Computer Engineering Department

Faculty of Engineering, Mahidol University

LAB03: Image Enhancement in the Spatial Domain (part3)

Objectives

Upon completion of this lab, you will be able to:

1. Understand the mechanics of spatial filtering.
2. Describe the different types of smoothing spatial filtering.
3. Describe the different types of sharpening spatial filters.
4. Write an user-defined function in MATLAB to remove noise using smoothing spatial filtering, including median filtering and average filtering.
5. Write the user-defined function in MATLAB for finding the edges of objects within an image, including first-order and second-order derivative filters.
6. Write the user-defined function in MATLAB to sharpen a blurry image by subtracting the Laplacian filtered image from the blurry image.

Exercises

Note that you should create your own function in MATLAB as MATLAB User-defined function. It means that you cannot call MATLAB built-in function, which generates output in the same manner as your own function. You can use the images provided in the folder \Google Drive\EGCO486(60-1)\LABs\LAB03_Part3 for your exercises.

1) Image enhancement in spatial domain using median filter

1.1 Consider the eight image corrupted by salt and pepper noise. Write the user-defined function in MATLAB to filter the noisy version of the eight image with a 3×3 median filter using zero padding, with the following function name: Mymed.m. When this program is used with the image "eight_salt_pepper.tif" result as shown in Figure 1.

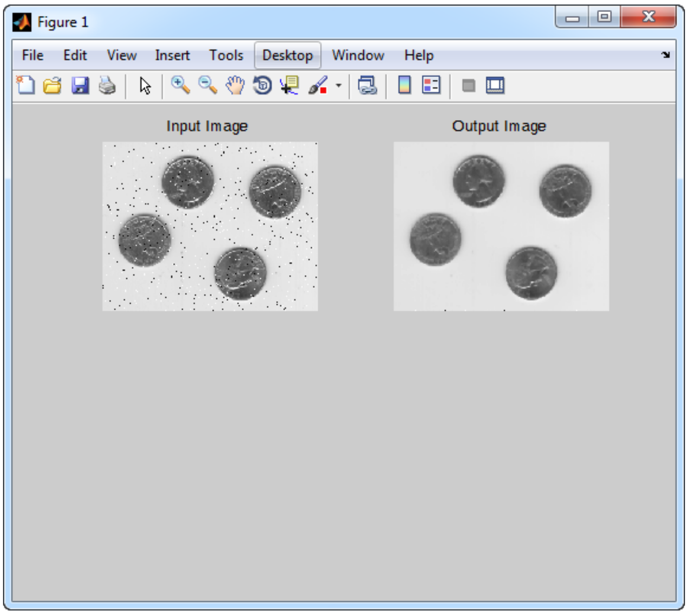

Figure 1: The result image of applying the 3×3 median filter on the noisy image.

2) Image enhancement in spatial domain using average filter

2.1 Consider the eight image corrupted by Gaussian noise. Write the user-defined function in MATLAB to filter the noisy version of the eight image with a 3×3 standard average filter using zero padding, with the following function name: Myaver.m. When this program is used with the image "eight_gaussian.tif" result as shown in Figure 2.

- The standard average filter (3×3)

$$\frac{1}{9}\times\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline\end{array}$$

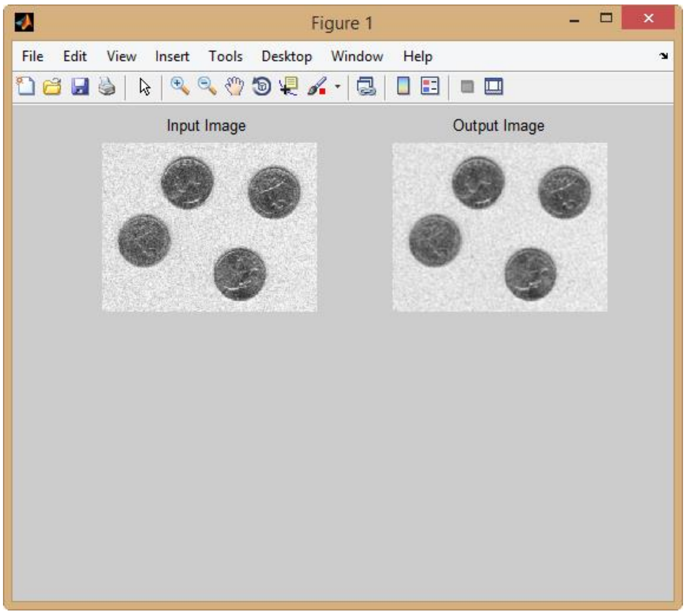

Figure 2: The result image of applying the 3×3 standard average filter on the noisy image.

3) Image enhancement in spatial domain using first-order derivative filter (Gradient method)

3.1 Write a user-defined function in MATLAB for finding the edges of objects within an image with the gradient method using the Prewitt mask filters. Take the following function name: Mygrad.m. Using this program on the image “aaa256.jpg” should give you result as shown in Figure 3.

- 3×3 Prewitt mask filters

-1	-1	-1
0	0	0
1	1	1

H_R

-1	0	1
-1	0	1
-1	0	1

H_C

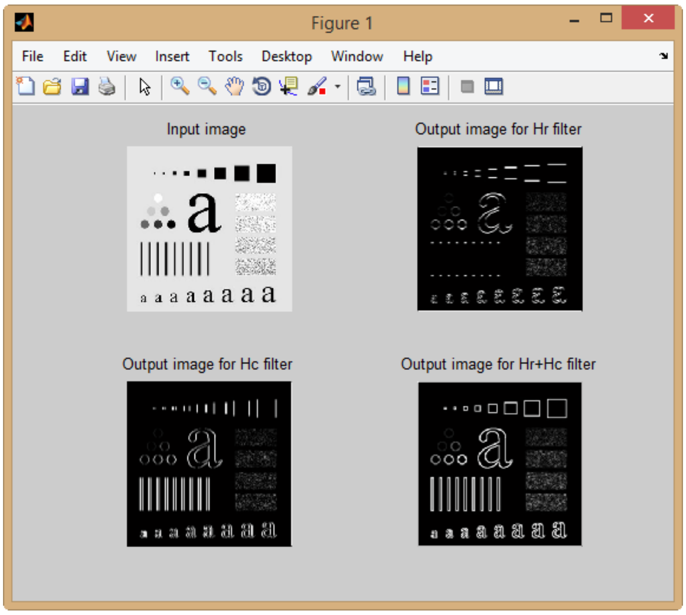
Figure 1
File
Edit
View
Insert
Tools
Desktop
Window
Help
Input image
Output image for Hr filter
Output image for Hc filter
Output image for Hr+Hc filter

Figure 3: The result image of applying two 3×3 Prewitt mask filters on the input image.

4) Image enhancement in spatial domain using second-order derivative filter (Laplacian method)

4.1 Write a user-defined function in MATLAB for finding the edges of objects within an image with the Laplacian method using the Laplacian mask filter. Take the following function name: MyLaplaNeg.m. When this program is used with the image "aaa256.jpg" result as shown in Figure 4.

- 3×3 Laplacian mask filter

0	1	0
1	-4	1
0	1	0

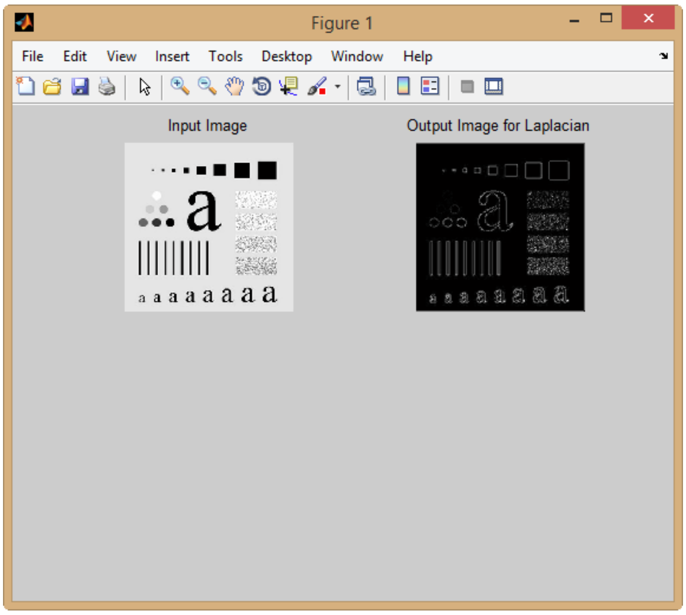

Figure 4: The result image of applying the 3×3 Laplacian mask filter on the input image.

5) Laplacian image enhancement

3.1 Write a user-defined function in MATLAB to sharpen a blurry image. Firstly, you should employ the Laplacian mask filter on the blurry image to obtain Laplacian filtered image. You then restore the gray tones lost simply by subtracting the Laplacian filtered image from the blurry image. Note that we have used the subtraction in the enhancement process, because the center coefficient of the Laplacian mask filter was negative. Take the following function name: EnhanLaplaNeg.m. Using this program on the image "blurry_moon.tif" should give you result as shown in Figure 5.

- 3×3 Laplacian mask filter

1	1	1
1	-8	1
1	1	1

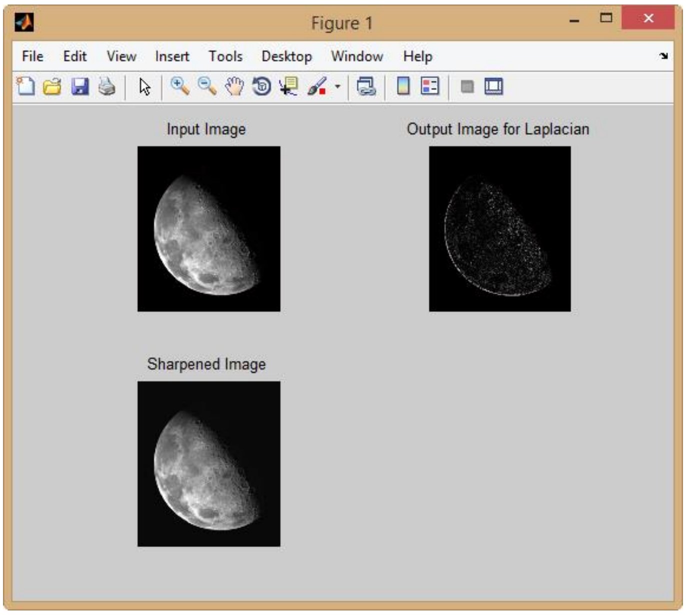
Figure 1
File Edit View Insert Tools Desktop Window Help
Input Image
Output Image for Laplacian
Sharpened Image

Figure 5: The sharper image obtained by subtracting the Laplacian filtered image from the blurry image.

<u>What you need to submit:</u>

Prepare a zip file that contains all matlab files (m-file extension). Email the zip file to the account **send2narit@hotmail.com** with the following subject line: **EGCO486_LABxx_yyy**, which xx is a number of LAB and yyy is the last 3 digits of the student identification number. Your email should reach us before Tuesday 11:59 PM.