Korea Advanced Institute of Science and Technology

School of Electrical Engineering

EE535 Digital Image Processing Spring 2018

Student’s Name: Dinh Vu

Student’s ID: 20184187

**Homework 5**

1. **Image Restoration**



* 1. ***Pseudo-inverse***
     1. *Theoretical background*

The pseudo-inverse filter is a stabilized version of the inverse filter. For a linear shift invariant system with frequency response , the pseudo-inverse filter is defined as:

Here, is also called the generalized inverse of in analogy with the definition of the generalized inverse of matrices. In practice, is set to zero whenever is less than a suitably chosen positive quantity.

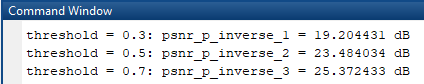
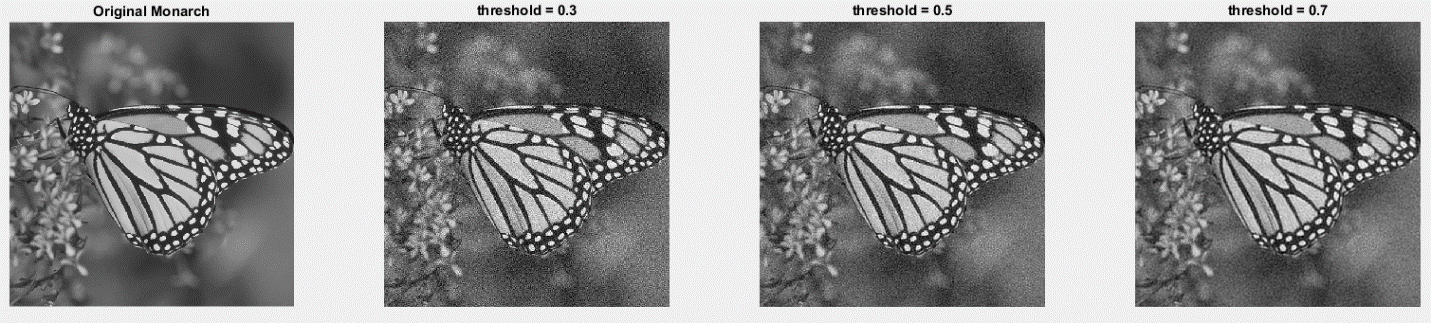
* + 1. *Programming strategy*

First, the original image is added Gaussian noise by function GaussianNoise with standard deviation equals to 0.086 in order to get SNR = 14 dB. After that, the image will be restored by function PseudoInvFilter using Hamming window. The parameter threshold of function PseudoInvFilter presents the chosen positive quantity in the theory.

When determining the threshold, the threshold must be set high to reduce noise significantly. However, in order to preserve the high frequency components of the image, the threshold must be set low. Therefore, that is a tradeoff between noise suppression and preservation of high frequency components.

* + 1. *Results & Analysis of results*

By the experimental results, in Figure 1.1, it can be seen that PSNR is low at 19.20 dB with threshold = 0.3 and the Gaussian noise is amplified much, but the high frequency components of the image still exist because the image look less blur. At threshold = 0.5 and 0.7, PSNR is higher but the high frequency regions is lost and blurred. However, we can see that the Gaussian noise is smaller than the previous case.



**Figure 1.1.** The original image and 3 output images of Pseudo-inverse filter and corresponding PSNR

* 1. ***Wiener Filter***
     1. *Theoretical background*

Wiener filtering is a method of restoring images in the presence of blur as well as noise. Let and be arbitrary, zero-mean, random sequences. It is desired to obtain an estimate, to minimize the mean square error:

The best linear estimate of the form:

The Fourier transform of is:

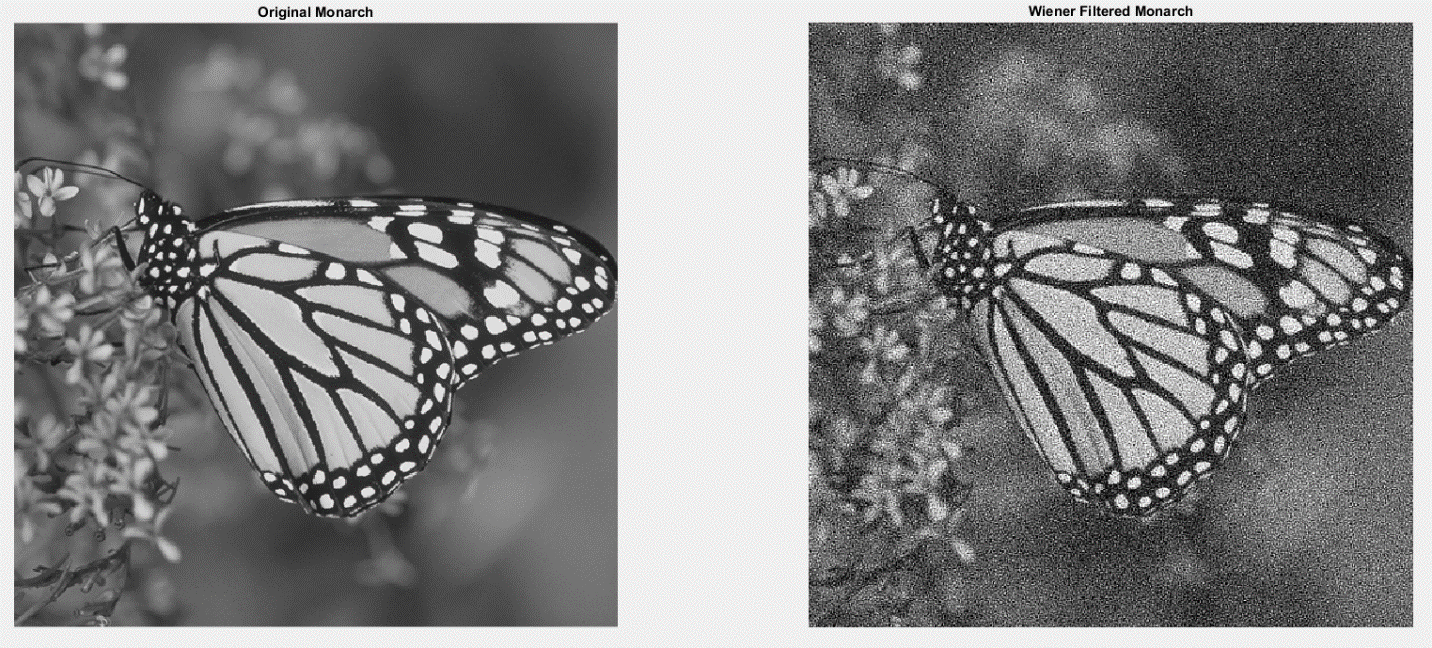
Where:

* is the power spectral density of
* is the power spectral density of the stationary noise sequence
  + 1. *Programming strategy*

After adding Gaussian noise by function GaussianNoise with standard deviation equals to 0.086, the result image is input of Wiener filter. The image will be restored by function WienerFilter using the same Hamming window in the function PseudoInvFilter.

* + 1. *Results & Analysis of results*

As a result, the Wiener filter measured at PSNR = 14 dB. The image was darker and the Gaussian noise tended to be even more pronounced.



**Figure 1.2.** The original and the output image of Wiener Filter

* 1. ***Constrained Least Square Restoration***
     1. *Theoretical background*

The constrained least square restoration filter output , which is an estimate of , minimizes a quantity: subject to the constraint: where .

Using the Parseval relation this implies minimization of subject to .

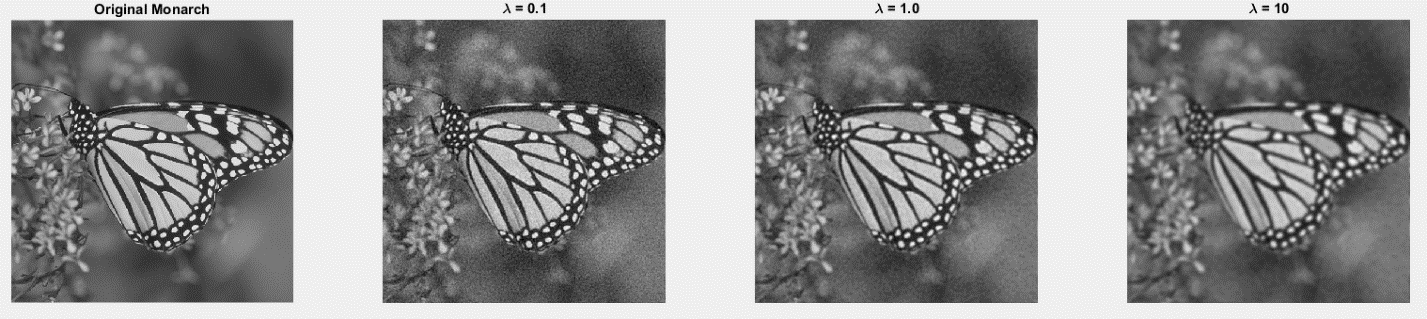
Applying the Lagrange multiplier method gives: . Where:

* + 1. *Programming strategy*

After adding Gaussian noise by function GaussianNoise with standard deviation equals to 0.086, the result image is input of Wiener filter. The image will be restored by function LeastSquareFilter using the same Hamming window in the function PseudoInvFilter. The Laplace operator is presented in matrix form.

* + 1. *Results & Analysis of results*

Figure 1.3 shows that the larger the λ, the more the images tend to blur. The reason for this tendency is in the equation (1.1). is the Laplacian operator and has a high payout at high frequency. Therefore, when λ becomes larger, the denominator becomes larger at the high frequency, so that it becomes lower as a whole and has the characteristic of the LPF. Comparing the below three λ values, we can see that λ = 1 is an appropriate value both from the subjective viewpoint and PSNR viewpoint. The above experiment was done to see the tendency of the change due to λ.

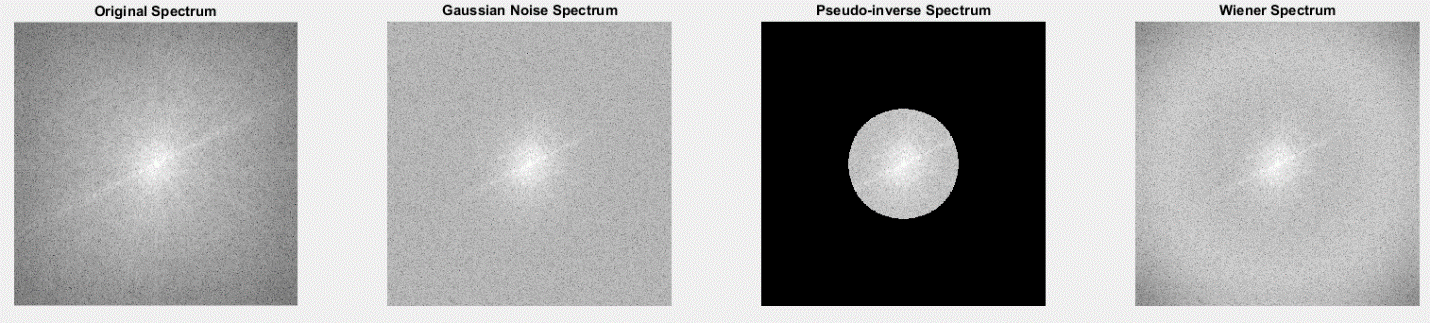


D:\KAIST\Courses\dip\hw\hw5\pic\prob_1_3_psnr.PNG

**Figure 1.3.** The original image and 3 output images of Constrained Least Square Filter and corresponding PSNR

* 1. ***Draw the power spectra of original image and noise. Describe the frequency*** ***characteristics of the pseudo-inverse and Wiener filter.***

FFT in 2-dimension is used to examine the frequency characteristics of images. The power spectrum of the original image mainly contains a lot of the low frequency components and few components in the high frequency. When the image passes through the Hamming filter (LPF), the number of high frequency components is reduced. However, when Gaussian random noise is added to the image, the frequency components are displayed in all regions including high and low as shown in Gaussian Noise Spectrum figure. In this figure, it can be seen that the low frequency components appears more clearly, but there is no obvious difference as in Original Spectrum.



**Figure 1.4.** The power spectra of the original and noise images

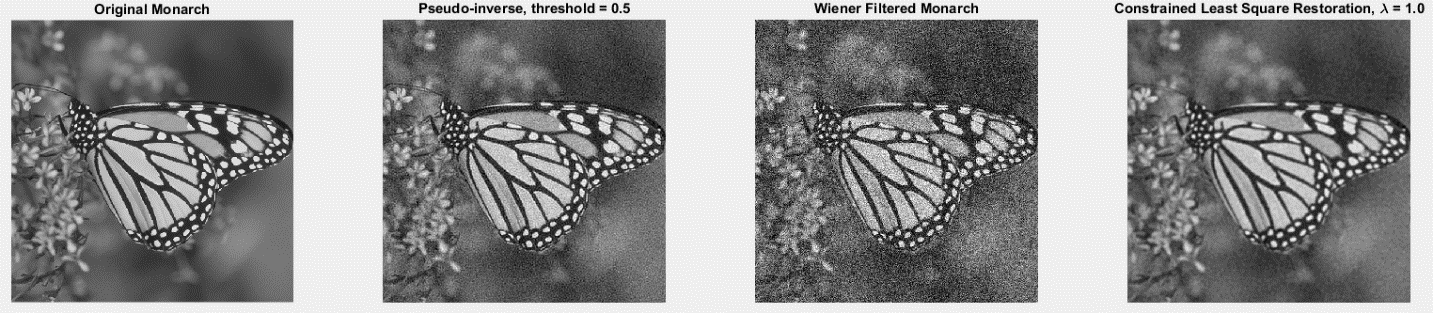
The pseudo inverse filter makes the values above the threshold in the high frequency region where noise becomes larger than the signal components in order to compensate the unstable characteristic of the general inverse filter. This characteristic has the disadvantage that noise can be reduced, but at the same time, the high frequency components of the original image is also reduced. From the above results, it can be seen that the high frequency above a certain value is processed as zero.

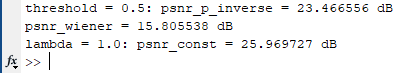
The Wiener filter is a filter that considers both the power spectrum of the noise and the original image. Hence, it looks like an inverse filter in the low frequency regions where the noise is less affected. In the high frequency regions, the noise size is similar to the size of the original components, the shape is smoothly reduced as in the smoothing filter. In Wiener Spectrum figure, the high frequency components decrease.

* 1. ***Compare and analyze the image characteristics results from 1.1, 1.2 and 1.3***

As mentioned earlier, in the case of the Pseudo-inverse filter, high frequency components higher than a certain threshold are assigned to zero. Experimental results show that PSNR tends to be increased when the threshold increase, but the high frequency components of the original image disappear as well, causing a mount of blur. When the threshold is low, the noise is less removed and amplified, but the high frequency components of the original image tend to remain. It is very difficult to balance between two criterions because most of the images are not perfect (PSNR is 19 - 25 dB).

On the other hand, the Wiener filter has a bimodal soft filtering in order to minimize the mean square error by considering the power spectral density of the image and noise. However, it is relatively inefficient to remove noise than pseudo inverse. It is also not perfect because the original image is restored.





**Figure 1.5.** The original and the output images of each filter

1. **Image Analysis**
   1. ***Transform Features***
      1. *Theoretical background*

Image transforms provide the frequency domain information in the data. Transform features are extracted by zonal-filtering the image in the selected transform space (Figure 2.1). The zonal filter, also called the feature mask, is simply a slit or an aperture. Generally, the high-frequency features can be used for edge and boundary detection, and angular slits can be used for detection of orientation.



**Figure 2.1.** Transform feature extraction

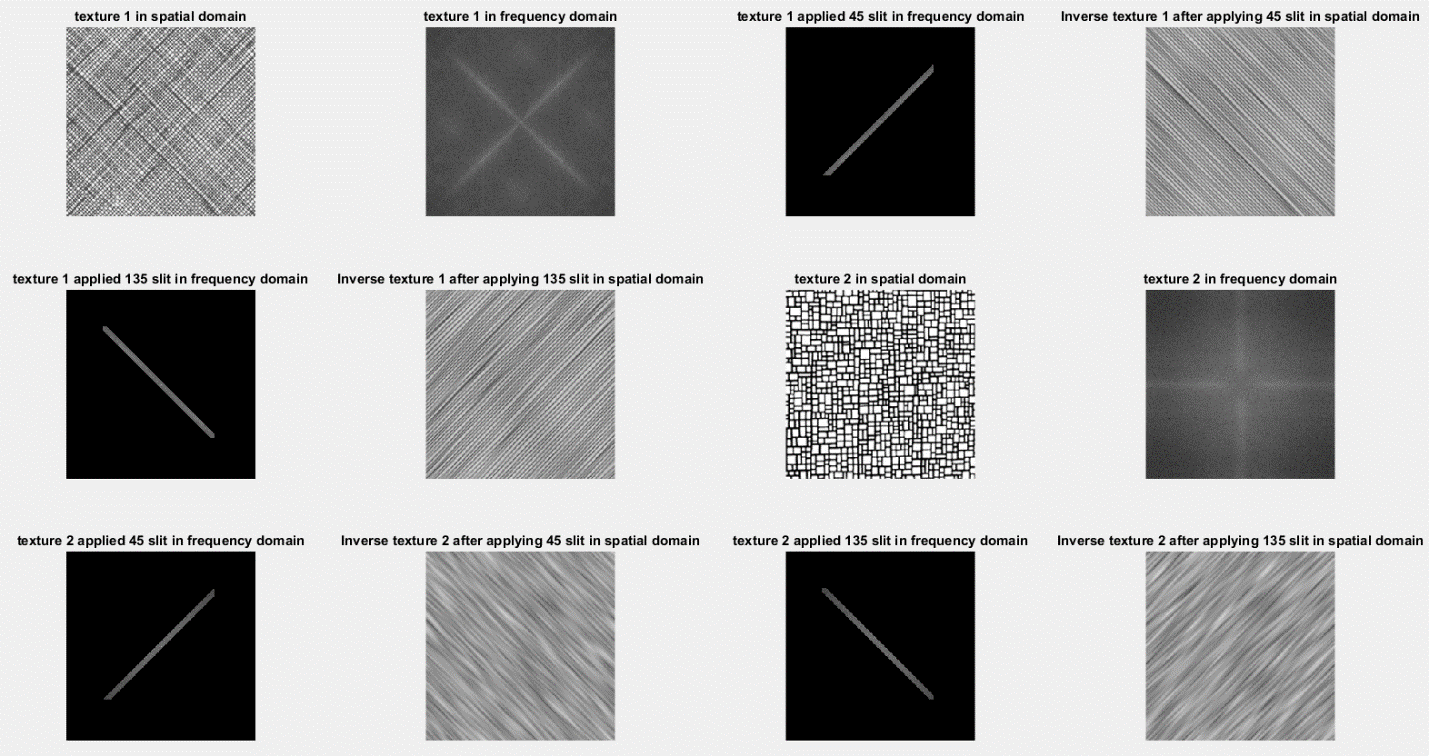
* + 1. *Programming strategy*

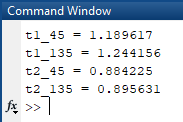
In problem 2.1, angular slits of 45o and 135o are implemented as slits with thickness of 13 pixels. After passing the slit, the energy ratio, which is the ratio of the slit energy to the Fourier transformed image energy in the frequency domain, is obtained by sum of square of the frequency elements.

* + 1. *Results & Analysis results*

In the spatial domain, texture 1 consists straight lines in 45o and 135o direction, so energy concentrates around 45o and 135o in the frequency domain. Therefore, when transform feature texture 1 by 45o - 135o angular slit, smooth 45o – 135o line of energy is obtained in the frequency domain while in the spatial domain, the 45o and 135o looks clear.

On the contrary, texture 2 mostly includes vertical and horizontal lines. Hence, its energy assembles at 0o and 90o in frequency domain. Consequently, applying 45o and 135o slit for texture 2 gives us discontinuous 45o and 135o line in the frequency domain. In the spatial domain, the inverse image of texture 2 is more blurred than texture 1. As the results, in 45o and 135o direction, the energy ratio of texture 1 is higher than texture 2. In the same texture, the energy ratio in each direction is nearly equal.





**Figure 2.2.** Transform features of texture 1, 2 and energy ratio

* 1. ***Threshold-based Segmentation***
     1. *Theoretical background*

Image segmentation refers to the decomposition of a scene into its components. There are many segmentation techniques such as threshold-based segmentation presented in this section. In threshold-based segmentation, there are also several methods to determine the threshold value like Ostu’s method. Otsu’s method is used to automatically perform clustering-based image [thresholding](https://en.wikipedia.org/wiki/Thresholding_(image_processing)" \o "Thresholding (image processing)) or the reduction of a gray level image to a binary image. The algorithm assumes that the image contains two classes of pixels following bimodal histogram (foreground pixels and background pixels), it then calculates the optimum threshold separating the two classes so that their combined spread (intra-class [variance](https://en.wikipedia.org/wiki/Variance)) is minimal, or equivalently (because the sum of pairwise squared distances is constant), so that their inter-class variance is maximal.

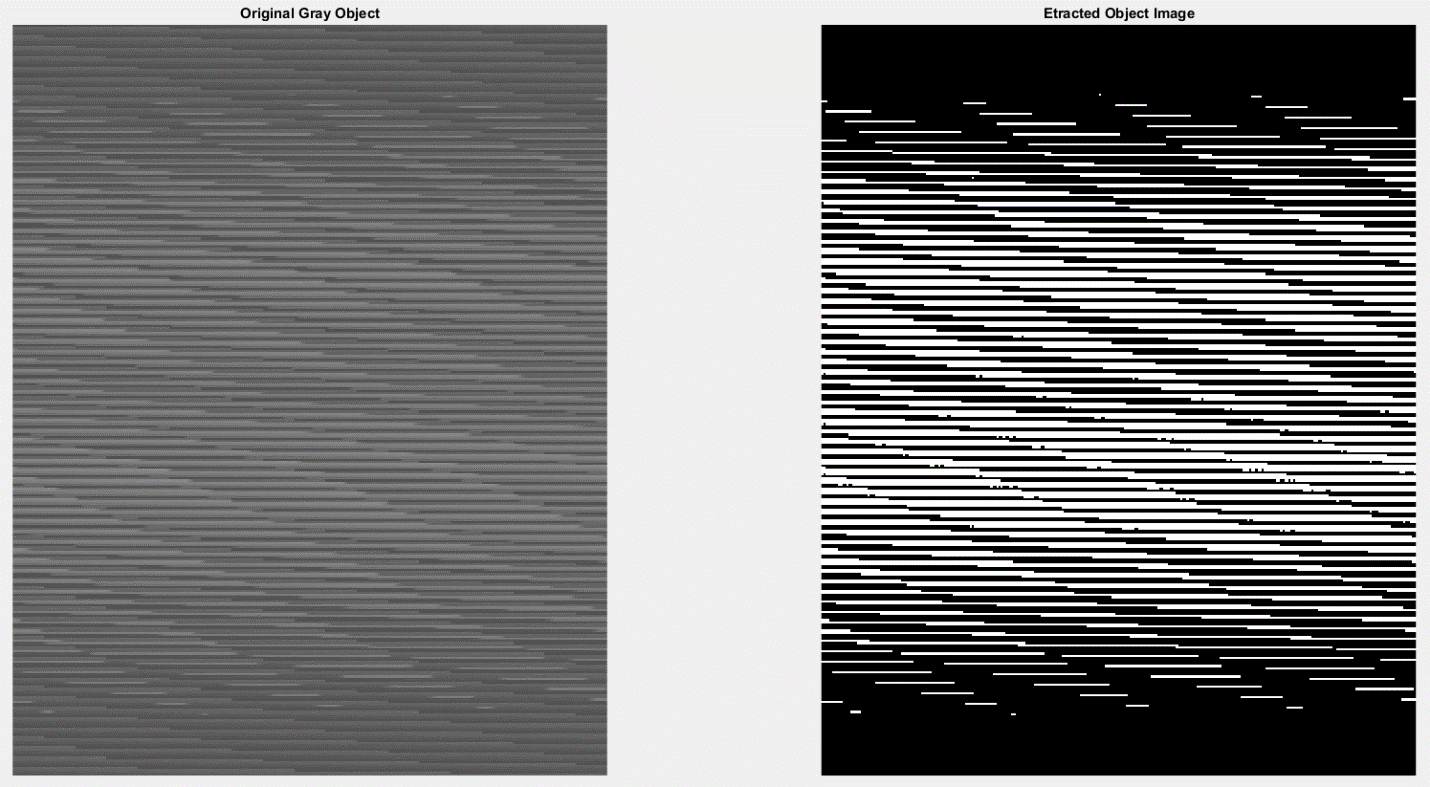
* + 1. *Programming strategy*

The algorithm of Ostu’s method:

* Compute histogram and probabilities of each intensity level
* Setup initial and
* Step through all possible threshold
* Update and
* Calculate
* Desired threshold corresponds to the maximum

If the intensity of pixels in the original image greater than above threshold, the pixels is assigned to white luminance. Otherwise, the other pixels is set to black.

* + 1. *Results & Analysis results*



D:\KAIST\Courses\dip\hw\hw5\pic\prob_2_2_threshold.PNG

**Figure 2.3.** The original and extracted object image and threshold

The original image has object look like wave shape making by diagonal and straight lines. So, the extracted object image gives us white straight line make wave shape brightest and the background decreased to zero.

* 1. ***General Hough Transform***
     1. *Theoretical background*

The Hough transform is a [feature extraction](https://en.wikipedia.org/wiki/Feature_extraction) technique used in [image analysis](https://en.wikipedia.org/wiki/Image_analysis), [computer vision](https://en.wikipedia.org/wiki/Computer_vision), and [digital image processing](https://en.wikipedia.org/wiki/Digital_image_processing). The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure. This voting procedure is carried out in a [parameter space](https://en.wikipedia.org/wiki/Parameter_space), from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform.



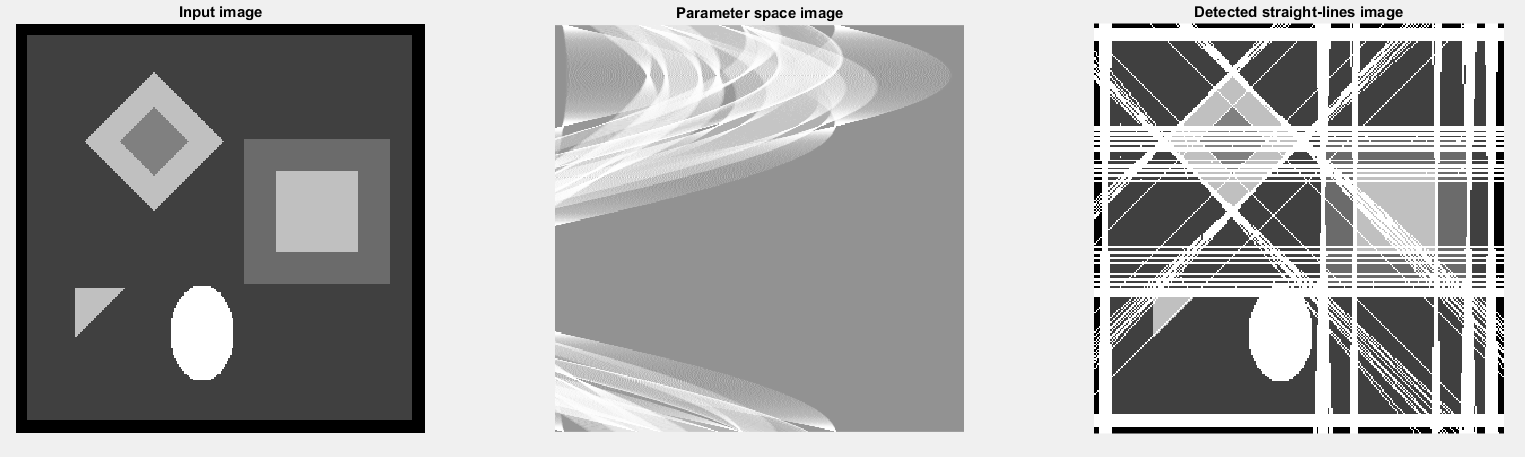
**Figure 2.4.** Hough transform

* + 1. *Programming strategy*

First, using compass operator and Hough transform to the lines with 360o passing through the edge candidate pixels in order to detect the edges of the image, then count the corresponding points in the parameter space. After that, applying the histogram equalization to the parameter space. The linear expressions in the original image are obtained by inverse transforming r and theta, which are the coordinates of the point, only for the points with a count greater than 60. Finally, the pixel value of this straight line is changed to 255 in order to get the detected straight lines image.

* + 1. *Results & Analysis results*

A pre-edge image is made using the compass operator and then performed Hough transform on the edges. The resultant ordered pair (r, theta) is counted as a relatively bright white point in the parameter space. This count number is reversed only for the points that are more than 60, so each pixel on the corresponding straight line is marked as a bright pixel on the original image. The parameter space has been histogram equalized, so the contrast is increased and relatively well-marked.



**Figure 2.5.** The output images of general Hough transform