

Haruka Kido  
EE456: Digital Image Processing  
Assignment 2

1. Write a MATLAB program that performs a 3x3 linear filtering operation on a given gray level image. Use linear filtering method discussed in the lecture. **Do NOT use any MATLAB built-in filtering commands.** Test your code with the following filter and “cameraman.tif” image. Submit the source code, input and filtered images.

Filter

0	1	0
1	-4	1
0	1	0

- a) Source Code (Linear Filtering: Sobel Edge Example):

```
A = imread('cameraman.tif');

%sets test filter
kernel = [0 1 0;
          1 -4 1;
          0 1 0];

%initializes row and col of image
img_row = size(A, 1);
img_col = size(A, 2);

%initializes row and col of filter
numofrows = size(kernel, 1);
numofcols = size(kernel, 2);

%borders of filter
row = numofrows - 1;
col = numofcols - 1;

filteredA = uint8(zeros(size(A)));

%input for linear filter convolution
input = zeros(img_row + row, img_col + col);
m = floor(numofrows/2);
n = floor(numofcols/2);
rowsize = size(input, 1) - row;
colsize = size(input, 2) - col;

for r = 1:img_row
    for c = 1:img_col
        input(r + m, c + n) = A(r, c);
    end
end
```

```

%Application of 3x3 filter
for r = 1:rowsize
    for c = 1:colsize
        m = r;
        n = c;
        sum = 0; %initializes sum to 0 during shift
        for u = 1:numofrows
            for v = 1:numofcols
                %summation of convolution
                sum = sum + (input(m, n) * kernel(u, v));
                n = n + 1;
            end
            m = m + 1;
            n = c;
        end
        %sobel edge linear filtering output image
        filteredA(r, c) = sum;
    end
end

subplot(1, 2, 1);
imshow(A);
subplot(1, 2, 2);
imshow(filteredA, []);

```

b) Input and Filtered Images:



2. Use your filtering code implemented in Question 1 and perform a sharpening filtering on “lena\_gray.jpg” image. Use sharpening filtering method discussed in the lecture. Give the results for two different high-boosting constant values. **Do NOT use any MATLAB built-in sharpening and filtering commands.** Submit the source code, input and filtered images.

a) Source Code (Sharpening Filter Example):

```
clear;
A = imread('lena_gray.jpg');

%high-boost filtering with constant > 1 as constant = 1.1
SHARP = [-1.1 -1.1 -1.1;
        -1.1 11 -1.1;
        -1.1 -1.1 -1.1];
B = conv2(A, SHARP, 'valid'); %convolution
C = uint8(B);

%high-boost filtering with constant > 1 as constant = 1.3
SHARP = [-1.3 -1.3 -1.3;
        -1.3 11 -1.3;
        -1.3 -1.3 -1.3];
D = conv2(A, SHARP, 'valid'); %convolution
E = uint8(D);

subplot(1, 3, 1);
imshow(A);
subplot(1, 3, 2);
imshow(C);
subplot(1, 3, 3);
imshow(E);
```

b) Input and Filtered Images (results for two different high-boosting constant values):



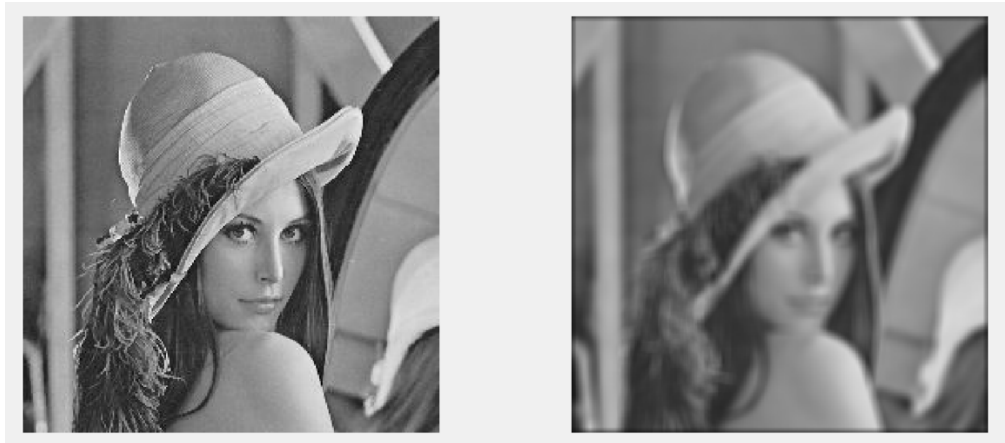
3. Apply the following filters to images using MATLAB built-in filtering commands. You can use any MATLAB built-in command. Submit the source code, input and filtered images.

- a. 7x7 Average filter to “lena\_gray.jpg” image:

- a) Source Code:

```
clear;  
A = imread('lena_gray.jpg');  
  
avgfilterA = fspecial('average', 7);  
B = imfilter(A, avgfilterA);  
  
subplot(1, 2, 1);  
imshow(A);  
subplot(1, 2, 2);  
imshow(B);
```

- b) Input and Filtered Images:



b. 7x7 Gaussian filter to “lena\_gray.jpg” image:

a) Source Code:

```
clear;  
A = imread('lena_gray.jpg');  
  
gaussianfiltA = fspecial('gaussian', 7, 5);  
B = imfilter(A, gaussianfiltA);  
  
subplot(1, 2, 1);  
imshow(A);  
subplot(1, 2, 2);  
imshow(B);
```

b) Input and Filtered Images:



c. 5x5 Median filter to “noisy\_img.png” image:

a) Source Code:

```
clear;  
A = imread('noisy_img.png');  
  
B = medfilt2(A, [5 5]);  
  
subplot(1, 2, 1);  
imshow(A);  
subplot(1, 2, 2);  
imshow(B);
```

b) Input and Filtered Images:



d. Any Sharpening filter to “lena\_gray.jpg” image:

a) Source Code:

```
clear;  
A = imread('lena_gray.jpg');  
  
B = imsharpen(A);  
  
subplot(1, 2, 1);  
imshow(A);  
subplot(1, 2, 2);  
imshow(B);
```

b) Input and Filtered Images:



e. Sobel horizontal filter to “cameraman.tif” image:

a) Source Code:

```
clear;  
A = imread('cameraman.tif');  
  
sobelhfilterA = fspecial('sobel');  
B = imfilter(A, sobelhfilterA);  
  
subplot(1, 2, 1);  
imshow(A);  
subplot(1, 2, 2);  
imshow(B);
```

b) Input and Filtered Images:





- f. Sobel vertical filter to “cameraman.tif” image:  
a) Source Code:

```
clear;  
A = imread('cameraman.tif');  
  
sobelhfilterA = fspecial('sobel');  
sobelvfilterA = transpose(sobelhfilterA);  
B = imfilter(A, sobelvfilterA);  
  
subplot(1, 2, 1);  
imshow(A);  
subplot(1, 2, 2);  
imshow(B);
```

- b) Input and Filtered Images:

