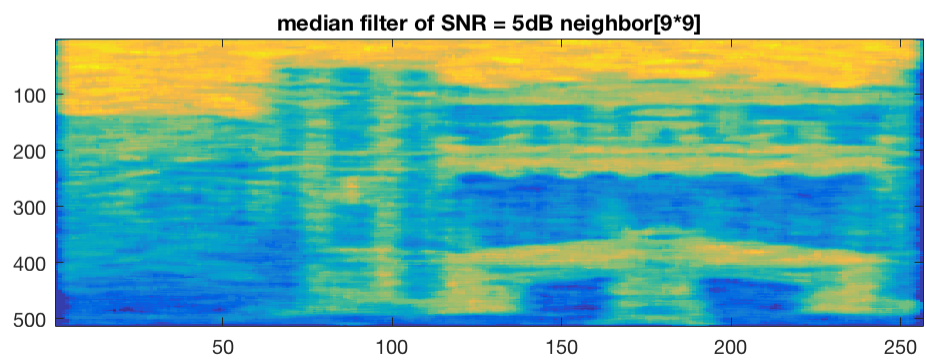
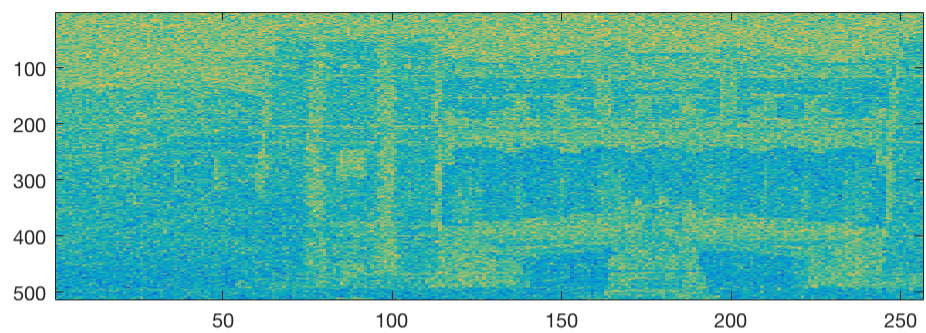
wavelet for SNR=5dB

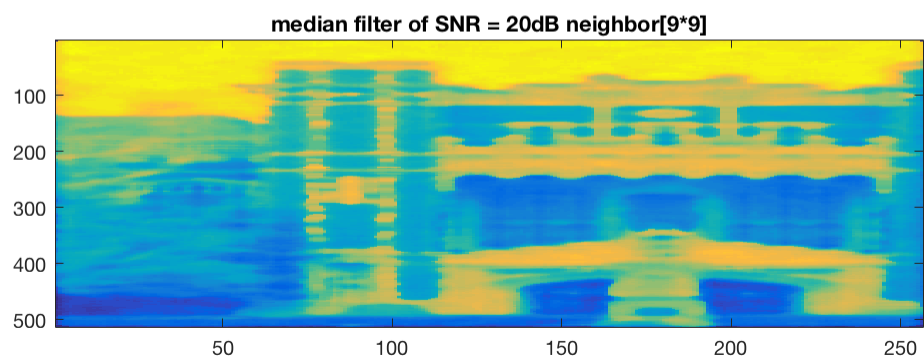
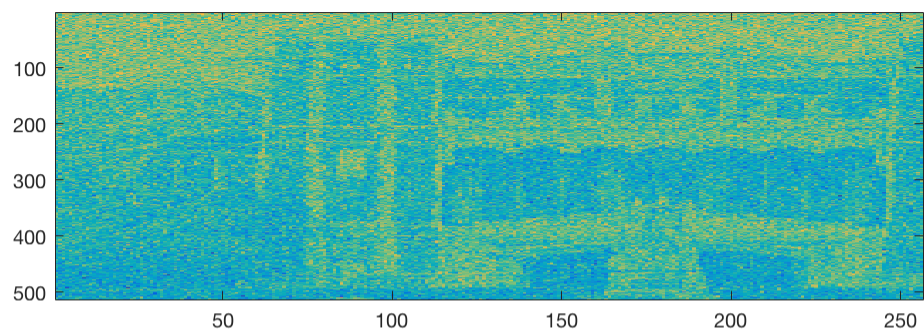
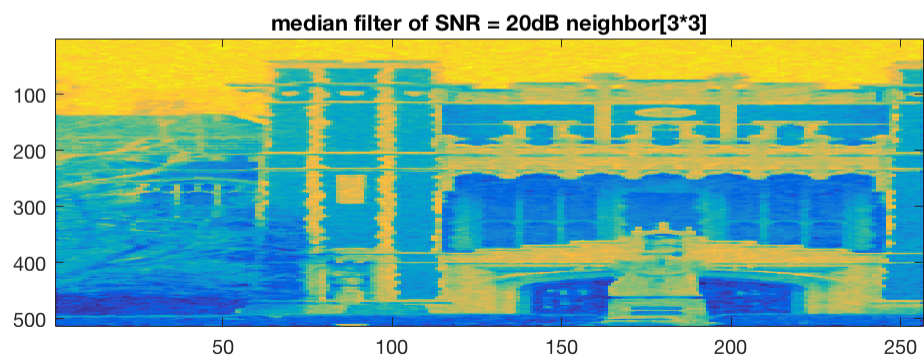
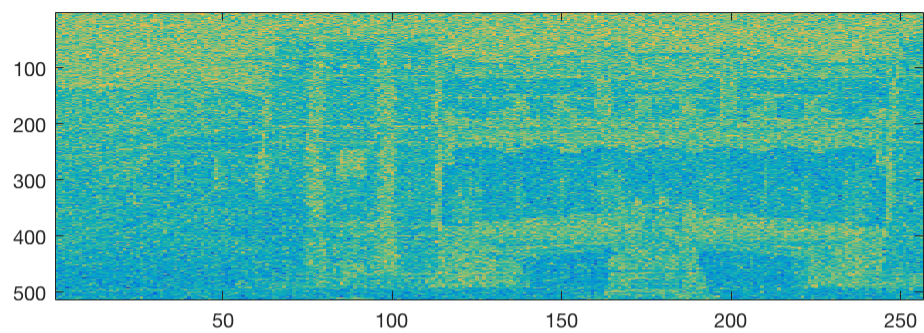


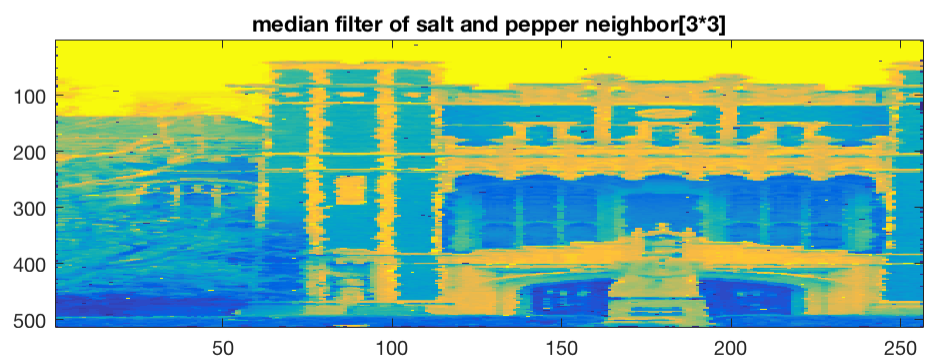
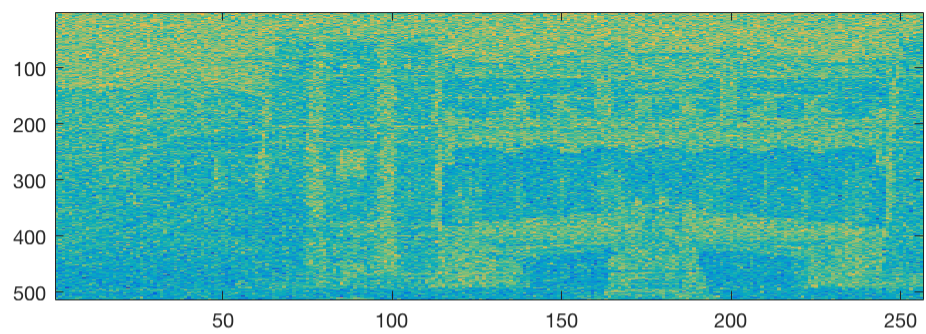
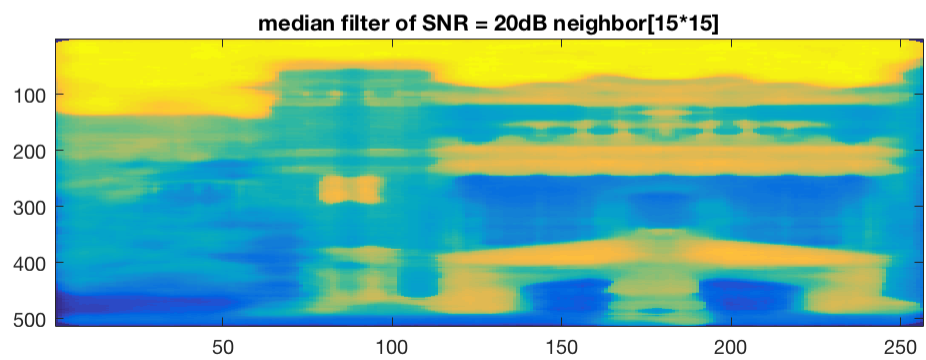
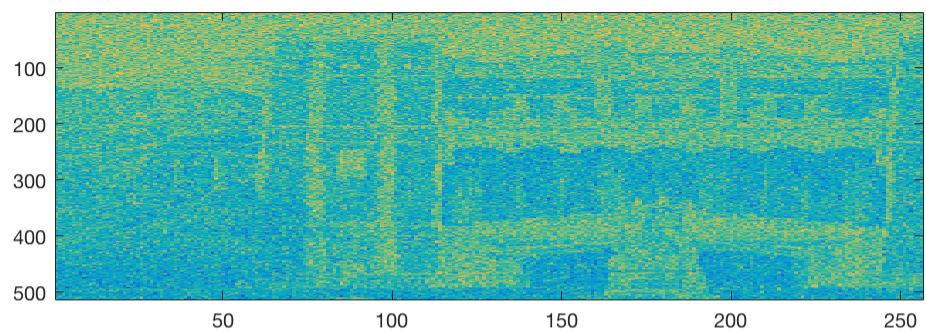
wavelet for SNR=20dB

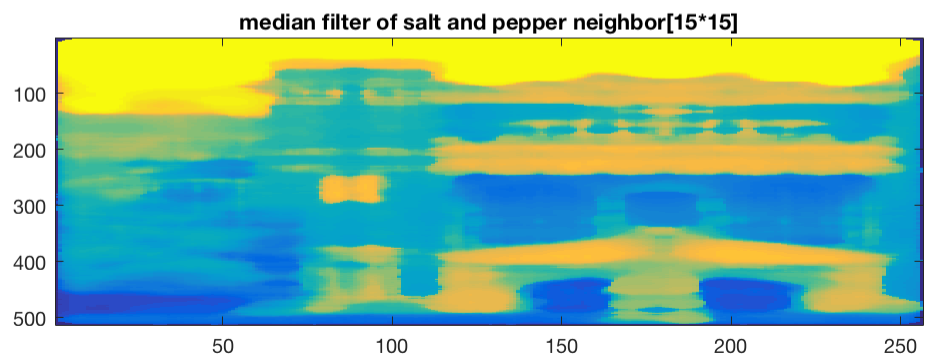
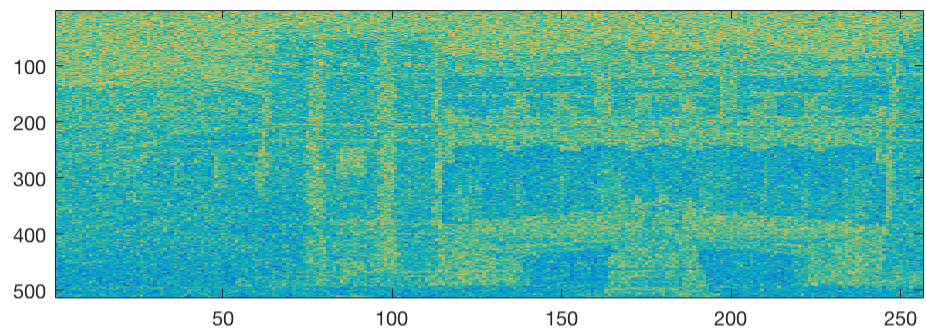
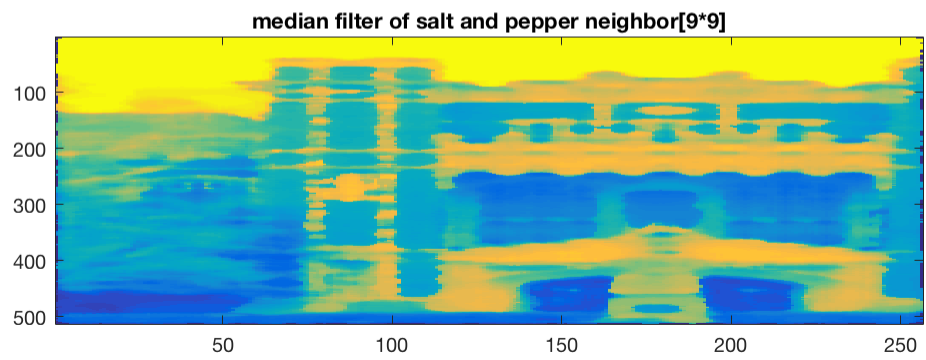
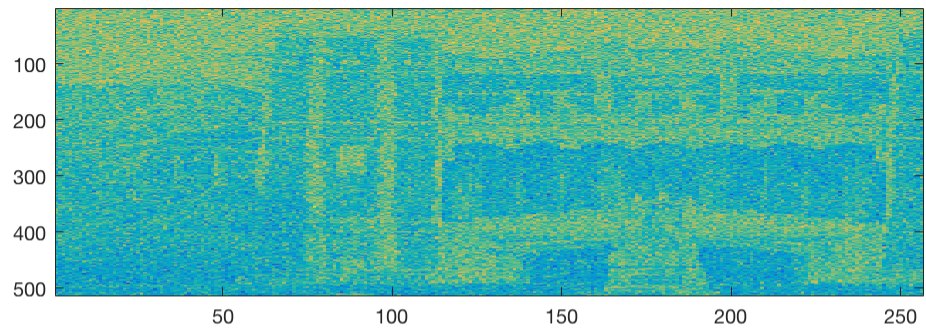


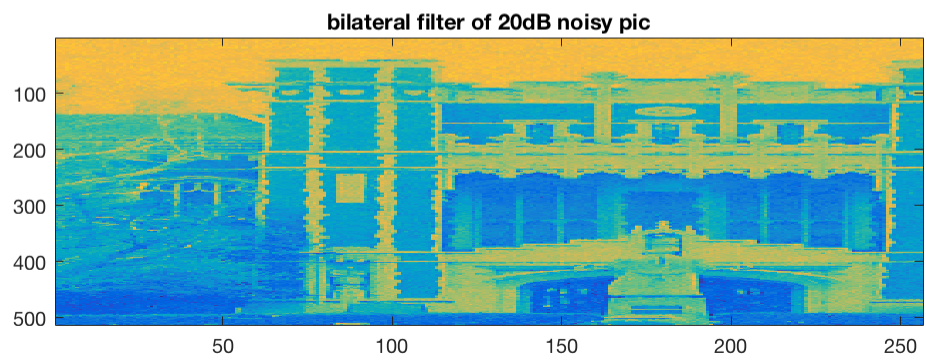
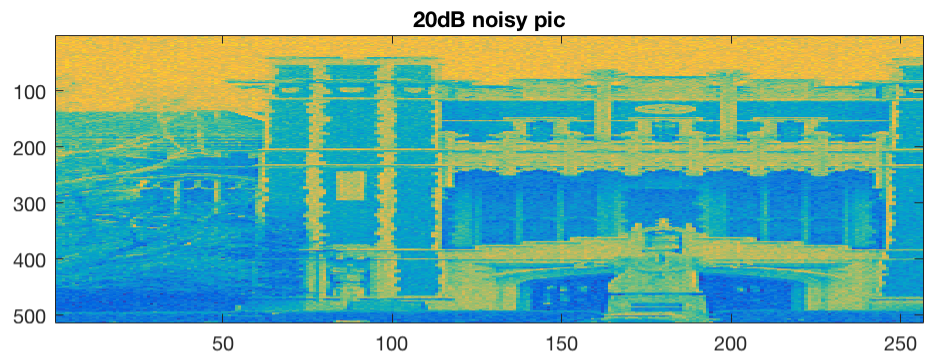
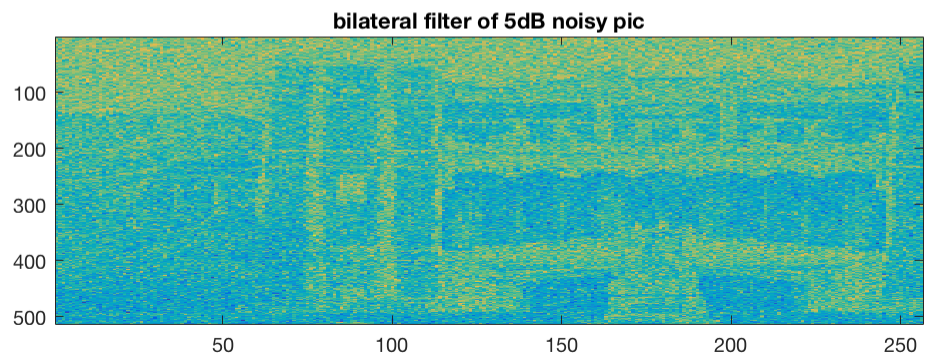
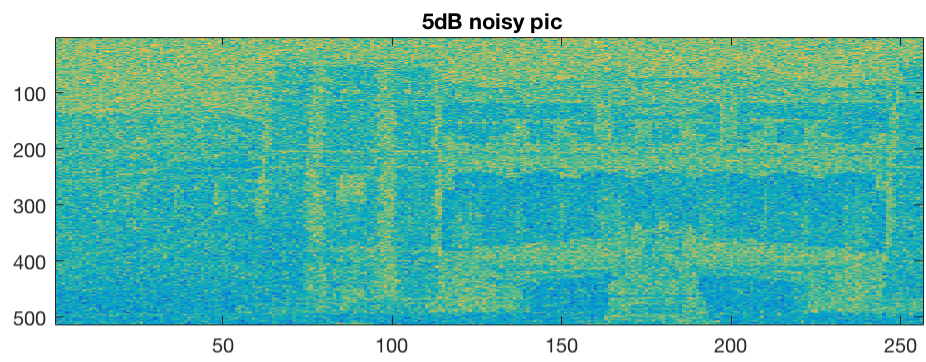


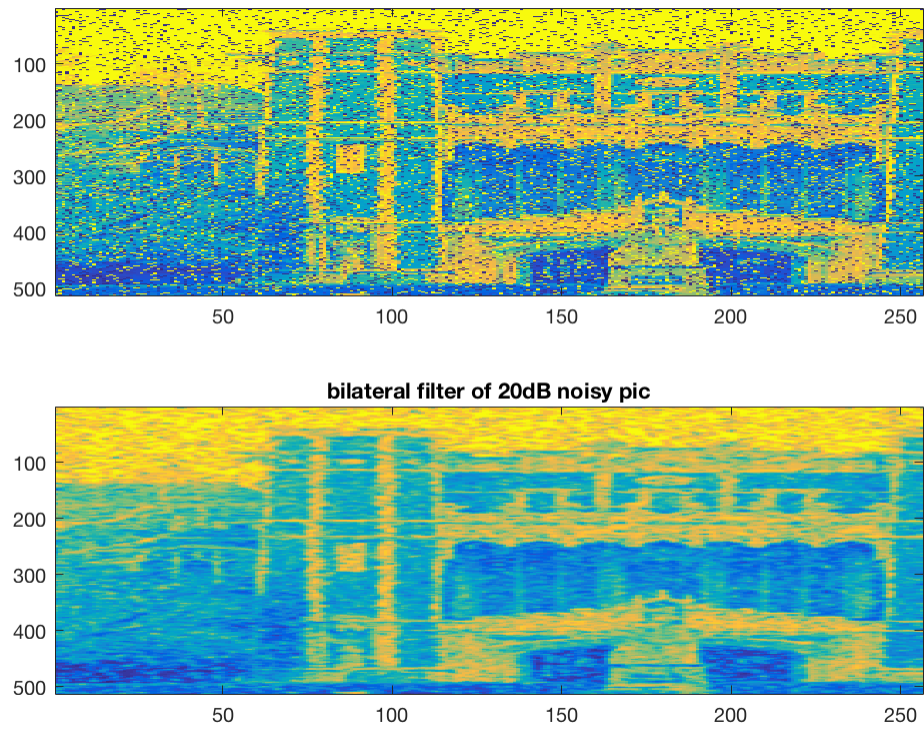












Published with MATLAB® R2016b