IMAGE PROCESSING I

IntuoN Lertrusdachakul

Tutorial III: Spatial Filtering

Intro…

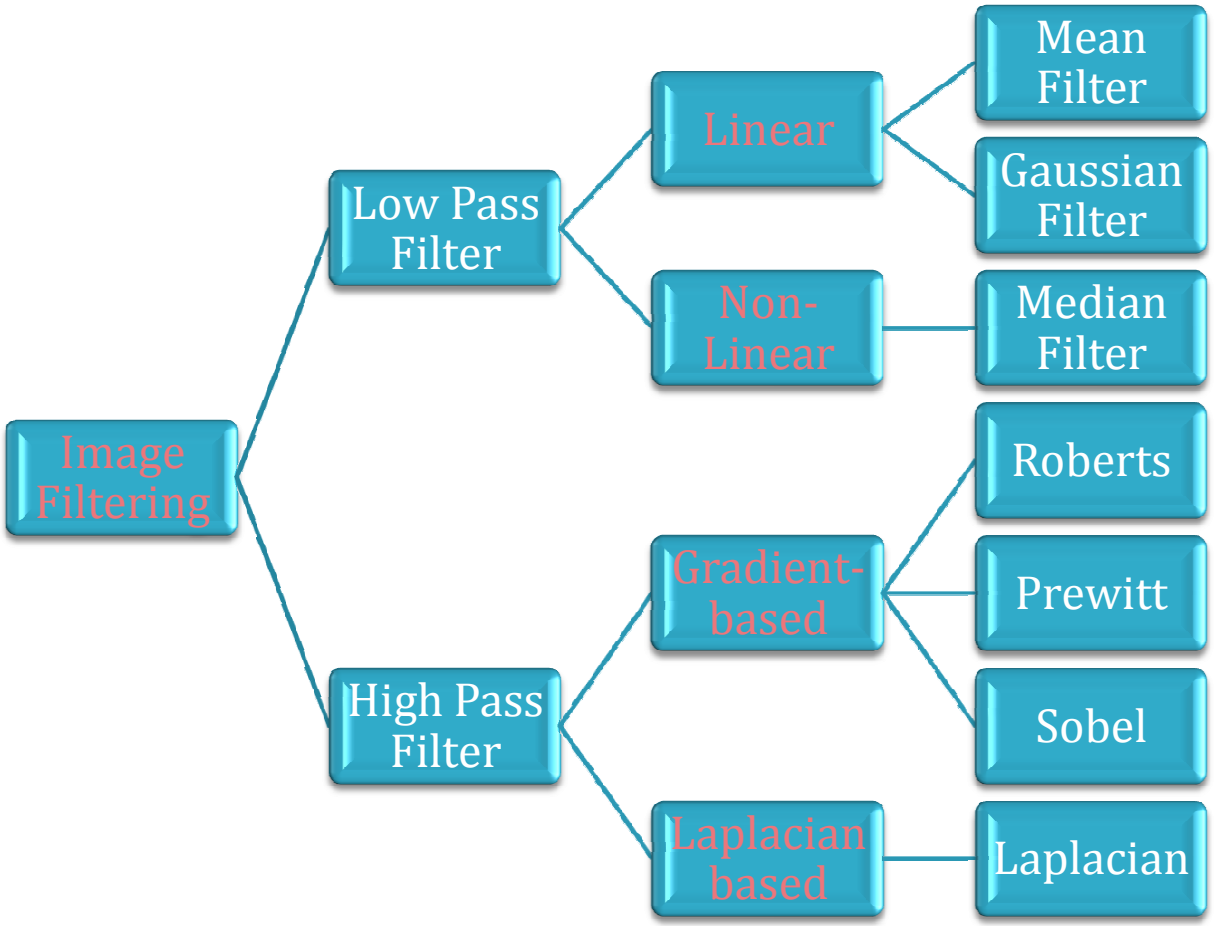


Image Filtering
Low Pass Filter
High Pass Filter
Linear
Non-Linear
Gradient-based
Laplacian based
Mean Filter
Gaussian Filter
Median Filter
Roberts
Prewitt
Sobel
Laplacian

Outline

- Perform Different Filtering

  (observe the effect to the image quality)

  - Manually
  - By built-in command (e.g. fspecial, medfilt2, conv2, filter2, etc.)
  - By your own code

Practical Session

6	6	6	6	6	6	6
5	5	5	5	5	5	5
4	4	4	4	4	4	4
3	3	3	10	3	3	3
2	2	2	2	2	2	2
1	1	1	1	1	1	1
0	0	0	0	0	0	0

A Digital Image

Perform LPF (Mean Filtering)

$$* \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad / 9$$

A Convolution Mask

Practical Session

6	6	6	6	6	6	6
5	5	5	5	5	5	5
4	4	4	4	4	4	4
3	3	3	10	3	3	3
2	2	2	2	2	2	2
1	1	1	1	1	1	1
0	0	0	0	0	0	0

A Digital Image

Perform LPF
(Weighted Averaging Filtering)

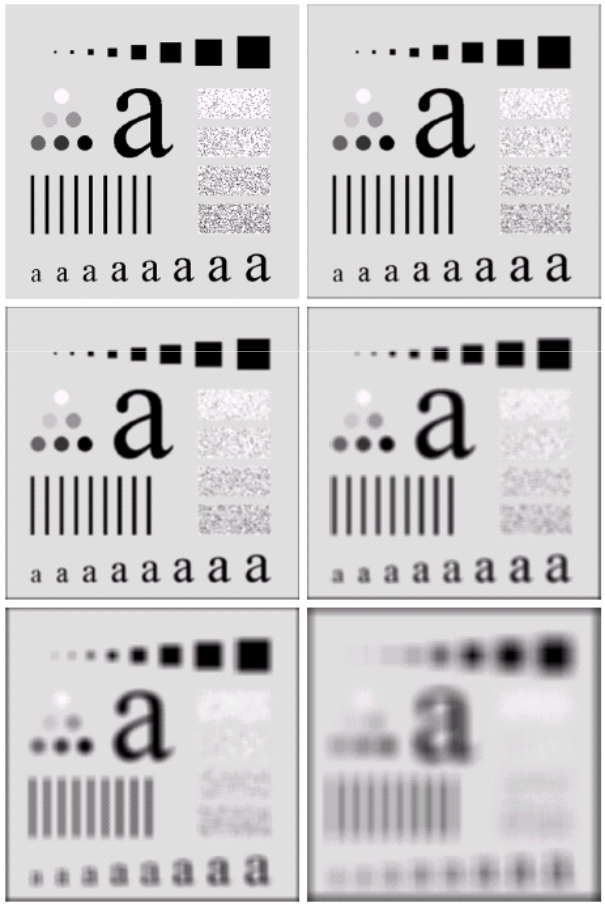
$$* \quad \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} \quad / 16$$

A Convolution Mask

Sample Code


```
%Define an image

img=[  6 6 6 6 6 6 6; 5 5 5 5 5 5 5;
       4 4 4 4 4 4 4; 3 3 3 10 3 3 3;
       2 2 2 2 2 2 2; 1 1 1 1 1 1 1;0 0 0 0 0 0 0]

%Define a mask for filtering

mask=fspecial('average',3) OR
mask=[1 1 1; 1 1 1; 1 1 1]/9

%Filtering in two ways

output=filter2(mask,img,'same') OR
output=conv2(img,mask,'same')
```

Effect of Low Pass Filtering



Practical Session

Perform LPF (Median Filtering)

Example 1

5	5	5	5	5
4	4	4	4	4
3	3	10	3	3
2	2	2	2	2
1	1	1	1	1

Practical Session

Perform LPF (Median Filtering)

Example 2

0	0	0	0	0
0	0	0	0	0
0	0	10	10	10
0	0	10	10	10
0	0	10	10	10

Practical Session

LPF (Median Filtering)

Example 3

10	10	0	10	10
10	10	0	10	10
10	10	0	10	10
10	10	0	10	10
10	10	0	10	10

Practical Session

LPF (Median Filtering)

Example 4

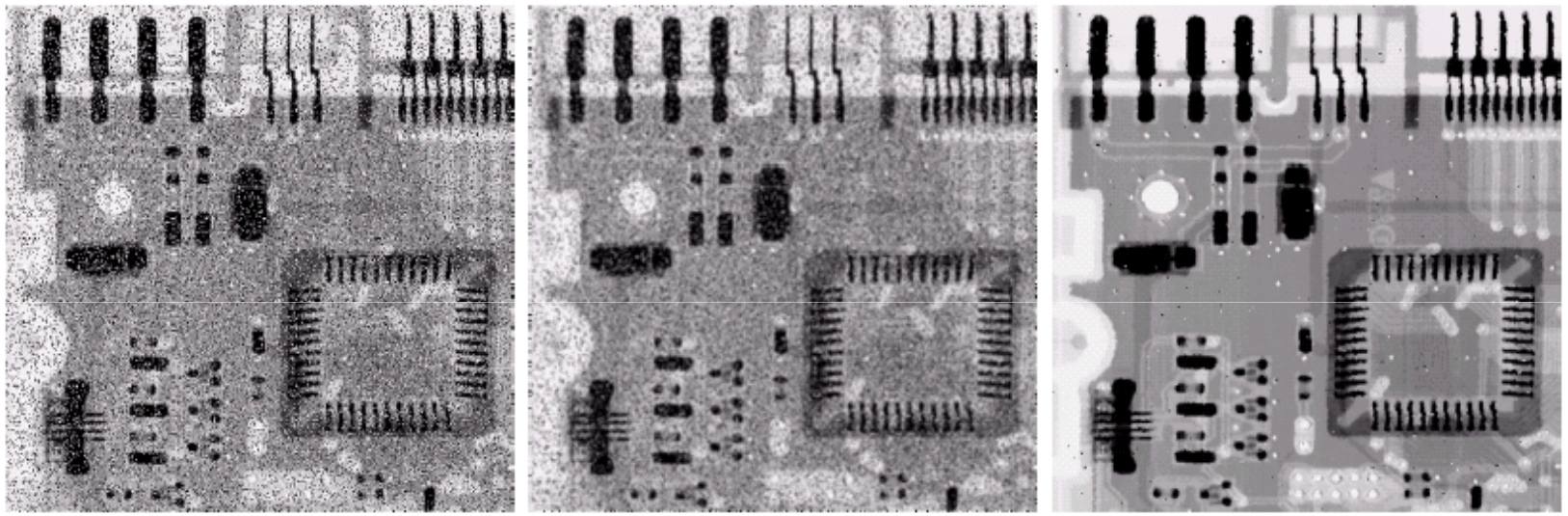
0	0	10	0	0
0	0	10	0	0
0	0	10	0	0
0	0	10	0	0
0	0	10	0	0

Sample Code


```
% Define an image

img=[5 5 5 5 5; 4 4 4 4 4; 3 3 10 3 3; 2 2 2 2 2; 1
  1 1 1 1]

% Median filtering

output=medfilt2(img,[3 3],'symmetric')
```

.

Effect of Median Filtering



a b c

FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3×3 averaging mask. (c) Noise reduction with a 3×3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

HPF (Gradient-based)

a
b c
d e
f g

FIGURE 10.8
A 3 × 3 region of an image (the z's are gray-level values) and various masks used to compute the gradient at point labeled z_5.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0	0	−1
0	1	1	0

Roberts

−1	−1	−1	−1	0	1
0	0	0	−1	0	1
1	1	1	−1	0	1

Prewitt

−1	−2	−1	−1	0	1
0	0	0	−2	0	2
1	2	1	−1	0	1

Sobel

Practical Session

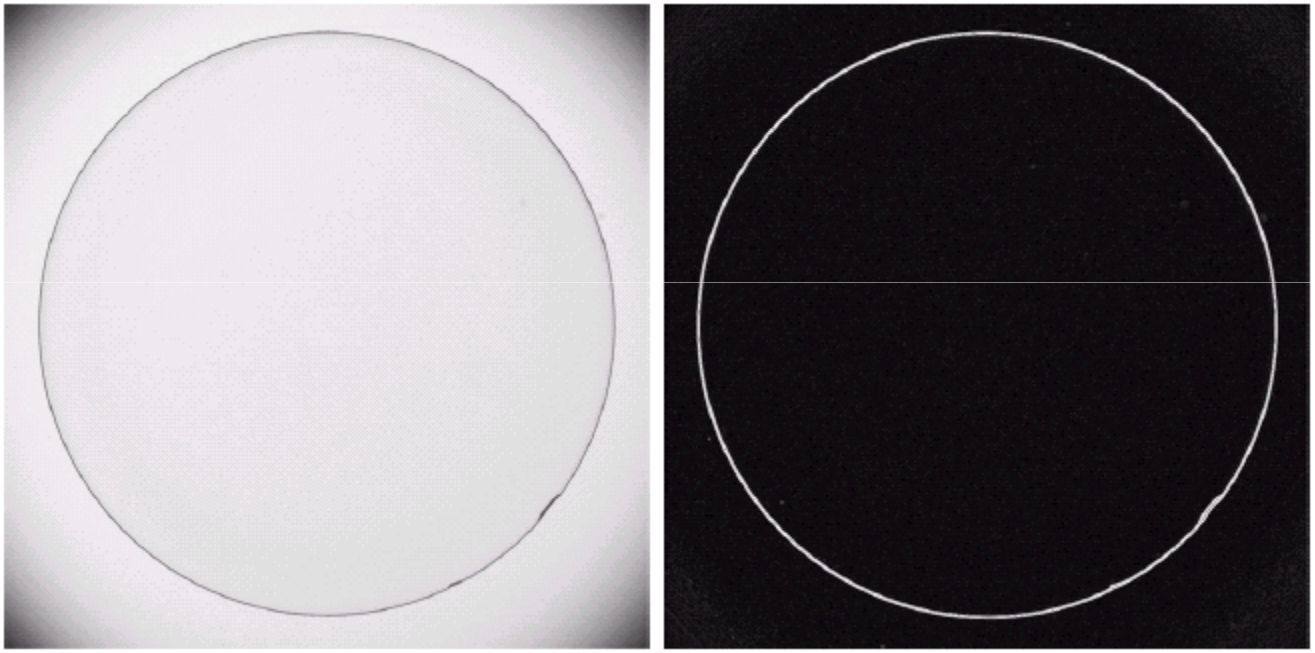
Perform HPF
(Gradient-based)
Prewitt Operators

0	0	0	0	0	0	0
0	0	0	0	0	0	1
0	0	0	0	0	1	1
0	0	0	0	1	1	1
0	0	0	1	1	1	1
0	0	1	1	1	1	1
0	1	1	1	1	1	1

Sample Code


```
% Call the Prewitt operators
mask1=fspecial('Prewitt')
mask2=mask1'

% Filtering
tmp1=conv2(img,mask1,'same');
tmp2=conv2(img,mask2,'same');

% Calculate the magnitude
mag=sqrt(tmp1.^2+tmp2.^2);

% Display it as an image
imshow(mat2gray(mag))
```

Edge Detection by Sobel Operators



a b

FIGURE 3.45
Optical image of contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Mr. Pete Sites, Perceptics Corporation.)

HPF (Laplacian-based)

0	1	0	1	1	1
1	−4	1	1	−8	1
0	1	0	1	1	1

0	−1	0	−1	−1	−1
−1	4	−1	−1	8	−1
0	−1	0	−1	−1	−1

a b
c d

FIGURE 3.39
(a) Filter mask used to implement the digital Laplacian, as defined in Eq. (3.7-4). (b) Mask used to implement an extension of this equation that includes the diagonal neighbors. (c) and (d) Two other implementations of the Laplacian.

Practical Session

0	0	0	0	0	0	0
0	0	0	0	0	0	1
0	0	0	0	0	1	1
0	0	0	0	1	1	1
0	0	0	1	1	1	1
0	0	1	1	1	1	1
0	1	1	1	1	1	1

Perform HPF
(Laplacian Operators)

*

1	1	1
1	-8	1
1	1	1

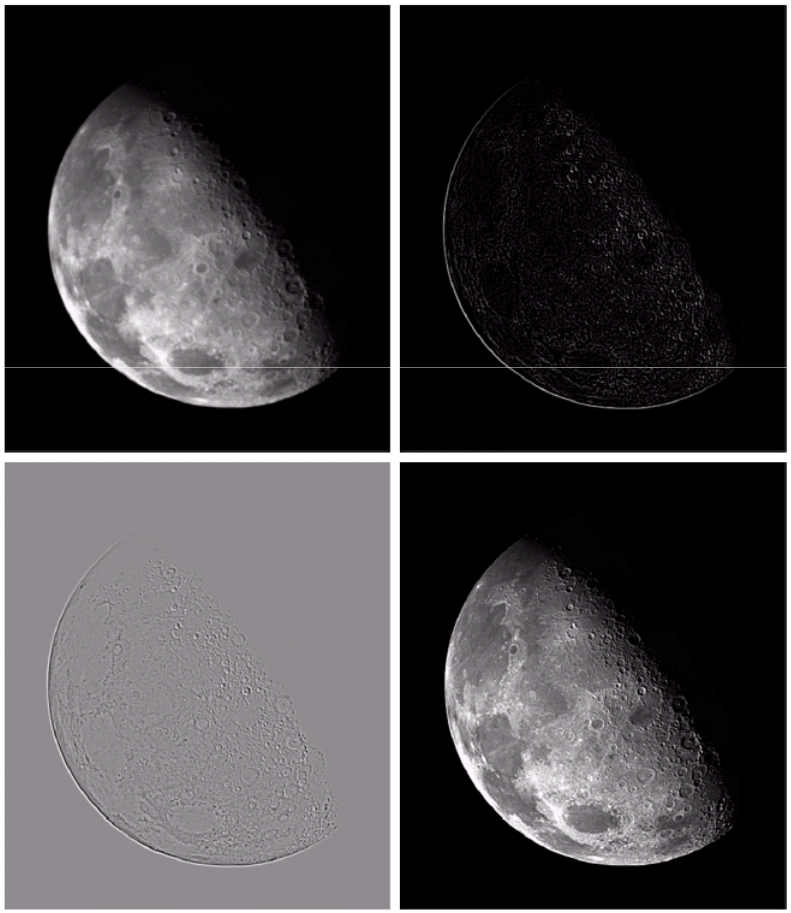
Sample Code


```
% Call the Laplacian operator
mask=fspecial('Laplacian')

% Filtering
tmp=conv2(img,mask,'same');

% Display it as an image
imshow(mat2gray(tmp))

% Enhancement
tmp=img-tmp*0.1

% Display the enhanced image
imshow(mat2gray(tmp))
```

Image Enhancement by Laplacian

a b
c d

FIGURE 3.40
(a) Image of the North Pole of the moon.
(b) Laplacian-filtered image.
(c) Laplacian image scaled for display purposes.
(d) Image enhanced by using Eq. (3.7-5).
(Original image courtesy of NASA.)



Assignment

- Add ‘Gaussian’ and ‘Salt&Pepper’ noise to your image separately. Then, apply Averaging filters and Median filters to the noisy images.

- Apply the Sobel and Laplacian filters to your original facial image separately. Also attempt to enhance your image using the Laplacian filter.

Additional Material

Sample Results
From Gaussian Noisy Image

Original Image

Noisy Image (Gaussian)

Averaging Filtered

Median Filtered

Additional Material

Sample Results
From Salt & Pepper Noisy Image

Original Image

Noisy Image (Salt&Pepper)

Averaging Filtered

Median Filtered

Additional Material

Sample Results
From laplacian Enhancement