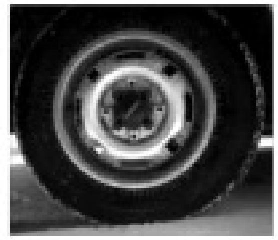
Local Enhancement

- Local Enhancement
  - Median filtering

Local enhancement

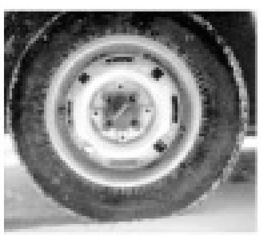


Sometimes Local Enhancement is Preferred.

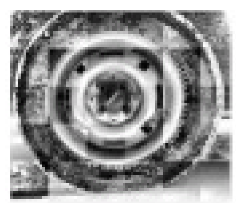
Malab: BlkProc operation for block processing.

Left: original "tire" image.

Histogram equalized

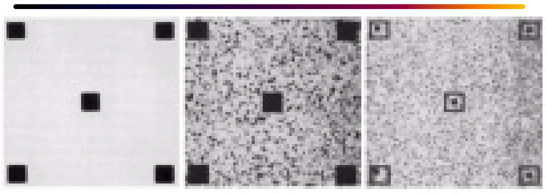


Local histogram equalized



```
F=@ histeq;
I=imread('tire.tif');
J=blkproc(I,[20 20], F);
```


Fig 3.23: Another example



a b c

FIGURE 3.23 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7 × 7 neighborhood about each pixel.

Local Contrast Enhancement

- Enhancing local contrast

$$g(x,y) = A(x,y)\,[\,f(x,y) - m(x,y)\,] + m(x,y)$$

$$A(x,y) = k\,M / \sigma(x,y) \qquad 0 < k < 1$$

M : Global mean

$m(x,y)$, $\sigma(x,y)$: Local mean and standard dev.

Areas with low contrast ➔ Larger gain $A(x,y)$ (fig 3.24-3.26)

Fig 3.24

FIGURE 3.24 SEM image of a tungsten filament and support, magnified approximately 130×. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

Fig 3.25

FIGURE 3.25 (a) Image formed from all local means obtained from Fig. 3.24 using Eq. (3.3-21). (b) Image formed from all local standard deviations obtained from Fig. 3.24 using Eq. (3.3-22). (c) Image formed from all multiplication constants used to produce the enhanced image shown in Fig. 3.26.

Fig 3.26

FIGURE 3.26 Enhanced SEM image. Compare with Fig. 3.24. Note in particular the enhanced area on the right side of the image.

Image Subtraction

g (x,y) = f (x,y) - h (x,y)

h(x,y)—a low pass filtered version of f(x,y).

- Application in medical imaging --“mask mode radiography”
- H(x,y) is the mask, e.g., an X-ray image of part of a body; f(x,y) –incoming image after injecting a contrast medium.

Subtraction: an example

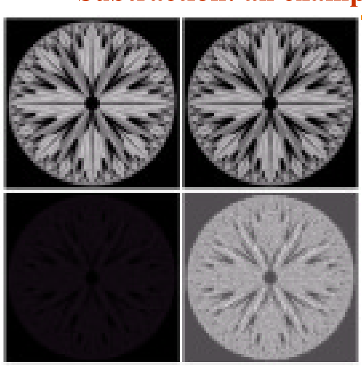


Fig 3.28: mask mode radiography

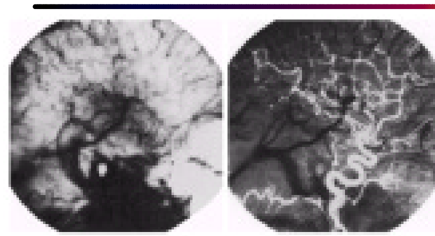


FIGURE 3.28 Enhancement by image subtraction. (a) Mask image. (b) An image (taken after injection of a contrast medium into the bloodstream) with mask subtracted out.

Averaging

$g(x,y) = f(x,y) + \eta(x,y)$

$\bar{g}(x,y) = \frac{1}{M}\sum_{i=1}^{M} g_i(x,y)$

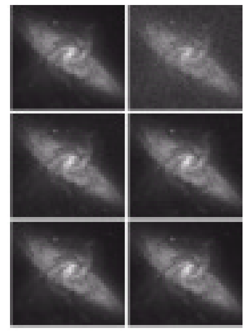
$E(\bar{g}(x,y)) = f(x,y)$ and $\sigma^2_{\bar{g}} = \frac{1}{M}\sigma^2_{\eta}(x,y)$

$\eta(x,y) \rightarrow$ Uncorrelated zero mean

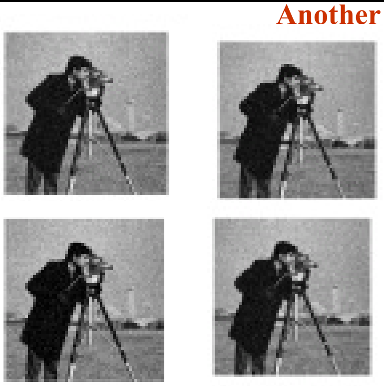
$\sigma^2_{\eta}(x,y) \rightarrow$ Reduces the noise variance

Fig 3.30

Fig 3.30



Another example



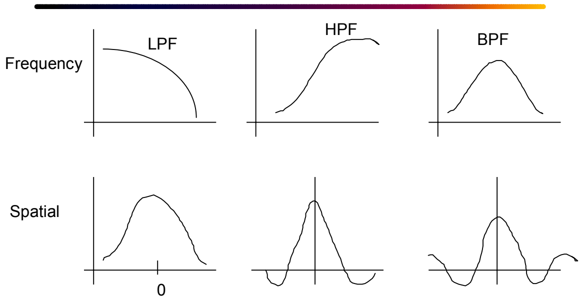
Images with additive Gausian Noise; Independent Samples.

I=imnoise(J,'Gaussian');

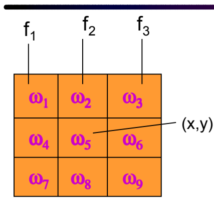
Averaged image

Left: averaged image (10 samples);
Right: original image

Spatial filtering



Smoothing (Low Pass) Filtering



Replace f (x,y) with

$$\hat{f}(x,y) = \sum_i \omega_i f_i$$

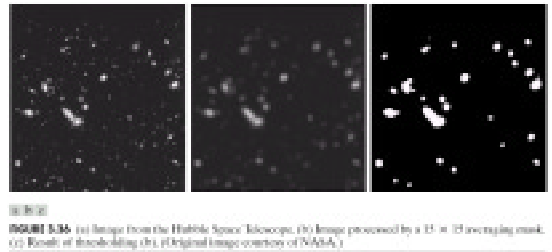
Linear filter

LPF: reduces additive noise → blurs the image
→ sharpness details are lost
(Example: Local averaging)

Fig 3.35

Fig 3.35: smoothing

Fig 3.36: another example



Median filtering

Replace f (x,y) with median [f (x' , y')]
(x' , y') ε neighbourhood

- Useful in eliminating intensity spikes. (salt & pepper noise)
- Better at preserving edges.

Example:

10	20	20
20	15	20
25	20	100

⟶ (10,15,20,20,20,20,20,25,100)

Median=20

So replace (15) with (20)

Median Filter: Root Signal

Repeated applications of median filter to a signal results in an invariant signal called the "root signal".
A root signal is invariant to further application of the medina filter.

Example: 1-D signal: Median filter length = 3

0 0 0 1 2 1 2 1 2 1 0 0 0
0 0 0 1 1 2 1 2 1 1 0 0 0
0 0 0 1 1 1 2 1 1 1 0 0 0
0 0 0 1 1 1 1 1 1 1 0 0 0 root signal

Invariant Signals

Invariant signals to a median filter:

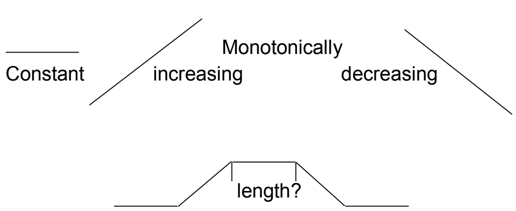
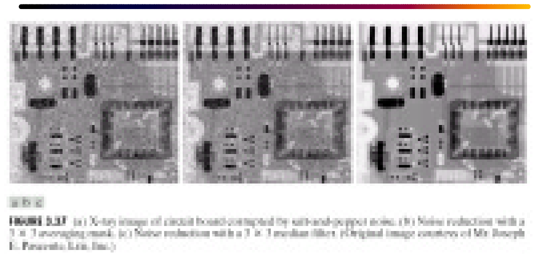


Fig 3.37: Median Filtering example



Media Filter: another example

Original and with salt & pepper noise
imnoise(image, 'salt & pepper');

Donoised images

Local averaging
K=filter2(fspecial('average',3),image)/255.

Median filtered
L=medfil2(image, [3 3]);

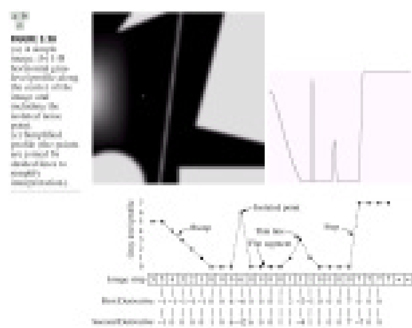
Sharpening Filters

- Enhance finer image details (such as edges)
- Detect region /object boundaries.

Example:

−1	−1	−1
−1	8	−1
−1	−1	−1

Edges (Fig 3.38)

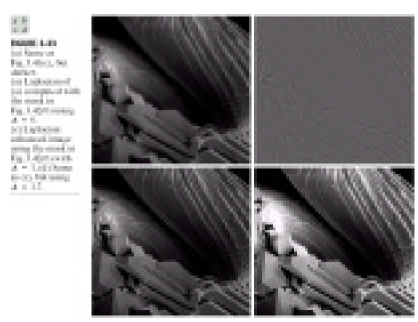


Unsharp Masking

Subtract Low pass filtered version from the original emphasizes high frequency information

I' = A (Original) - Low pass
HP = O - LP A > 1
I' = (A - 1) O + HP
A = 1 => I' = HP
A > 1 => LF components added back.

Fig 3.43 –example of unsharp masking



Derivative Filters

1/9

−1	−1	−1
−1	ω	−1
−1	−1	−1

Gradient

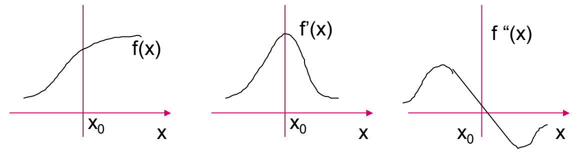
$$\nabla f = \left[\frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y}\right]^T$$

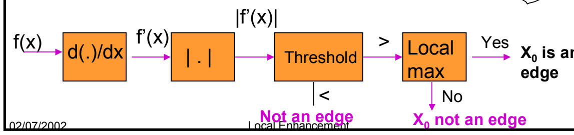
$$\|\nabla f\| = \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right]^{1/2}$$

Edge Detection

Gradient based methods

$$\nabla f = \left(\frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y}\right)^T$$





Digital edge detectors

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$$|\nabla f| \approx \left[(z_5 - z_8)^2 + (z_5 - z_6)^2\right]^{1/2}$$

$$|\nabla f| \approx |z_5 - z_8| + |z_5 - z_6|$$

Robert's operator

1	0
0	-1

0	1
-1	0

$| z_5 - z_9 |$ $| z_6 - z_8 |$

prewitt

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

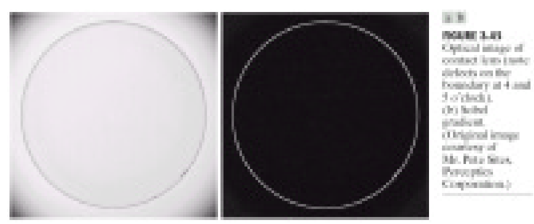
-1	0	1
-1	0	1
-1	0	1

Sobel's

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Fig 3.45: Sobel edge detector



Laplacian based edge detectors

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

	1	
1	-4	1
	1	

- Rotationally symmetric, linear operator
- Check for the zero crossings to detect edges
- Second derivatives => sensitive to noise.

Fig 3.40: an example