EEE4420 : Digital Signal Processing

Project 2

Prof. Yong-June Shin

[Problem 1] FIR Filtering for Speech Processing using Notch Filters

This MATLAB project focuses on designing a notch filter to remove colored noise from a speech signal. The project involves two examples, and the steps for each example are given below.

(a) Colored Noise

Consider a communication channel shown in Figure 1. A vocal signal $x[n]$ is transmitted through this channel, but it gets distorted by "colored" noise $n[n]$, which is characterized by its frequency components. The goal is to design a filter $H(z)$ that removes the colored noise from the signal so that we can obtain a clean signal $y[n]$ that is close to the original signal $x[n]$.

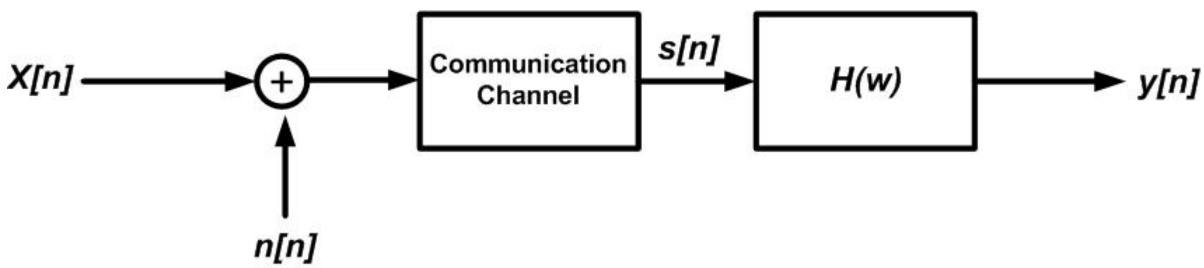


Figure 1: Simplified block diagram of the communication channel with noise.

Vocal Signal with Noise (Example 1)

Download a sample vocal signal from Learnus "`10000.mat`". The signal is a voice signal of your TA saying “만점 받으세요!(I wish you get a perfect score!)” for record testing, which is provided in Figure 2.

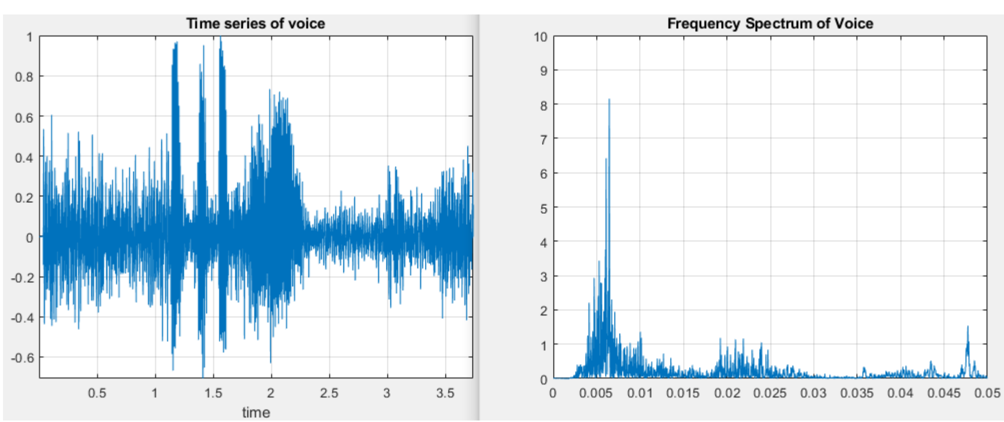


Figure 2: Vocal signal in time domain (on left) and its frequency spectrum (on right)

For the loading and plotting of the signal in the time domain, please refer to the following sample code from MATLAB:

```
load 10000.mat; Fs = 11025;%Sampling Rate of Signal

voice=voice-mean(voice);% Remove DC offset

maxv=max(voice); time=1:length(voice);

figure;plot(time/Fs,voice); grid on;

axis tight; xlabel('time');

title('Time series of voice');
```

In the frequency domain, please refer to the following sample code:

```
voice_len = length(voice);

f=1:voice_len/2;%Plotpositive frequency only

fn=f/voice_len;%Normalization of the frequency

figure;

vdft =fft(voice,voice_len); % Spectrum Computation

vspec =vdft.*conj(vdft)/voice_len; %Power Spectrum

%semilogy(fn,abs(vspec(1:(length(voice)/2)))) % You may plot in log scale

plot(fn,abs(vspec(1:(length(voice)/2))))

title('Frequency Spectrum of Voice');grid on;
```

Unfortunately, a strong sinusoidal signal is introduced to the system and the signal is distorted as shown in Figure 3. It is a notable fact that a strong frequency component is observed at 0.1 in normalized frequency (0.2π in normalized angular frequency), which was not observed in Figure 2. One can listen to the distorted signal by loading the file "**`voice_noise_1.mat`**" from the course website (**`Be careful with the volume!!!`**). For the playing of the vocal signal on your PC, please use the following command in MATLAB. Please use the same sampling rate, i.e., 55.125 kHz.

```
audioplayer(voice,Fs);
```

Notch Filter Design

Now we will design a digital filter to filter out the noise and reconstruct the original signal. If the noise is characterized by a frequency, one can block that frequency component by designing a "nulling filter" (notch Filter) as discussed in this section. (In German, "null" means "zero".) In Figure 4, a system block diagram of a notch filter is provided as a cascaded system.

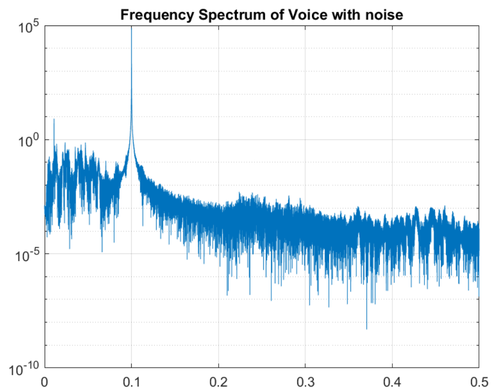


Figure 3: Frequency spectrum of distorted signal (File: **voice_noise_1.mat**)

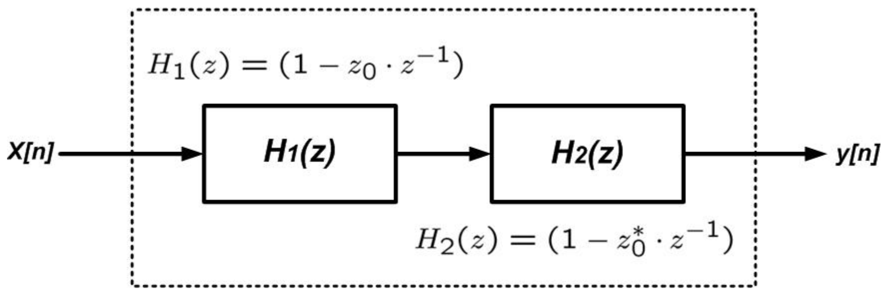


Figure 4: A block diagram of a notch filter.

Consider the transfer function of the overall cascaded system with the z-transform provided below:

$$\begin{aligned} H(z) &= H_1(z) \cdot H_2(z) \\ &= (1 - z_0 \cdot z^{-1}) \cdot (1 - z_0^* \cdot z^{-1}) \\ &= 1 - (z_0 + z_0^*)z^{-1} + z^{-2} \\ & \quad (\text{let} \quad z_0 = e^{j\omega_0}) \\ &= 1 - (e^{j\omega_0} + e^{-j\omega_0})z^{-1} + z^{-2} \\ &= 1 - (2\cos(\omega_0))z^{-1} + z^{-2} \end{aligned}$$

Therefore, one can determine the FIR filter coefficients of the notch filter if the frequency of the noise is known. In this project, the frequency of the noise can be obtained by the power spectrum provided in Figure 3. Please refer to the following "incomplete" parts of MATLAB code and filter the noise by completing the code. Please provide the power spectrum of your output signal. Make sure to confirm the filter output by listening to the signal.

```
FIR_Filter1=[????? ????? ????];
Output_signal=????(signal, FIR_Filter1);
audioplayer(voice,Fs);
```

After the filtering, please provide your filter coefficients and its frequency response plots (magnitude and phase). Also, provide the frequency spectrum of the filtered signal with your MATLAB code.

(b) Vocal Signal with Two Frequency Components (Example 2)

In Figure 5, the frequency spectrum of an unknown signal is provided. One can identify the two major frequency components of the noise. Please load the file "**voice_noise_2.mat**" and listen to the unknown signal. Show your MATLAB code and the frequency spectrum of the filtered signal. Please highlight the filter coefficients and explain why you have selected those filter coefficients. After successful filtering, please write down the message from the unknown signal.

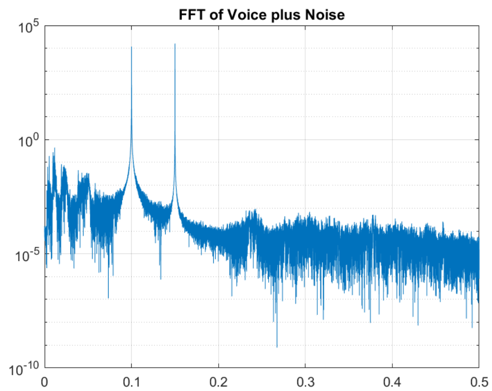


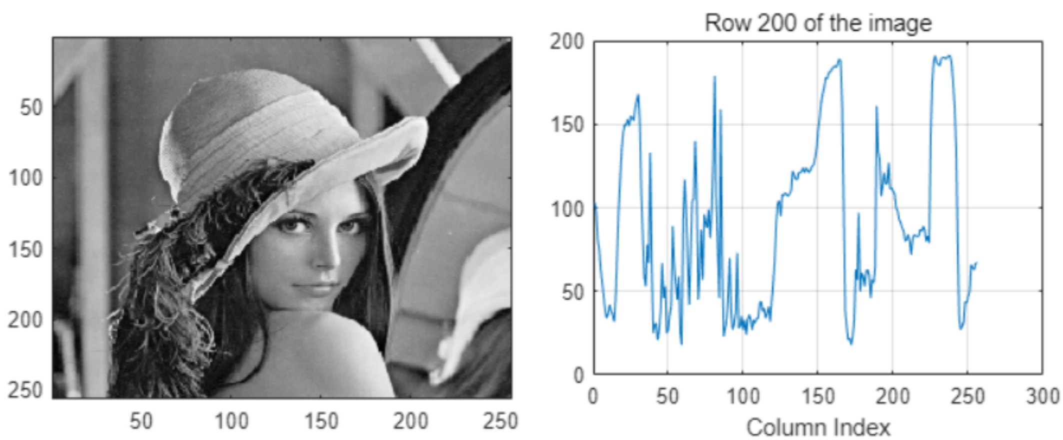
Figure 5: Frequency spectrum of distorted signal (File: **voice_noise_2.mat**)

After the filtering, please provide your filter coefficients and the filtered signal's frequency response plots (magnitude and phase). Also, provide the time domain waveform and the frequency spectrum of the filtered signal with your MATLAB code.

[Problem 2] FIR Filtering for Image Processing

This MATLAB project focuses on designing an FIR filter to remove noise from an image. The project involves five examples, and the steps for each example are given below.
In this MATLAB project, we consider an application of FIR filter to image. you may simply regard an image as a multi-dimensional signal. Please download an "`LENNA.MAT`" from the course website. For the display of image, **please use the following command in MATLAB.** If you can successfully download and execute the command, you may see the image provided in Figure 6a.

```
Data = load("LENNA.MAT");
img = cell2mat(struct2cell(Data));
imagesc(img)
colormap gray
```



(a) (b)

Figure 6: Original image of in (a) and one horizontal scan of the image at row 200 in (b)

In Figure 6a, you can find the sample image (256 by 256). The numbers in axes represent the row and column index, respectively. The horizontal white line in Figure 6a is a cross section of the image at row 200, whose signal is provided in Figure 6b. As you see in Figure 6a and Figure 6b, the image in Figure 6a is a collection of the column vector signals like Figure 6b, in a muti-dimension.
For the display of image and plotting the horizontal scan in Figure 6, please refer following MATALB code:

```
Data = load("LENNA.MAT");
img = cell2mat(struct2cell(Data));
imagesc(img)
colormap gray

Row200oftheImg = ???;
plot(Row200oftheImg)
grid on
xlabel('Column Index')
title('Row 200 of the image')
```

(a) Lowpass and Highpass Filtering

Consider a block diagram representation of a linear and time invariant systems provided in Figure 7. The impulse response of each filter is provided in Figure 7.

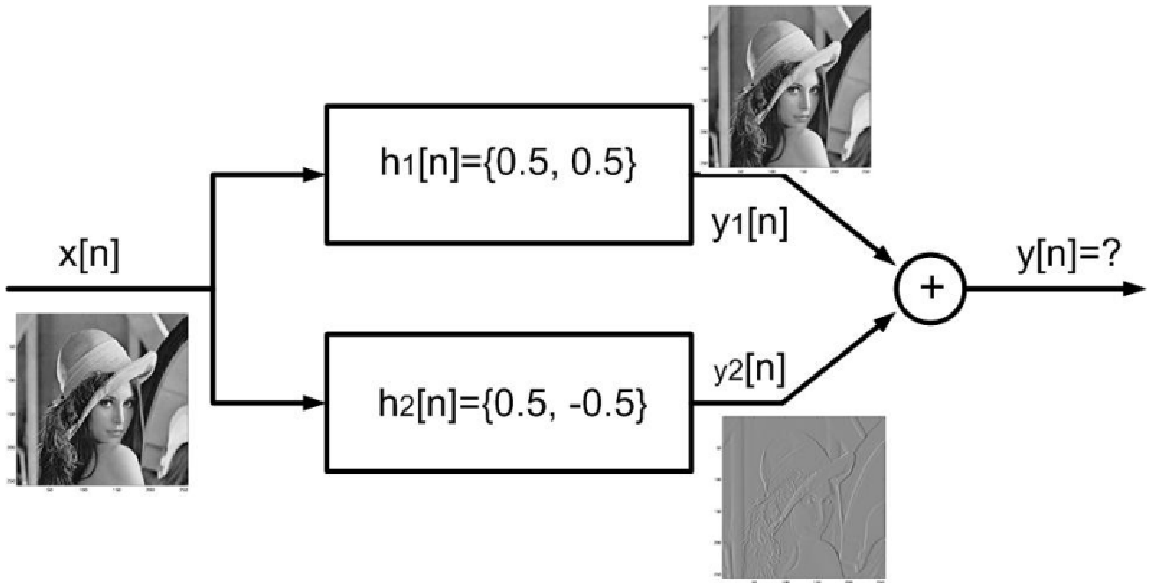


Figure 7: A linear and time invariant system composed of two FIR filters

Assume that the input to the system will be the image file provided in Figure 6.

Then, what would be the output image of the system provided in Figure 7?

Please provide your analytic approach and verify your answer with MATLAB.

For the verification of MATLAB, please refer following "incomplete" MATLAB code.

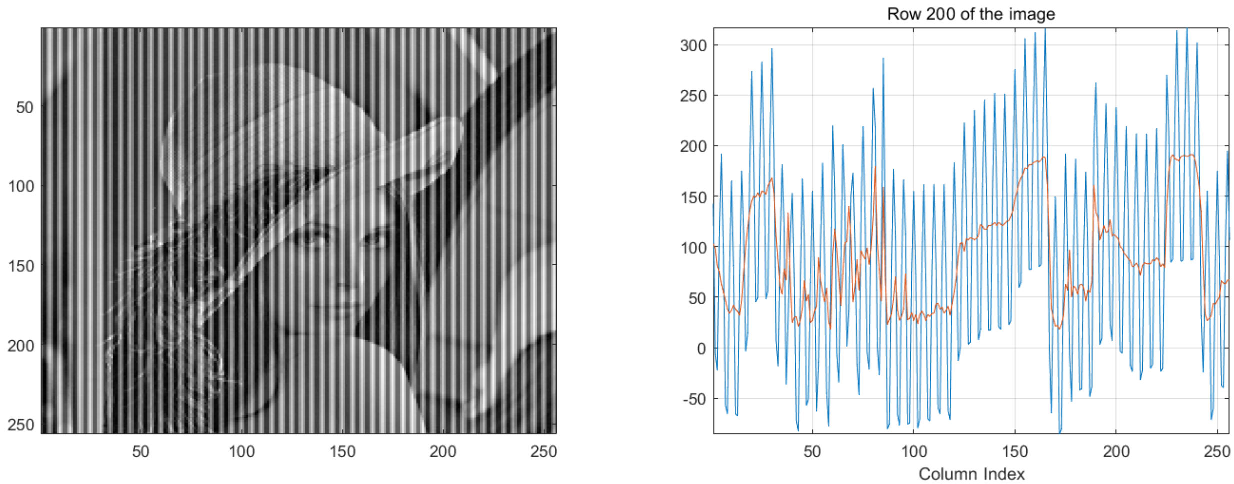
```
h1 = [???,???];
h2 = [???,???];

y1 = conv2(???, ???);
y2 = conv2(???, ???);
y3 = ??????;

imagesc(y1); colormap gray; title('after h1 filter')
imagesc(y2); colormap gray; title('after h2 filter')
imagesc(y3); colormap gray; title('after h1+h2 filter')
```

(b) L-point Running Average Filter in Frequency Domain

The distorted image is provided in Figure 8a. Now we will resolve the distortion of the image by use of FIR filtering.



(a) (b)

Figure 8: Distorted image of in (a) and one horizontal scan of the image at row 200 in (b)

The distorted image is provided in Figure 8a and one horizontal scan of the image at row 200 is provided in Figure 8b. You may download the distorted image file, "**`lenna_noise1.mat`**" in course website. For your information, the horizontal scan of the original image (in blue) and that of the distorted image (in red) are plotted together in Figure 8b. In Figure 8b, it is clear that a strong noise has been introduced which has a specific frequency of oscillation, which results in a set of vertical stripes in Figure 8a. Hence, let us consider the frequency characteristics of the original image and distorted image.

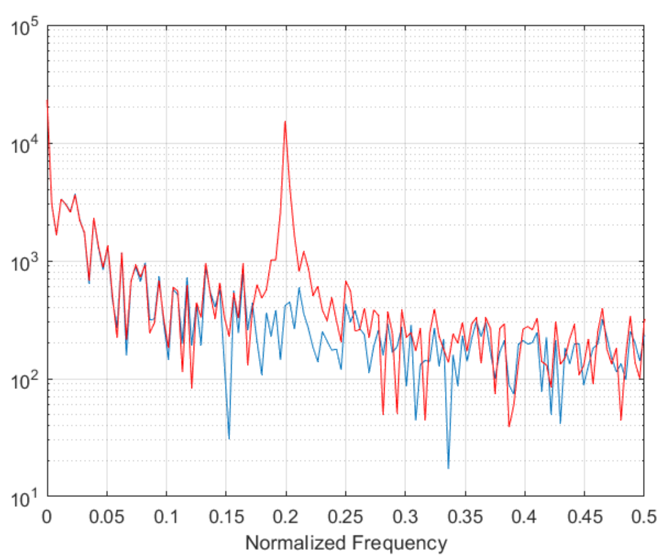


Figure 9: Frequency spectrum of the original (blue) and distorted (red) image of Lenna

In Figure 9, a frequency spectrum of the original image (blue) and the distorted image (red) are provided together along the normalized frequency (0   0.5) in logarithmic scale. Be careful on the scales of the normalized frequency: 0.5 of normalized frequency corresponds to the “pi” of the normalized angular frequency. Also note that the image is simply collection of the 256 signals along the row: the spectrum of the signal is obtained by the averaging of the 256 individual frequency spectra. There is no significant difference between the original and distorted images in lower frequency band, i.e., below 0.1 normalized frequency band.

However, in Figure 9, there is a significantly dominating frequency component at 0.2 normalized frequency in the distorted image, which is the suspect of the strong noise observed in Figure 8a and 8b. **For the calculation and plot of the spectrum, please refer following MATLAB code:**

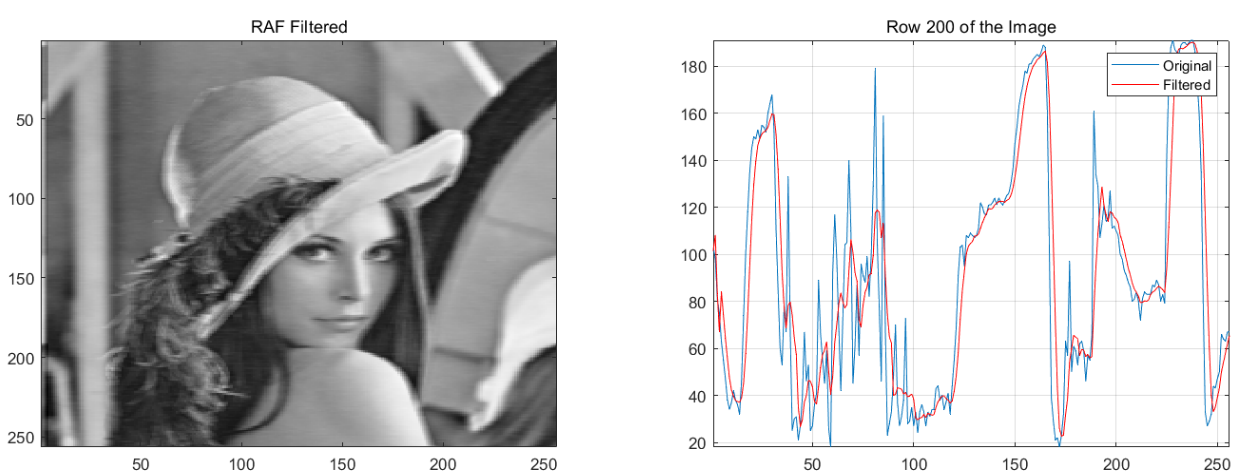
```
N = 256;
Spectrum_temp0=fft(???, N);
Spectrum_temp1=fft(???, N);
F = (0:N-1)/N; % Frequency scaling
semilogy(F, ???); % logarithmic plot of the spectrum from original image
hold on; grid on;
semilogy(F, ???,'r'); % logarithmic plot of the spectrum from noisy image
xlim([0 .5]);
xlabel('Normalized Frequency');
```

Now, based on the frequency spectrum of the distorted image, please design a FIR filter that removes the “strong” noise at 0.2 normalized frequency. In this project, the type of FIR filter is limited to the L-point Running-Average filters. If you refer to the frequency characteristics of the L-point Running-Average filters, you may determine the proper length of the FIR filter. **Please provide following figures with your code and comments:**

1 Frequency response of your FIR filter. Also explain your design scheme of the filter

2 Filtered image with you FIR filter

3 Frequency spectrum of distorted image and processed image (like Figure 9)



(a)

(b)

Figure 10: Recovered image of the distorted image by FIR filtering

If you properly designed FIR filter and processed the distorted image, you can obtain the recovered image as shown in Figure 10a. The signal in Figure 10b is the horizontal scan of filtered image at row 200. As you see in Figure 10b, the FIR filtering successfully removes noise and the filtered signal is very close to that of the original signals. However, owe to the averaging operations with FIR filter, the filtered image is not as good as the resolution of the original image in Figure 6a.

For an additional help and hint to solve this problem, please read L-point Running-Average Filtering in your materials.

(c) Application of Notch Filter

In the class, we have discussed applications of cascaded FIR filters. One can implement a notch filter in order to block a specific frequency component.

Consider the transfer function of both notch filter without poles (Figure 11a) and notch filter with poles (Figure 11b):

$$H_z(\omega) = b_0\left(1 - e^{-j\omega_0}e^{-j\omega}\right) \cdot \left(1 - e^{j\omega_0}e^{-j\omega}\right)$$

(a)

$$H_{zp}(\omega) = \frac{b_0\left(1 - e^{-j\omega_0}e^{-j\omega}\right) \cdot \left(1 - e^{j\omega_0}e^{-j\omega}\right)}{\left(1 - re^{-j\omega_0}e^{-j\omega}\right) \cdot \left(1 - re^{j\omega_0}e^{-j\omega}\right)}$$

(b)

Figure 11: Transfer functions of both notch filter without poles (a) and notch filter with poles(b)

Consider the effect of poles and determine the appropriate coefficients for the given functions. (Figure 11)
Please try to remove the noise in the image by use of both given notch filters.
In addition, please compare the results of both given notch filters with that of Running-Average filters. (like Figure 10a and Figure 10b)

(d) Unknown Image Identification

Based on the discussion and experiment of FIR filter, we will challenge two problems of image processing. The first mission is to recover an unknown, but distorted image shown in Figure 12. Please download the file from the course website. The name of the image file is **"unknown.mat."** Using the FIR filter design (L-point Running-Average filter), please remove the noise and recover the original image. **Please provide following figures with your code and comments:**

1 Frequency spectrum of distorted image

2 Frequency response of your FIR filter with explanation

3 Frequency spectrum of distorted image and processed image (like Figure 9)

4 Filtered image (What can you see in the filtered image?)

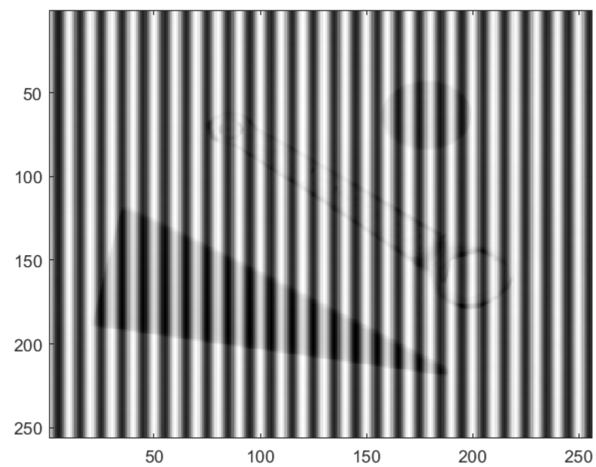


Figure 12: A unknown distorted image to be processed by FIR filtering (**unknown.mat**)

(e) Unknown Noisy Image

The second mission is to enhance the image of the Lenna, which is deteriorated by other types of noise as shown in Figure 13. You may download the image file, **"lenna_sand.mat"** at the course website. By use of FIR filters (L-point average filter) enhance the quality of the image. **Please provide following figures with your code and comments:**

1 Frequency spectrum of original and distorted image (like Figure 9)

2 Frequency response of your FIR filter with explanation

3 Frequency spectrum of distorted image and processed image (like Figure 9)

4 Filtered image

Figure 13: Distorted image of Lenna with different types of noise (**lenna_sand.mat**)

With your FIR filters, can you "significantly" enhance the quality of distorted image in Figure 13? **If you can, please explain how your filter works for the deteriorated image. If not, please explain why FIR filter does not work for the deteriorated image in Figure 13.**