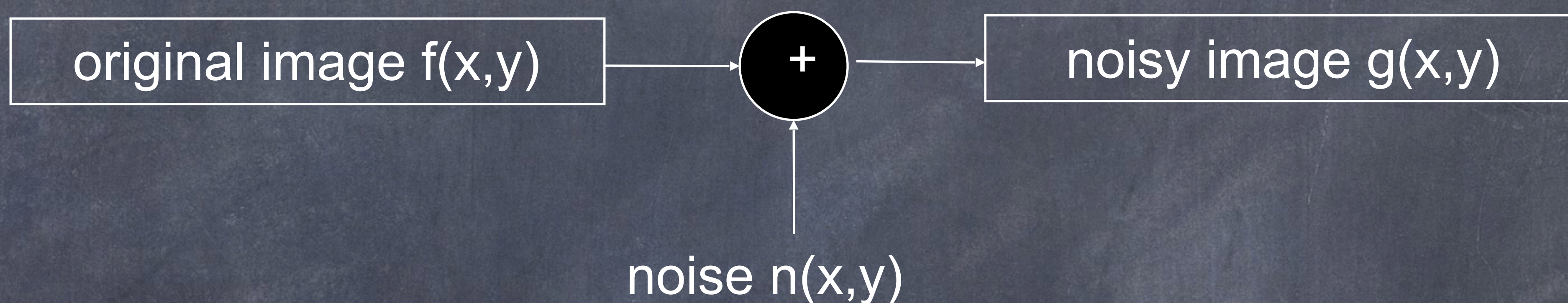
Computer Vision

CSE/ECE 344/544

Filter an image

What do you understand by image filtering?

Image Enhancement - Point Processing



$$g(x,y) = f(x,y) + n(x,y)$$

$$\sum_{i=1}^{N} g_i(x,y) = \sum_{i=1}^{N} f(x,y) + \sum_{i=1}^{N} n(x,y)$$

$$\bar{g}(x,y) = \bar{f}(x,y) + \bar{n}(x,y)$$

Noisy Image

Original image

Noised image (Gaussian)

Image Enhancement

Objective of enhancement is to process image so that result is more suitable than original image for specific application.

Enhancement approaches fall into two broad categories

spatial domain : direct manipulation of pixels in an image
frequency domain : manipulation of Fourier transform of an image.

Basics of Spatial Filtering - Linear

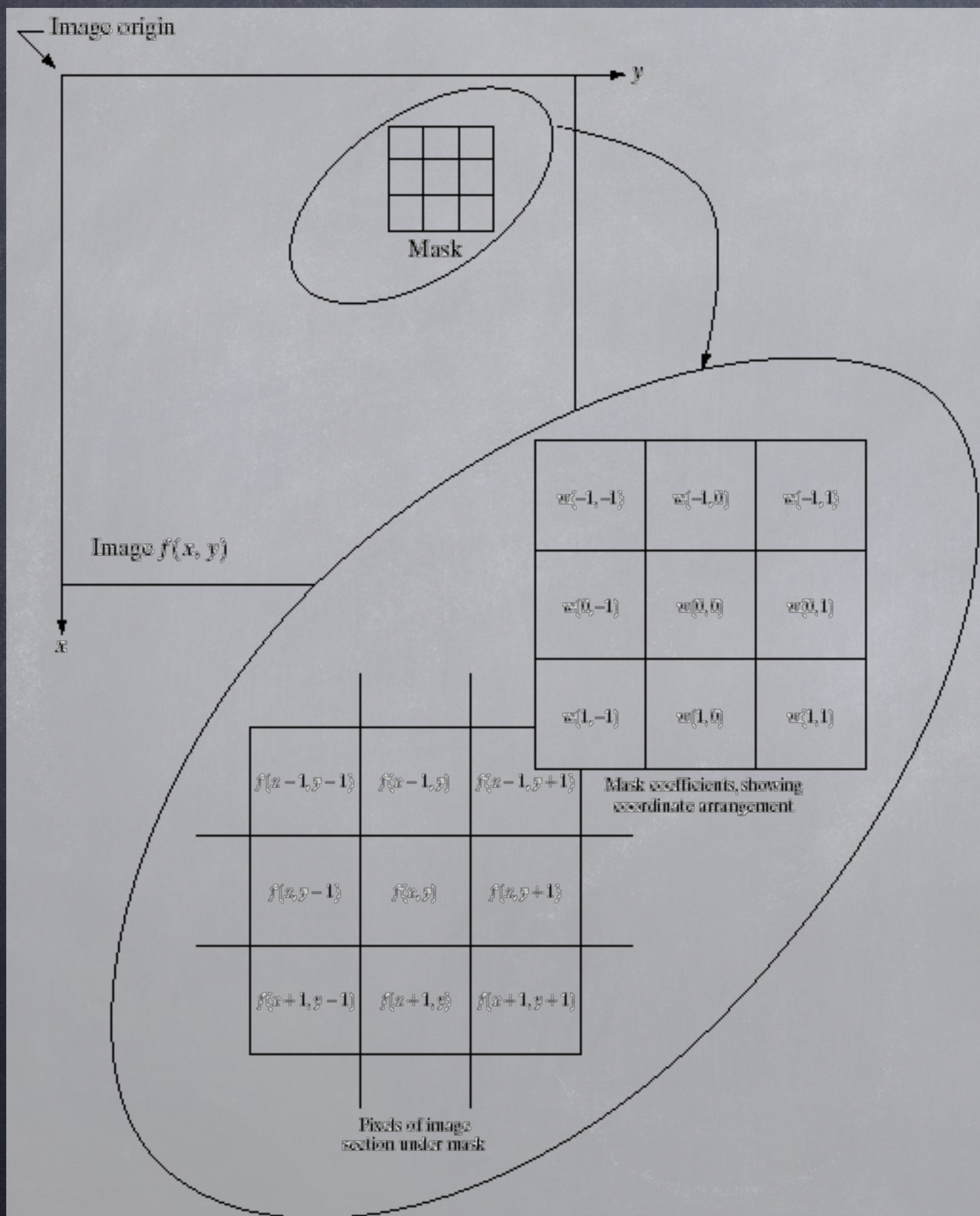


FIGURE 3.32 The mechanics of spatial filtering. The magnified drawing shows a 3 × 3 mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.

Spatial filtering are filtering operations performed on the pixel intensities of an image.

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

$a = (m - 1) / 2$ $\qquad$ $b = (n - 1) / 2$

Basic of spatial filtering

- Linear spatial filtering often is referred to as "convolving a mask with an image"

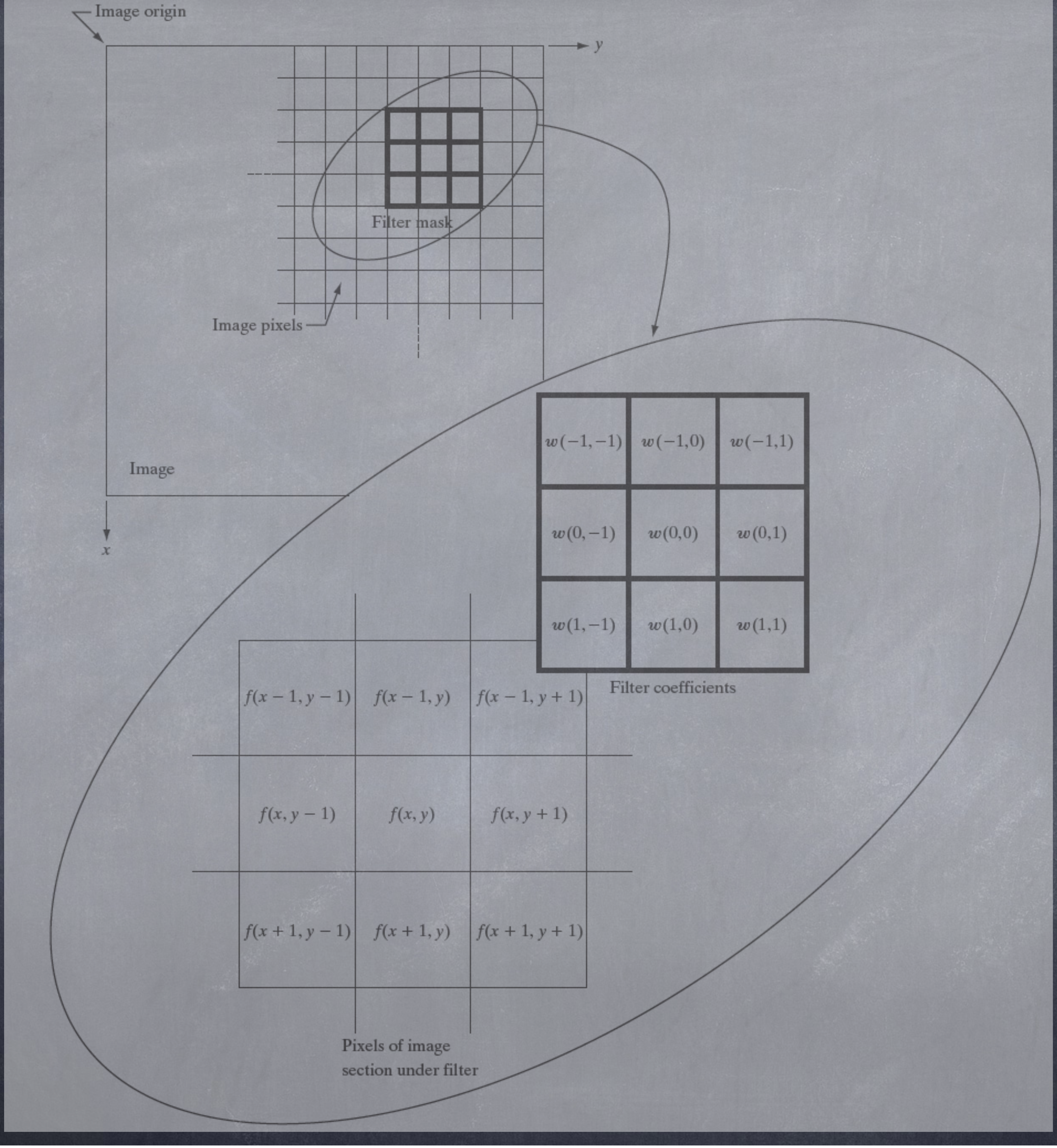
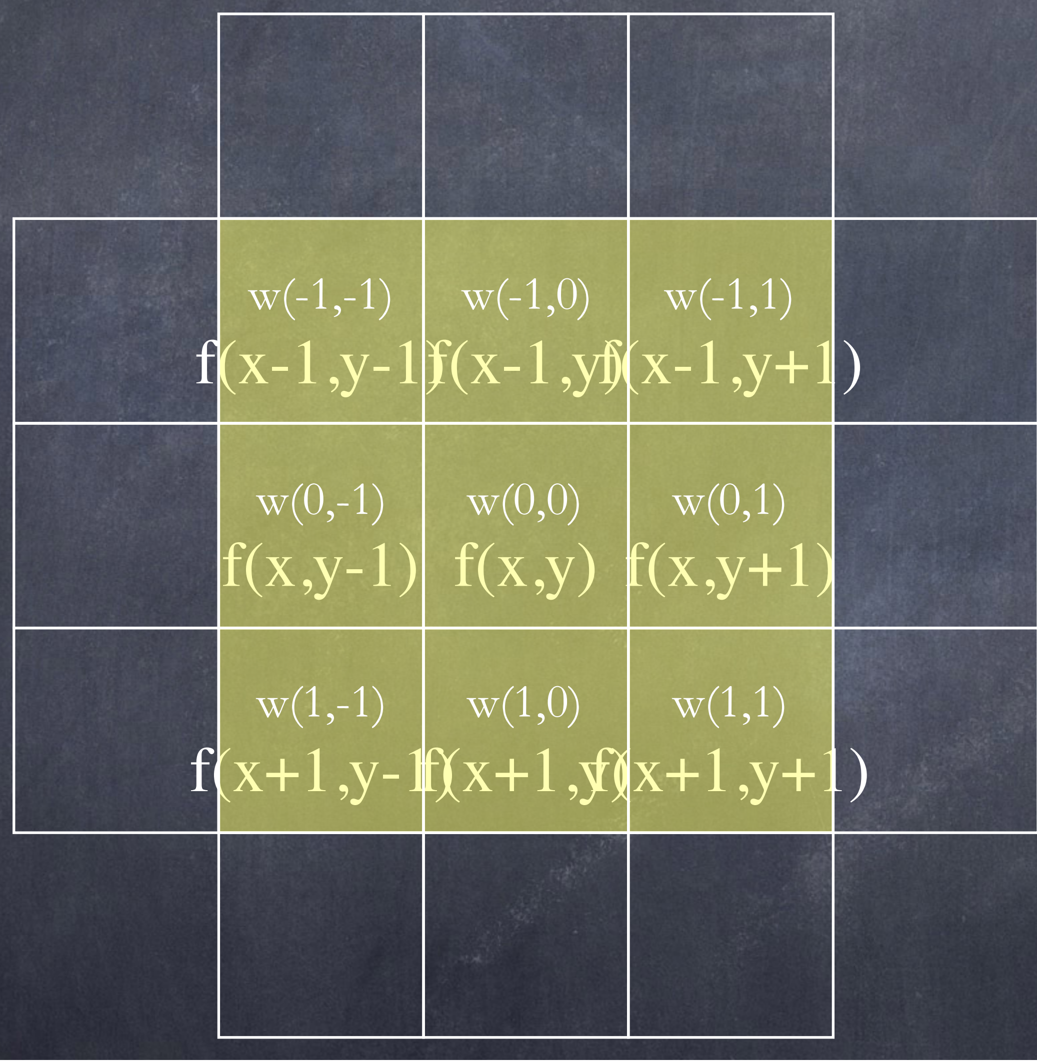


FIGURE 3.28 The mechanics of linear spatial filtering using a 3×3 filter mask. The form chosen to denote the coordinates of the filter mask coefficients simplifies writing expressions for linear filtering.

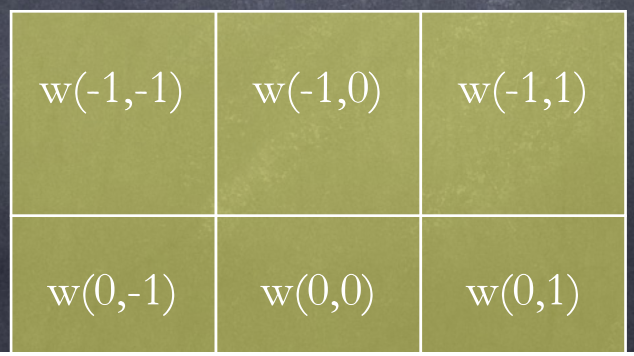
Linear spatial filtering

Pixels of image



The result is the sum of products of the mask coefficients with the corresponding pixels directly under the mask

Mask coefficients



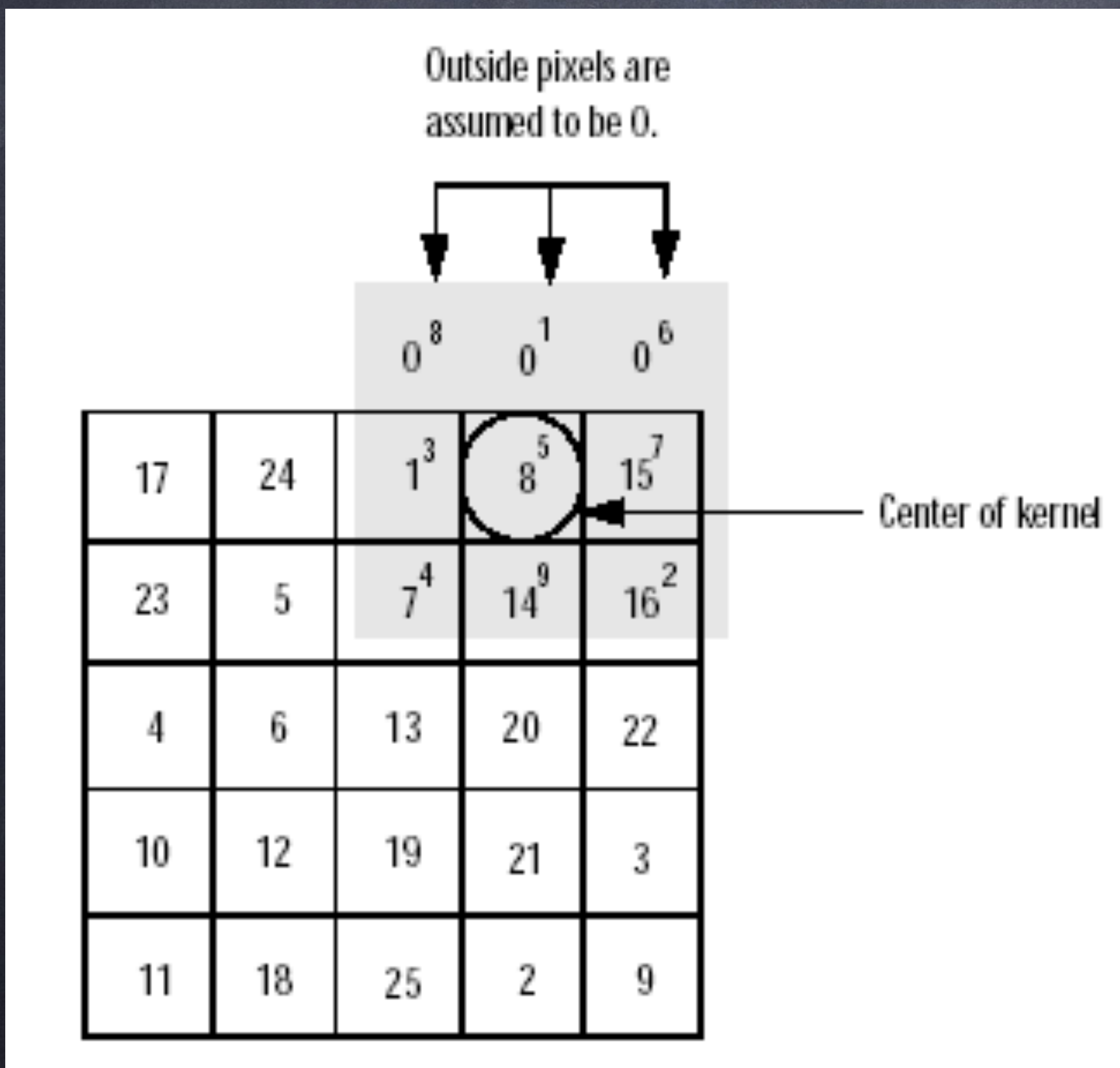
$$f(x,y) = w(-1,-1)\,f(x-1,y-1) + w(-1,0)\,f(x-1,y) + w(-1,1)\,f(x-1,y+1) + w(0,-1)\,f(x,y-1) + w(0,0)\,f(x,y) + w(0,1)\,f(x,y+1) + w(1,-1)\,f(x+1,y-1) + w(1,0)\,f(x+1,y) + w(1,1)\,f(x+1,y+1)$$

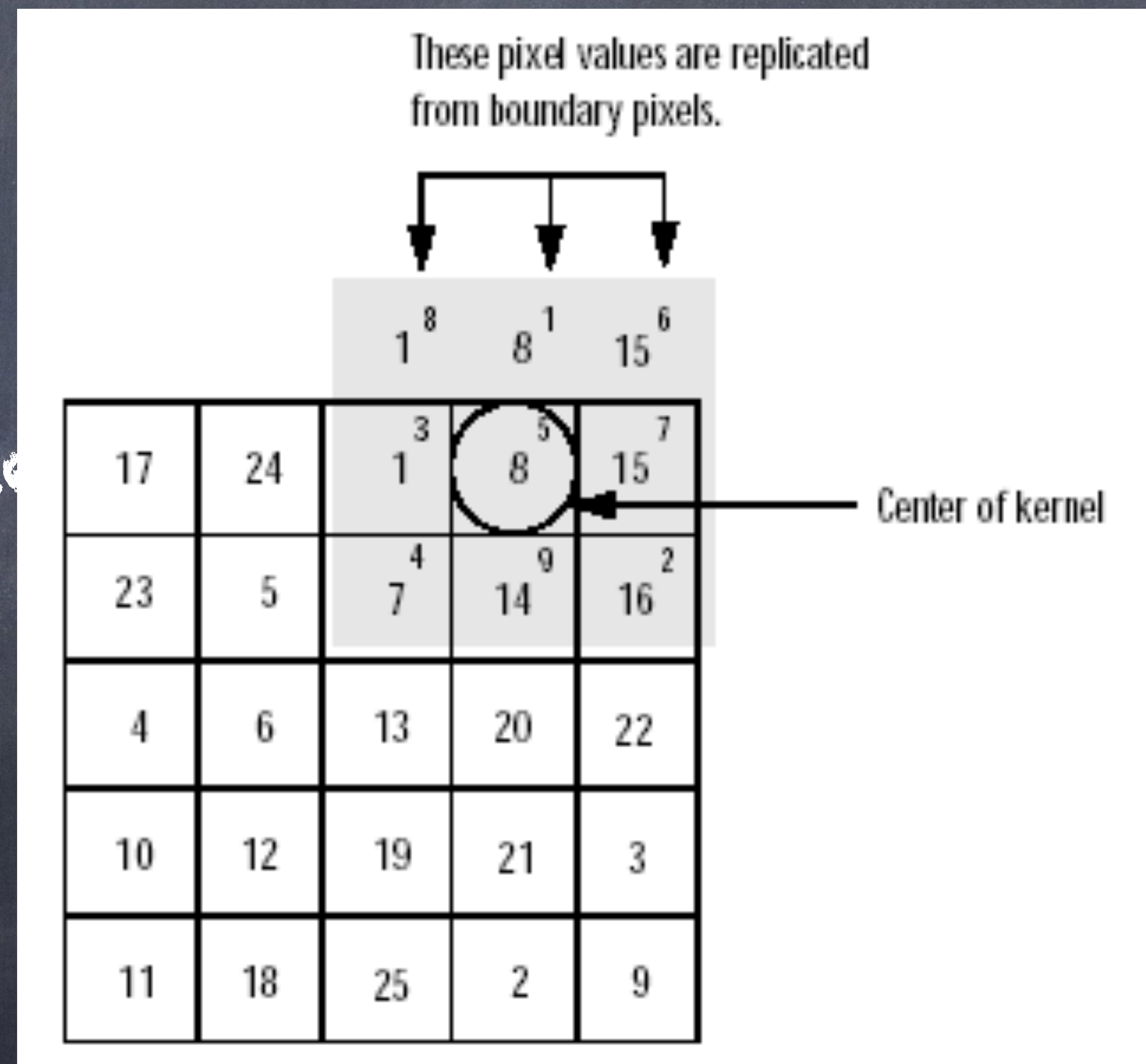
Image Enhancement

Operation Steps

- Select a single pixel.
- Determine the pixel's neighborhood.
- Apply a function to the values of the pixels in the neighborhood. This function must return a scalar.
- Find the pixel in the output image whose position corresponds to that of the center pixel in the input image. Set this output pixel to the value returned by the function.
- Repeat steps 1 through 4 for each pixel in the input image.

Image Enhancement





w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

FIGURE 3.31 Another representation of a general 3×3 filter mask.

Square Averaging Filter Mask

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \qquad \frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

a b

FIGURE 3.32 Two 3×3 smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to 1 divided by the sum of the values of its coefficients, as is required to compute an average.

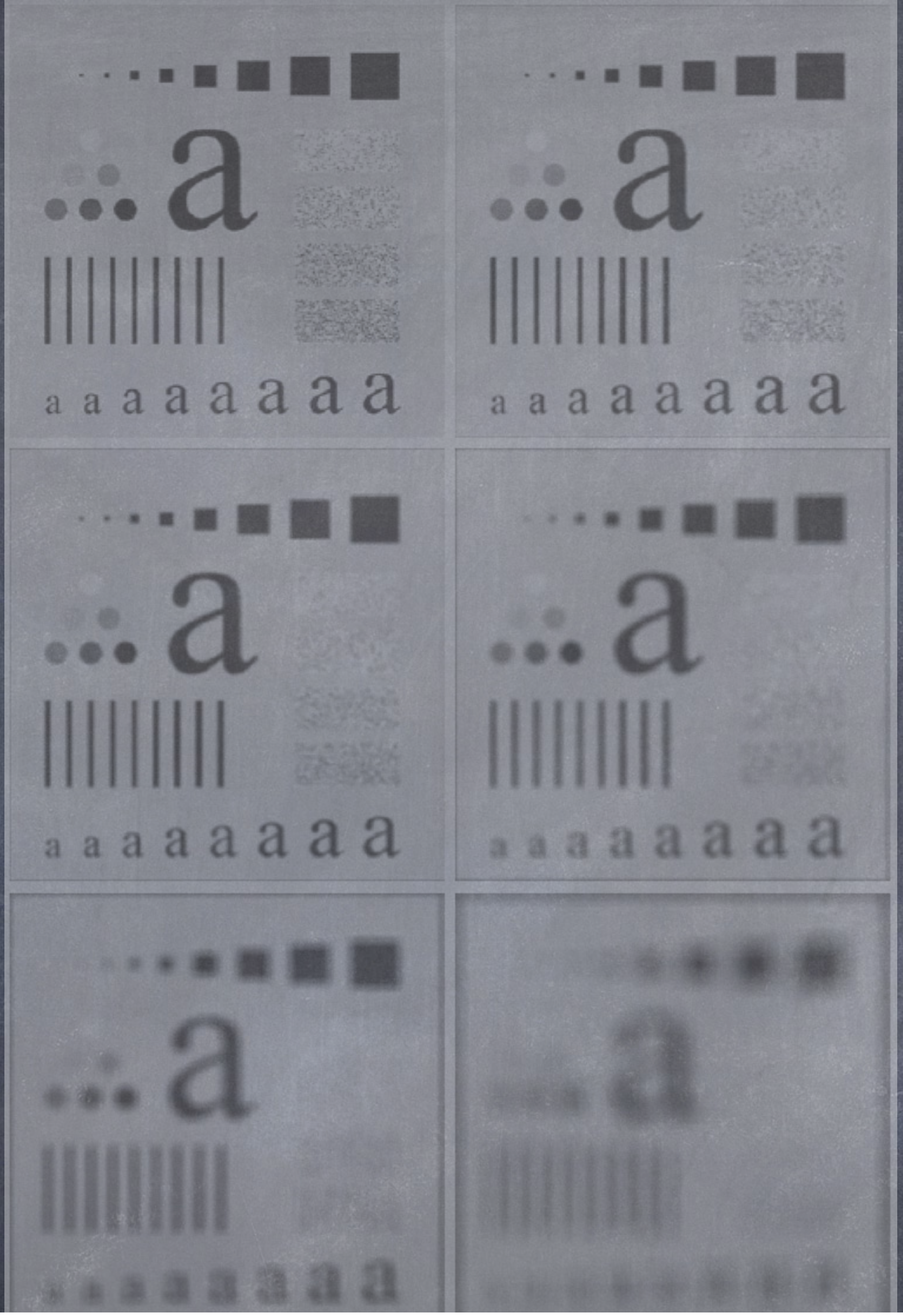


FIGURE 3.33 (a) Original image, of size 500 × 500 pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes $m = 3, 5, 9, 15$, and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart. The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points. The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels. The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their intensity levels range from 0% to 100% black in increments of 20%. The background of the image is 10% black. The noisy rectangles are of size 50 × 120 pixels.

a b
c d
e f

Order-Statistics Filters

- Order-statistics filters are nonlinear spatial filters whose response is based on ordering (ranking) the pixels contained in the image area encompassed by the filter, and then replacing the value of the center pixel with the value determined by the ranking result.
- Best-known "median filter"

Order-Statistics Filters

Median filter examines the neighborhood pixel values and sort them, replace the current pixel value by the median value.

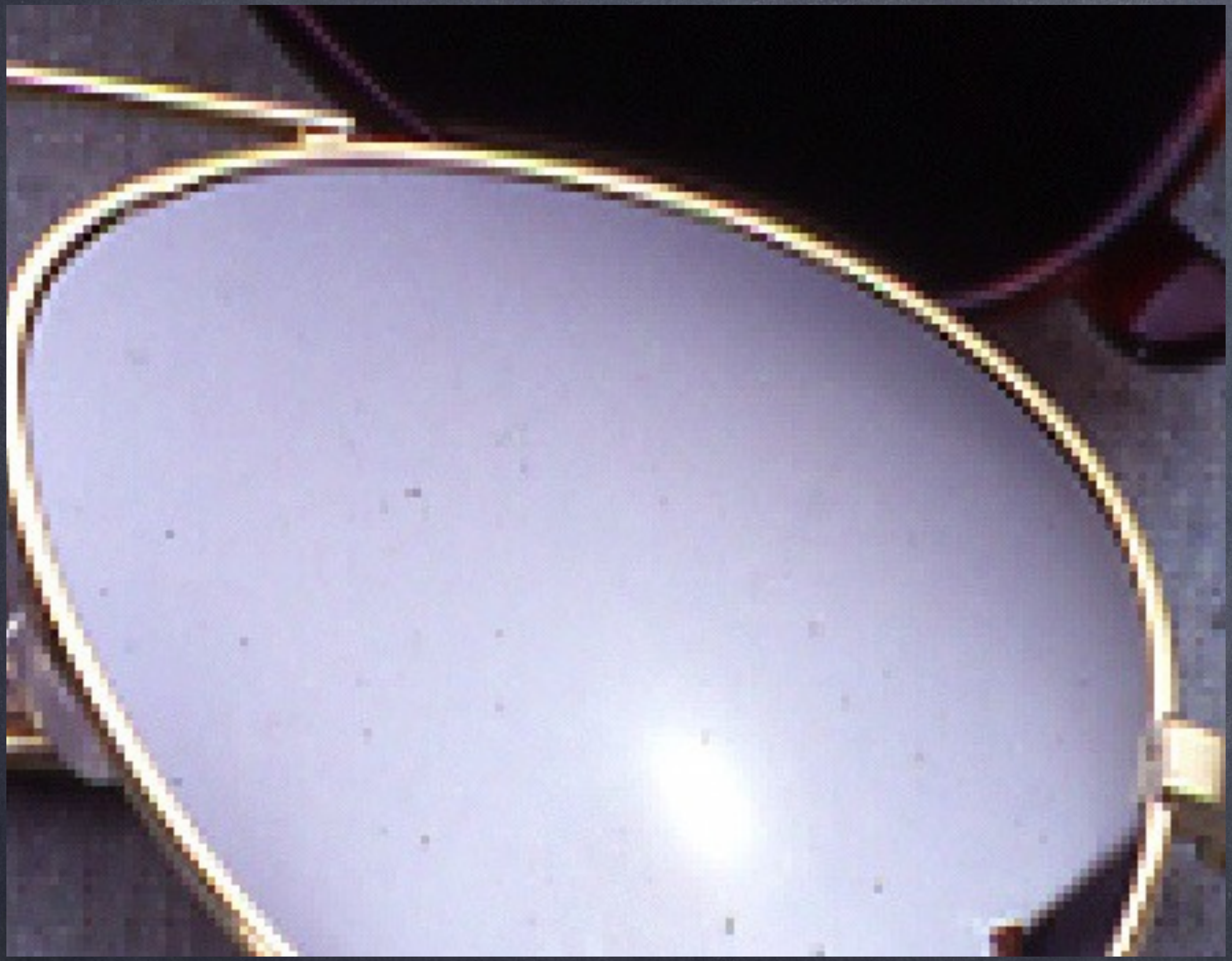
Example:

37	41	39
40	234	38
42	38	44

Sorted sequence:

37, 38, 38, 39, 40, 41, 42, 44, 234

Result of median filter



Noise from Glass effect   Remove noise by median filter

Sharpening Spatial Filters

- The principal objective of sharpening is to highlight fine detail in an image or to enhance detail that has been blurred, either in error or as an natural effect of a particular method of image acquisition.

Sharpening Spatial Filters

- The image blurring is accomplished in the spatial domain by pixel averaging in a neighborhood.
- Averaging is analogous to integration.
- Sharpening could be accomplished by spatial differentiation.

Definition of the 1^{st}-order derivative

- A basic definition of the first-order derivative of a one-dimensional function f(x) is

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

Definition of the 2nd-order derivative

- We define a second-order derivative as the difference

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x).$$

The Laplacian (2nd order derivative)

- Shown by Rosenfeld and Kak[1982] that the simplest isotropic derivative operator is the Laplacian, defined as

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

Discrete form of derivative

$f(x-1,y)$	$f(x,y)$	$f(x+1,y)$

$$\frac{\partial^2 f}{\partial x^2} = f(x+1,y) + f(x-1,y) - 2f(x,y)$$

$f(x,y-1)$
$f(x,y)$
$f(x,y+1)$

$$\frac{\partial^2 f}{\partial y^2} = f(x,y+1) + f(x,y-1) - 2f(x,y)$$

2-Dimentional Laplacian

- The digital implementation of the 2-Dimensional Laplacian is obtained by summing 2 components

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

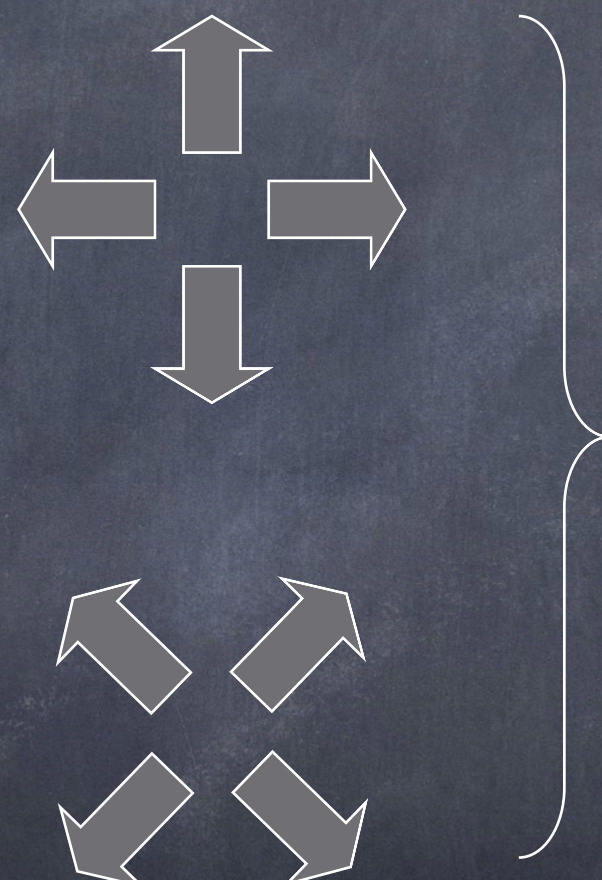
$$\nabla^2 f = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

	1	
1	-4	1
	1	

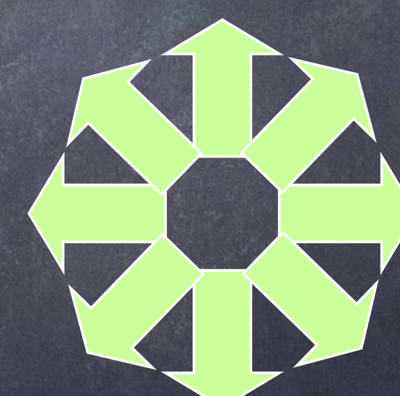
Laplacian

0	1	0
1	-4	1
0	1	0

1	0	1
0	-4	0
1	0	1

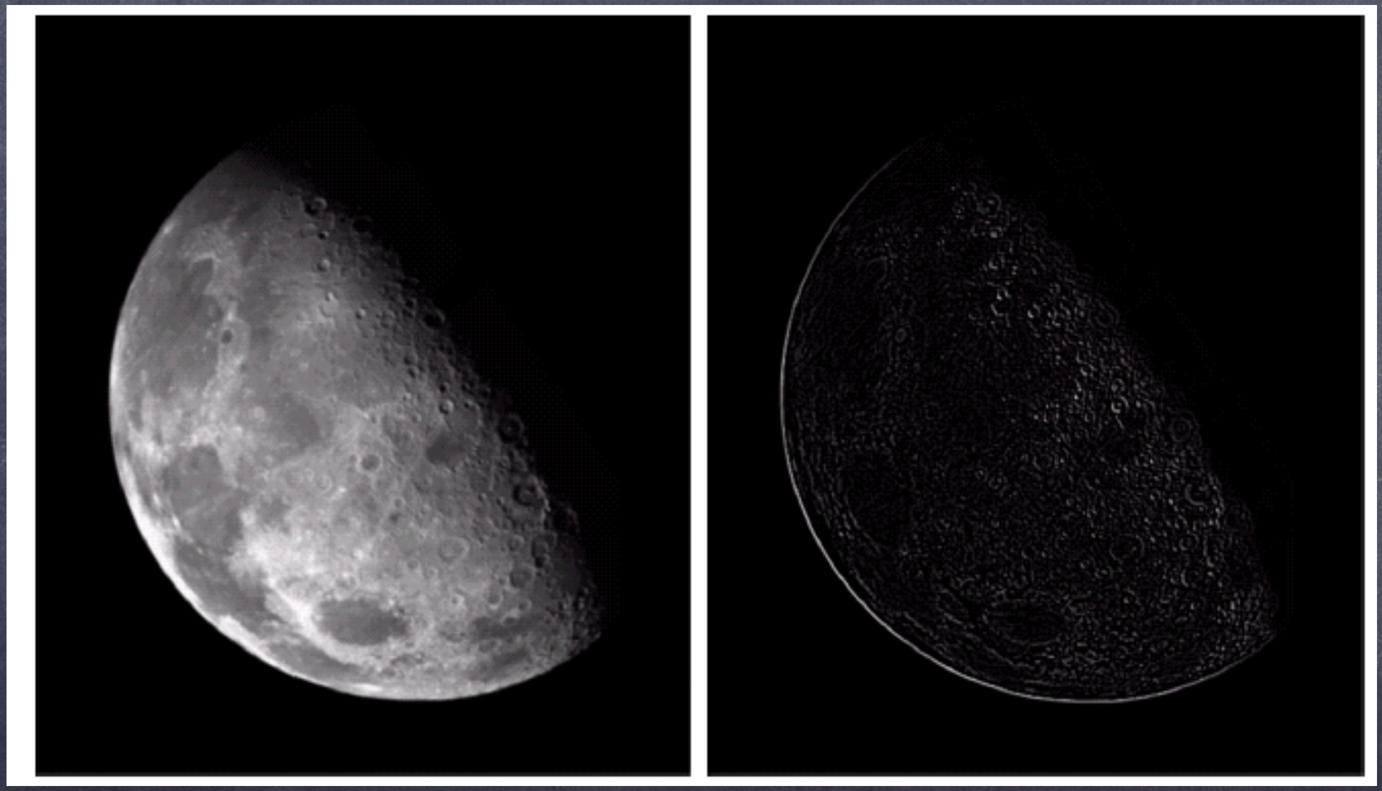


1	1	1
1	-8	1
1	1	1



Laplacian

- Applying the Laplacian to an image we get a new image that highlights edges and other discontinuities

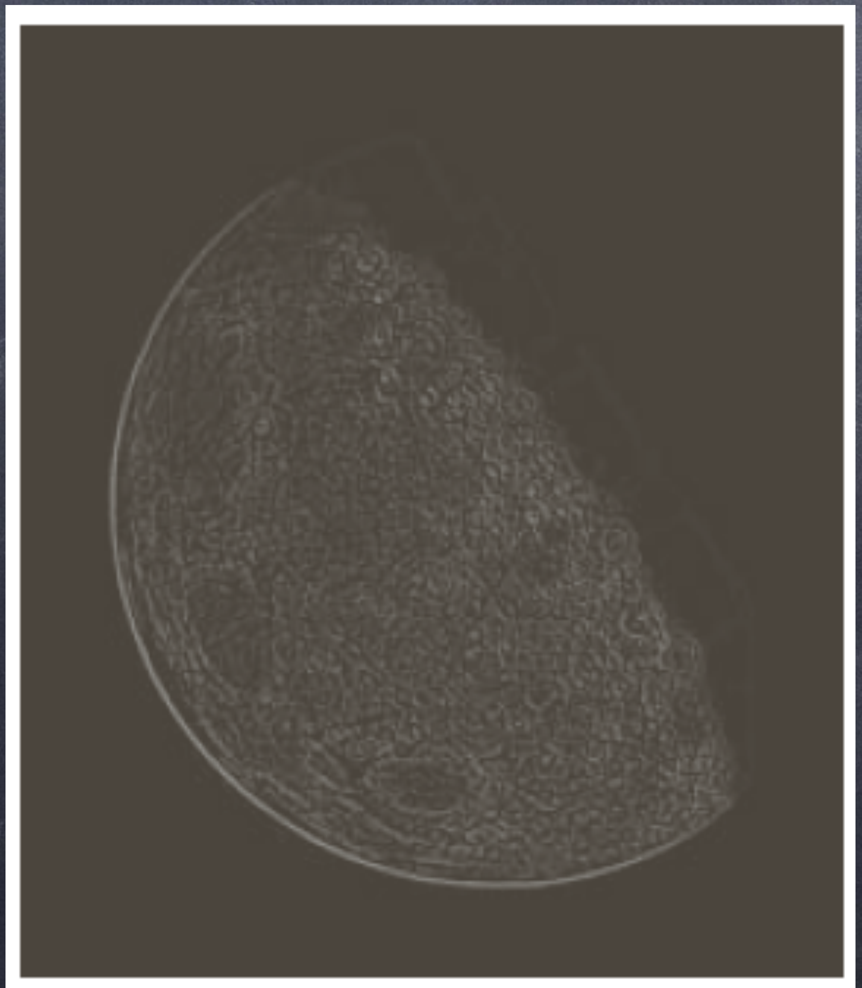


Original Image

Laplacian Filtered Image

Question

- If I have an image ... and I want to make it sharp and clear ... based on filtering technique, how can we deblur/sharp it?

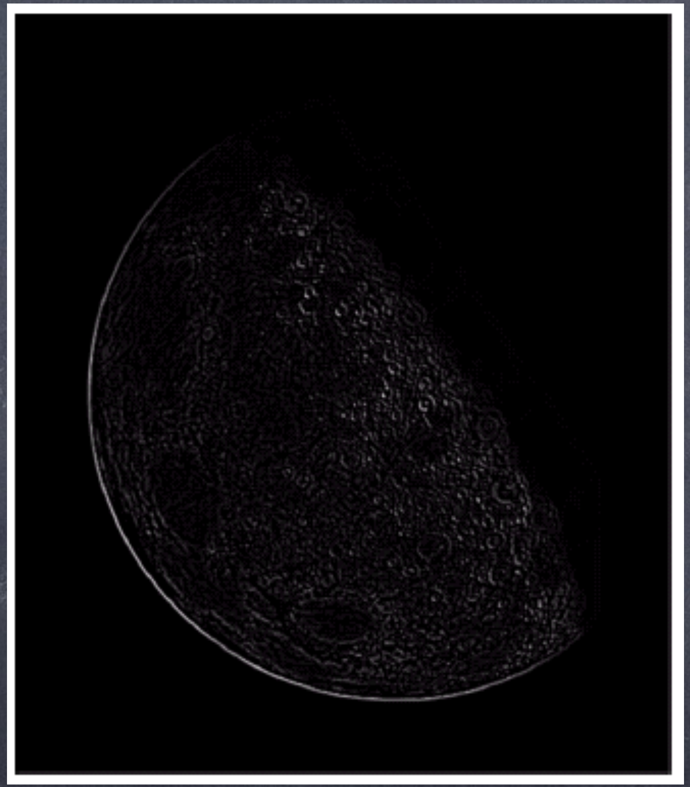


Laplacian Image Enhancement

- In the final sharpened image edges and fine detail are much more obvious

Original Image

-



Laplacian Filtered Image

=

Sharpened Image

$$g(x,y) = f(x,y) - \nabla^2 f$$

Laplacian Image Enhancement

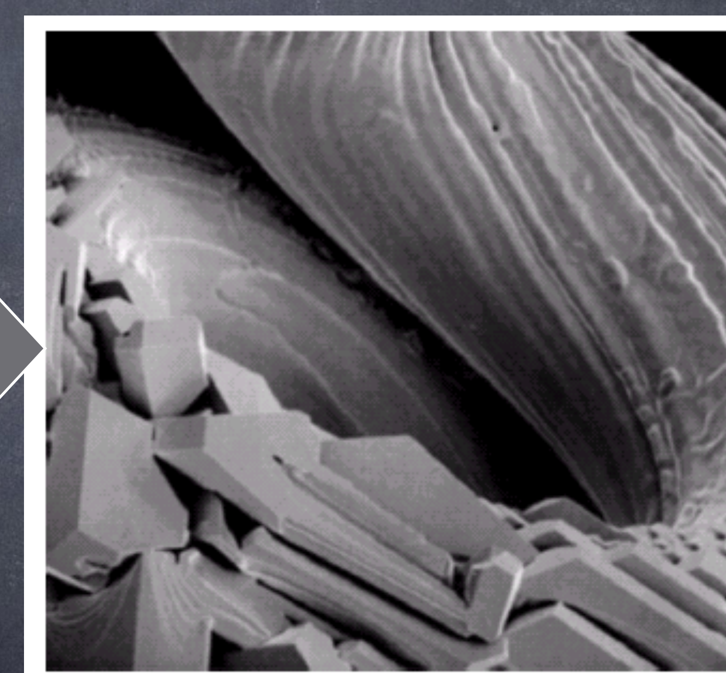
Simplified Image Enhancement

$$
\begin{aligned}
g(x,y) &= f(x,y) - \nabla^2 f \\
&= f(x,y) - [f(x+1,y) + f(x-1,y) \\
&\qquad + f(x,y+1) + f(x,y-1) \\
&\qquad - 4f(x,y)] \\
&= 5f(x,y) - f(x+1,y) - f(x-1,y) \\
&\qquad - f(x,y+1) - f(x,y-1)
\end{aligned}
$$

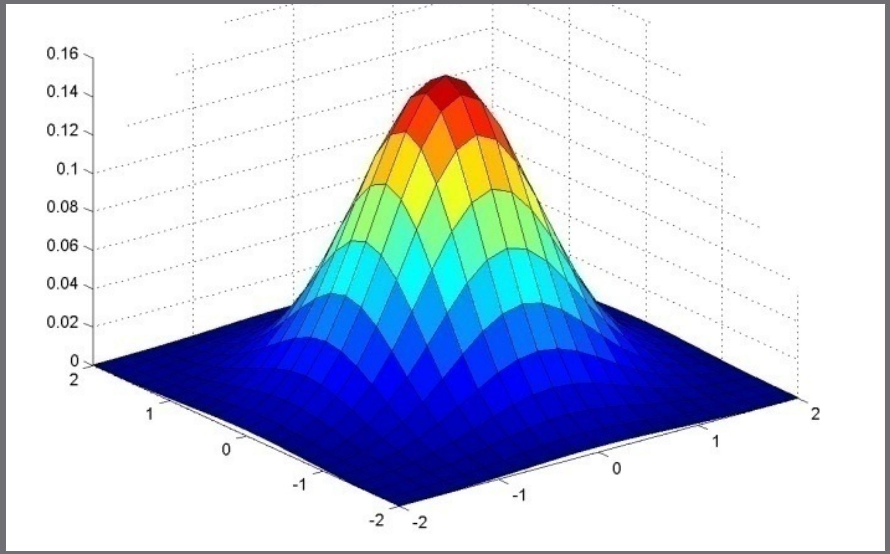
Simplified Image Enhancement

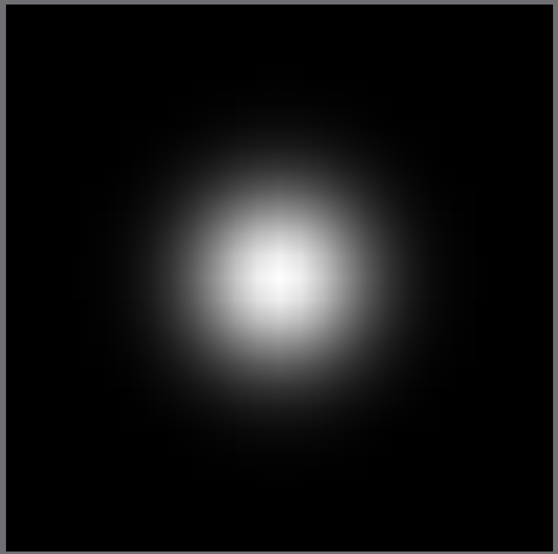
- This gives us a new filter which does the whole job for us in one step

0	-1	0
-1	5	-1
0	-1	0



Gaussian Filter





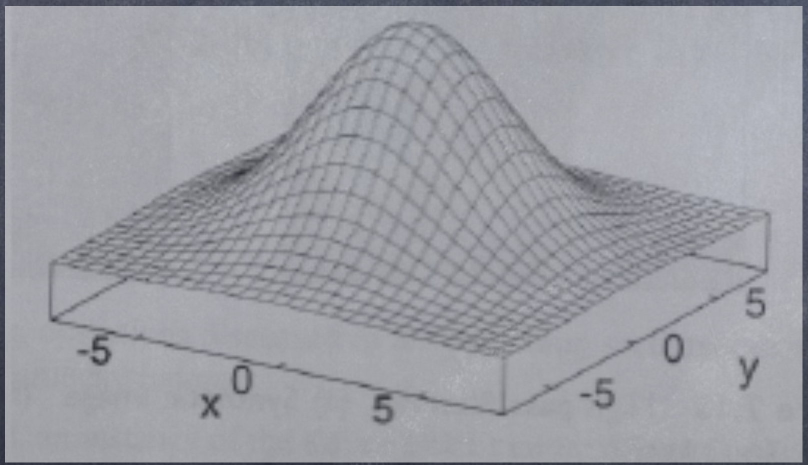
$$G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Source: C. Rasmussen

Gaussian Filter

- The weights are samples of the Gaussian function

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{x^2 + y^2}{2\sigma^2}}$$



7×7 Gaussian mask

1	1	2	2	2	1	1
1	2	2	4	2	2	1
2	2	4	8	4	2	2
2	4	8	16	8	4	2
2	2	4	8	4	2	2
1	2	2	4	2	2	1
1	1	2	2	2	1	1

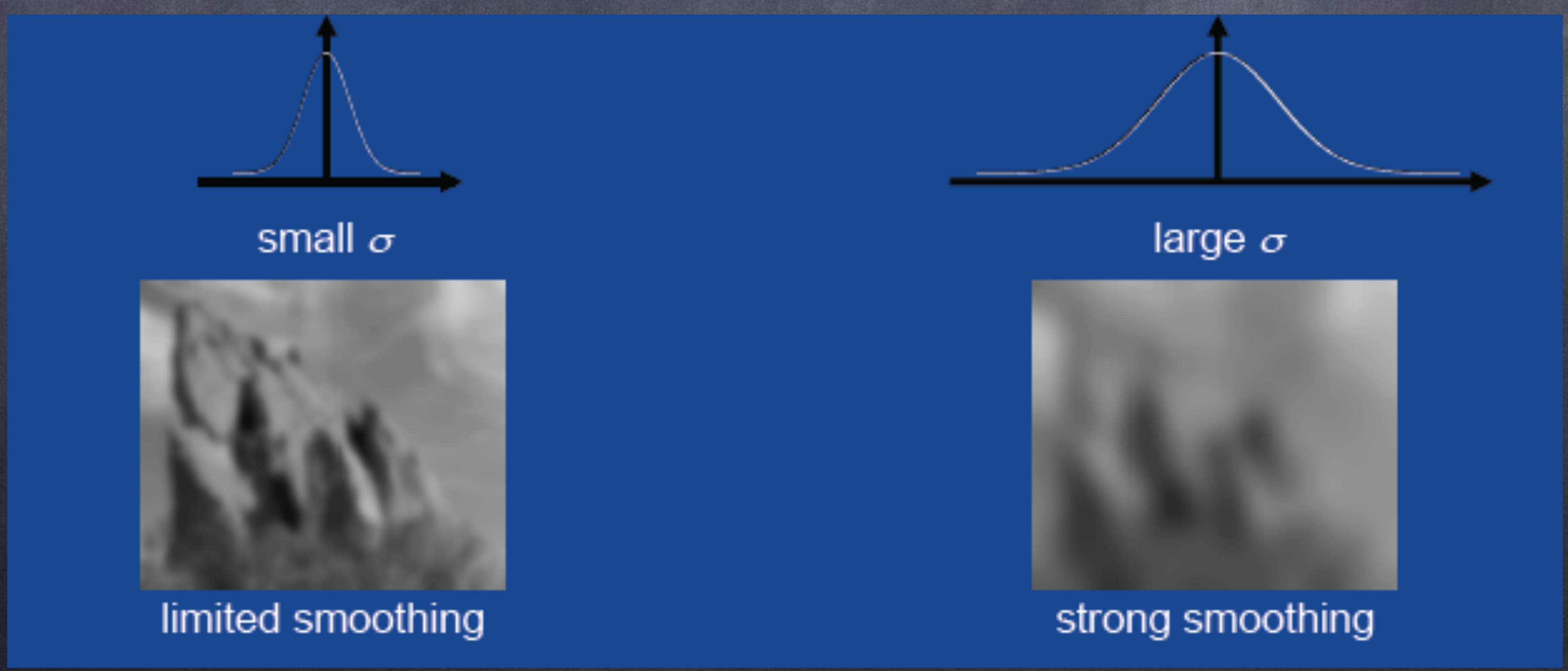
Gaussian Filter

- As **σ** increases, more samples must be obtained to represent the Gaussian function accurately.
- Therefore, **σ** controls the amount of smoothing

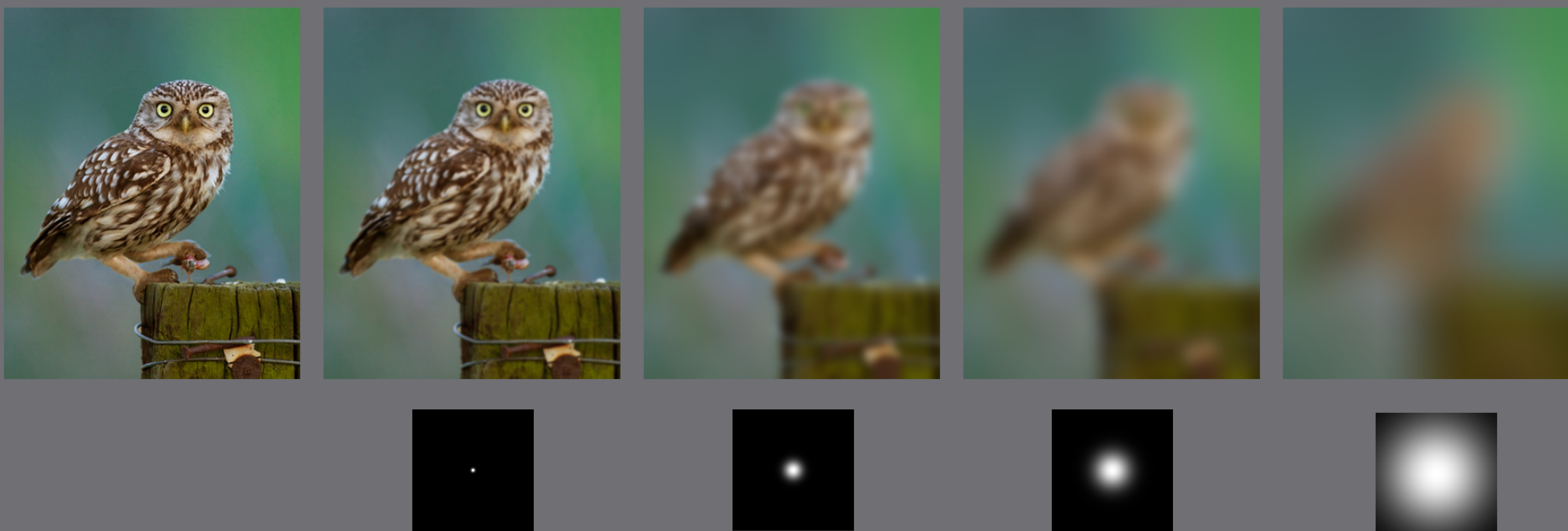
15 × 15 Gaussian mask

2	2	3	4	5	5	6	6	6	5	5	4	3	2	2
2	3	4	5	7	7	8	8	8	7	7	5	4	3	2
3	4	6	7	9	10	10	11	10	10	9	7	6	4	3
4	5	7	9	10	12	13	13	13	12	10	9	7	5	4
5	7	9	11	13	14	15	16	15	14	13	11	9	7	5
5	7	10	12	14	16	17	18	17	16	14	12	10	7	5
6	8	10	13	15	17	19	19	19	17	15	13	10	8	6
6	8	11	13	16	18	19	20	19	18	16	13	11	8	6
6	8	10	13	15	17	19	19	19	17	15	13	10	8	6
5	7	10	12	14	16	17	18	17	16	14	12	10	7	5
5	7	9	11	13	14	15	16	15	14	13	11	9	7	5
4	5	7	9	10	12	13	13	13	12	10	9	7	5	4
3	4	6	7	9	10	10	11	10	10	9	7	6	4	3
2	3	4	5	7	7	8	8	8	7	7	5	4	3	2
2	2	3	4	5	5	6	6	6	5	5	4	3	2	2

Gaussian Filter



Gaussian filters



σ = 1 pixel σ = 5 pixels σ = 10 pixels σ = 30 pixels

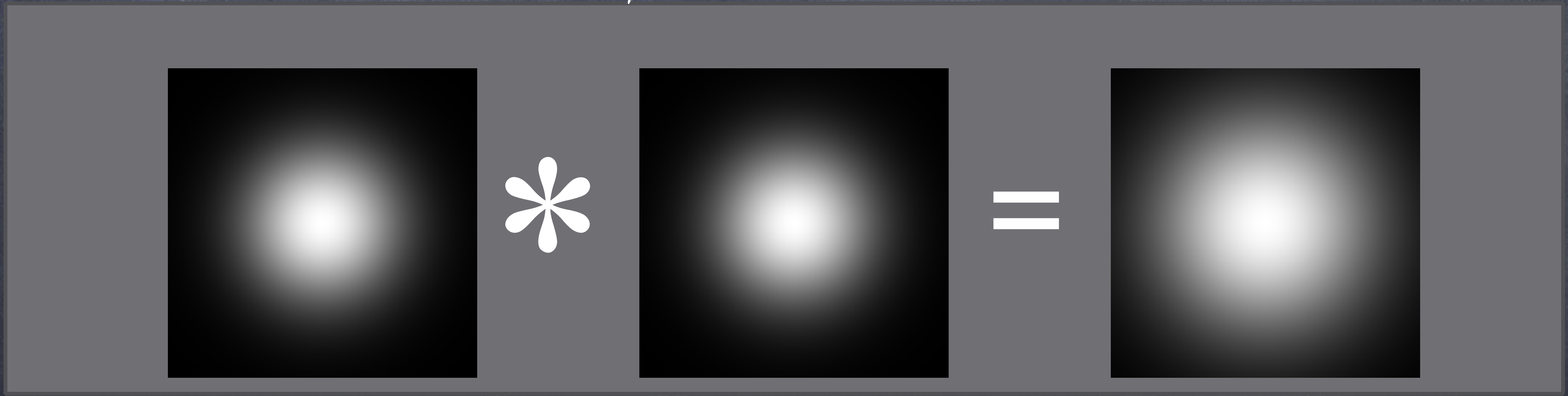
Averaging vs Gaussian Smoothing

Averaging

Gaussian

Gaussian filter

- Removes "high-frequency" components from the image (low-pass filter)
- Convolution with self is another Gaussian



- Convolving two times with Gaussian kernel of width σ = convolving once with kernel of width $\sigma\sqrt{2}$

Source: K. Grauman

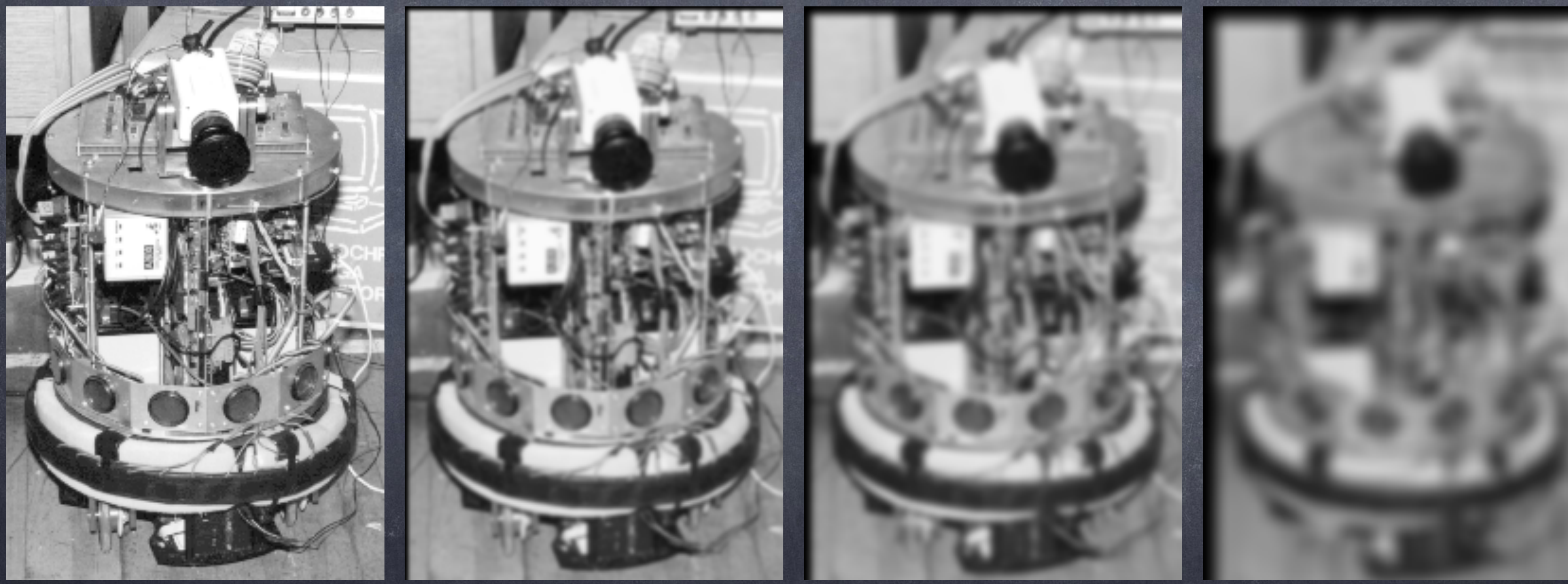
More on Gaussian Filters

- What are two parameters in Gaussian Filter?

$\frac{1}{273}$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

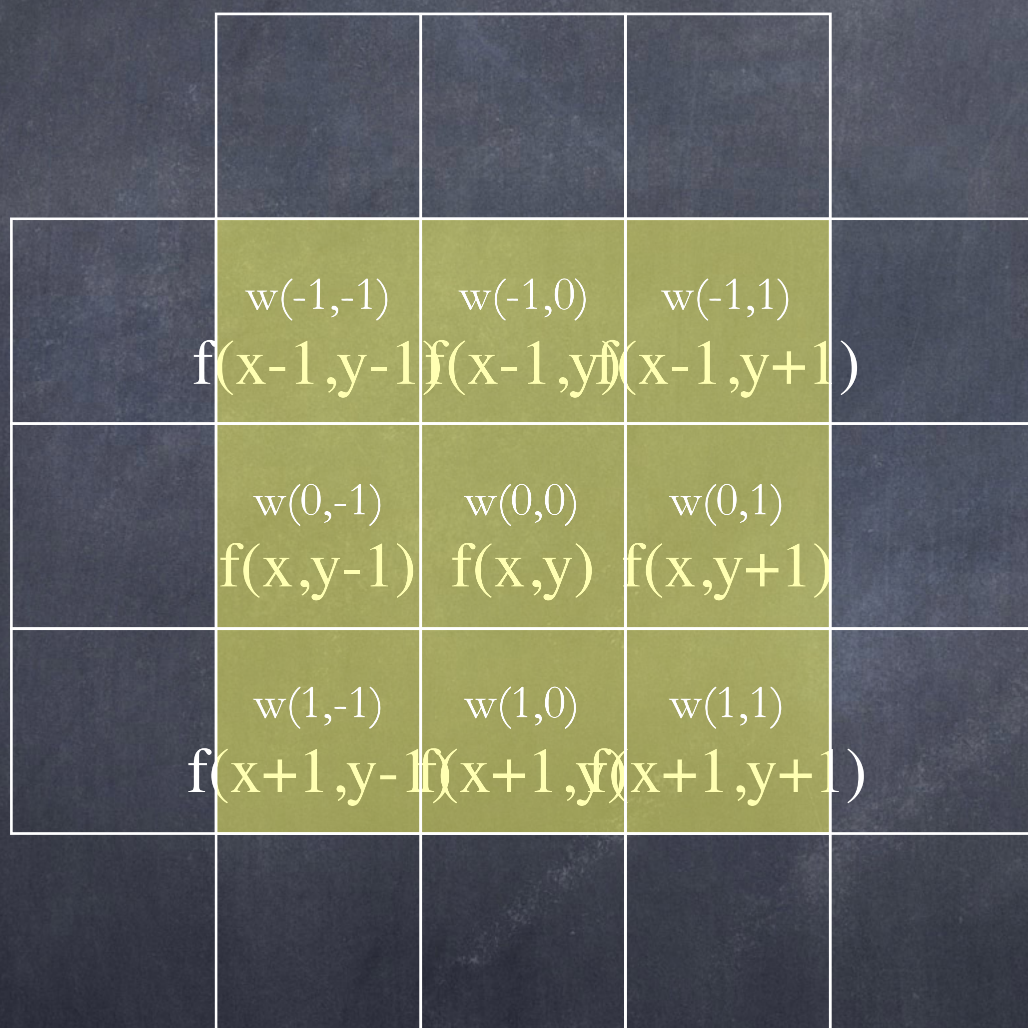
What happens with varying kernel Size and Gaussian parameter



Filtering Revisited

Linear spatial filtering

Pixels of image



The result is the sum of products of the mask coefficients with the corresponding pixels directly under the mask

Mask coefficients

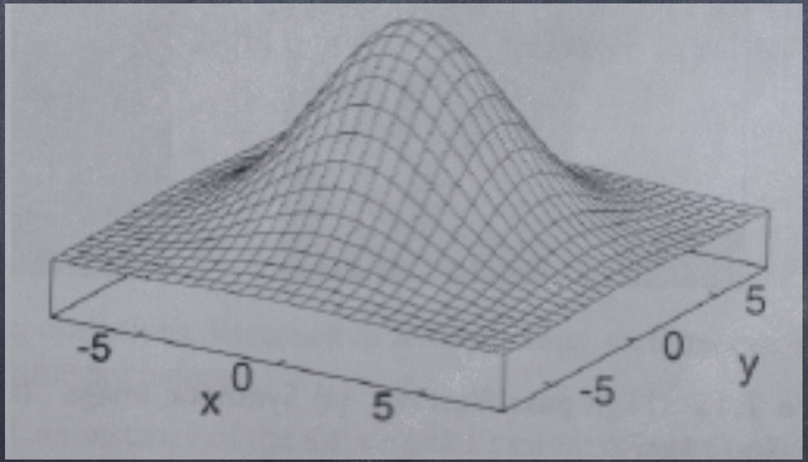
w(-1,-1)	w(-1,0)	w(-1,1)
w(0,-1)	w(0,0)	w(0,1)

$$
\begin{aligned}
f(x,y) = {} & w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + w(-1,1)f(x-1,y+1) + \\
& w(0,-1)f(x,y-1) + w(0,0)f(x,y) + w(0,1)f(x,y+1) + \\
& w(1,-1)f(x+1,y-1) + w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1)
\end{aligned}
$$

Gaussian Filter

- The weights are samples of the Gaussian function

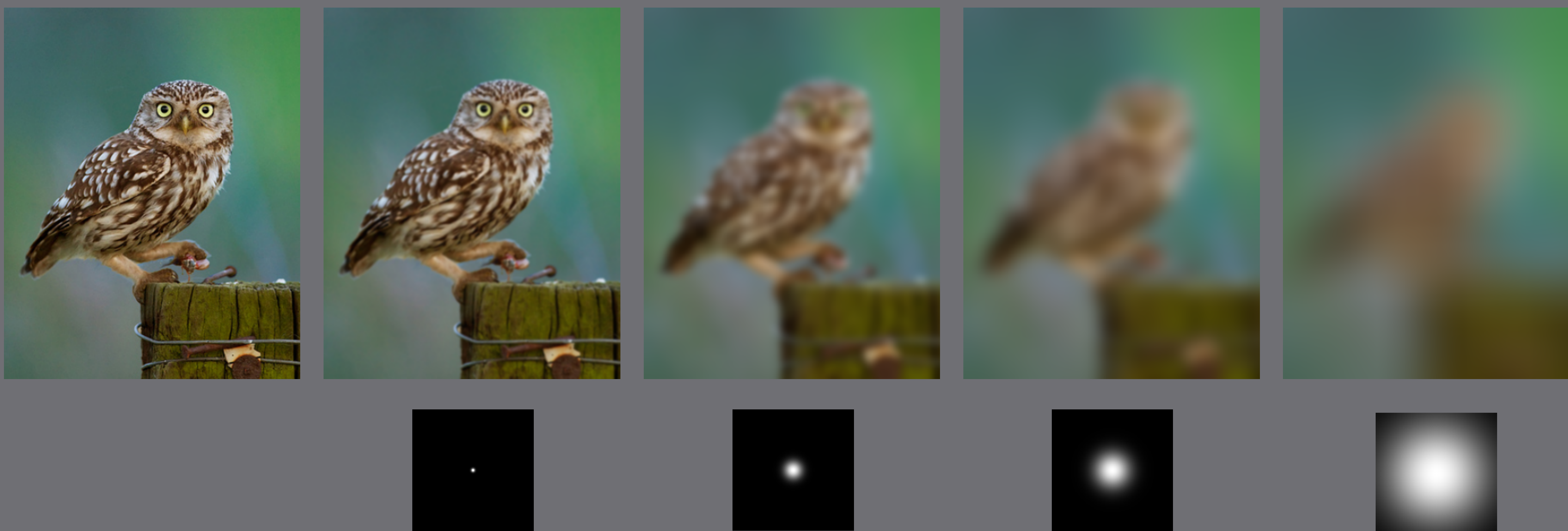
$$G_\sigma(x,y) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{x^2+y^2}{2\sigma^2}}$$



7 × 7 Gaussian mask

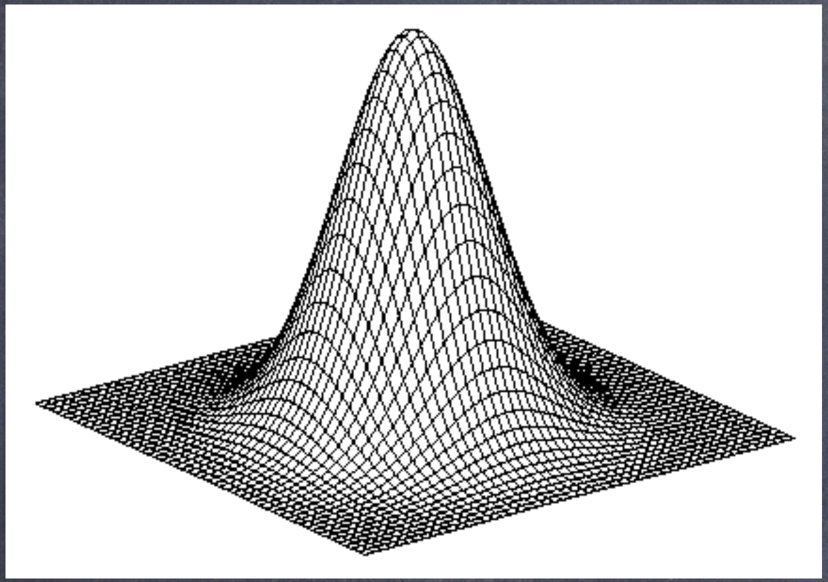
1	1	2	2	2	1	1
1	2	2	4	2	2	1
2	2	4	8	4	2	2
2	4	8	16	8	4	2
2	2	4	8	4	2	2
1	2	2	4	2	2	1
1	1	2	2	2	1	1

Gaussian filters



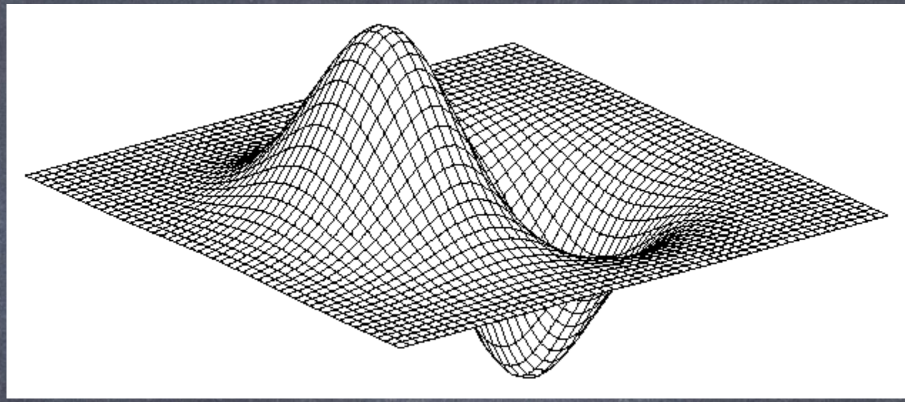
σ = 1 pixel σ = 5 pixels σ = 10 pixels σ = 30 pixels

2D edge detection filters



Gaussian

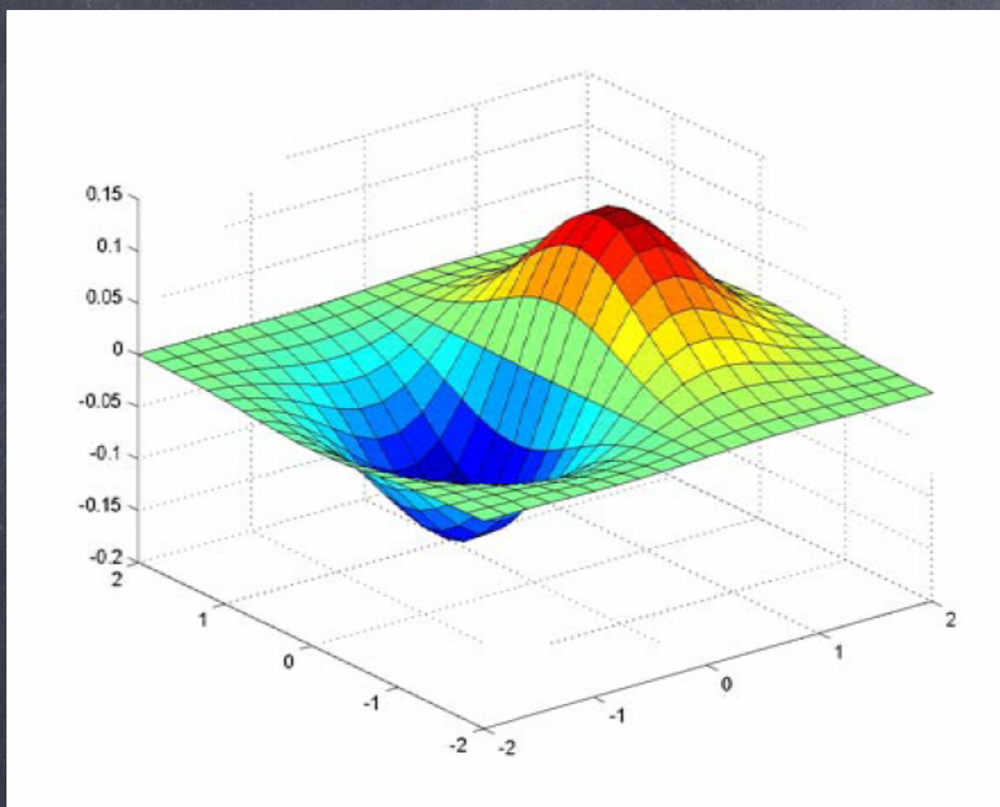
$$h_\sigma(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$



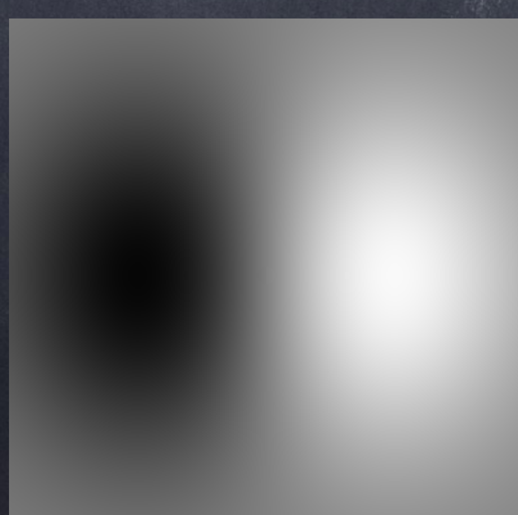
derivative of Gaussian (x)

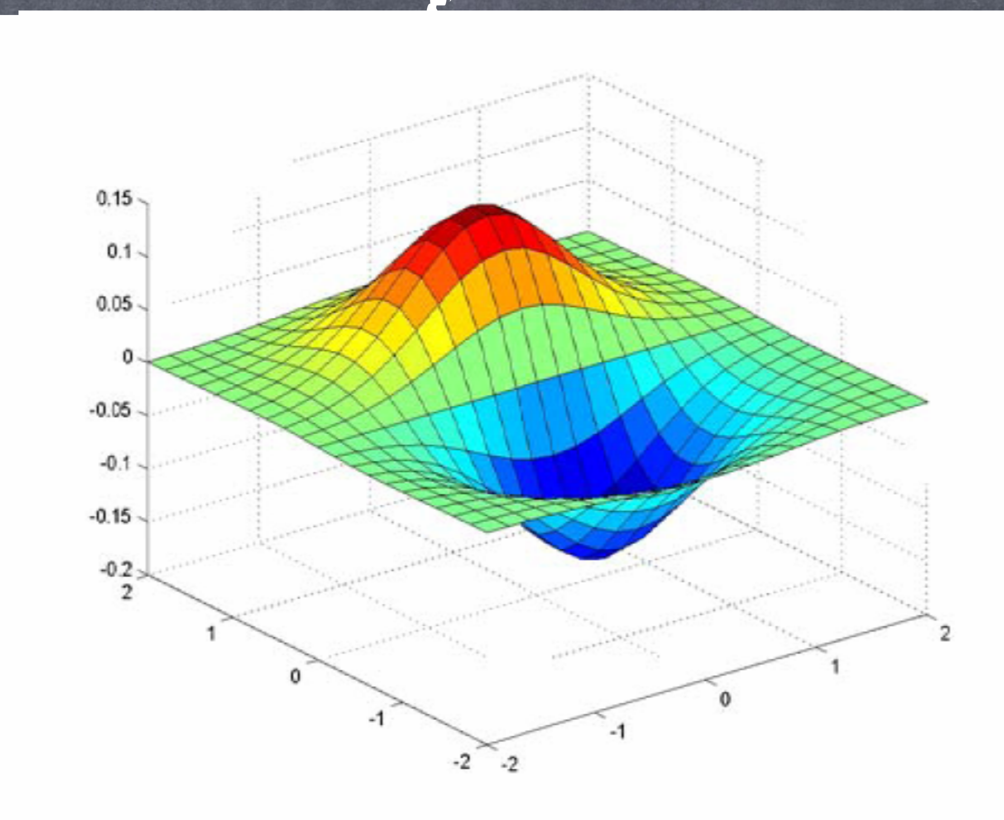
$$\frac{\partial}{\partial x} h_\sigma(u,v)$$

Derivative of Gaussian filter

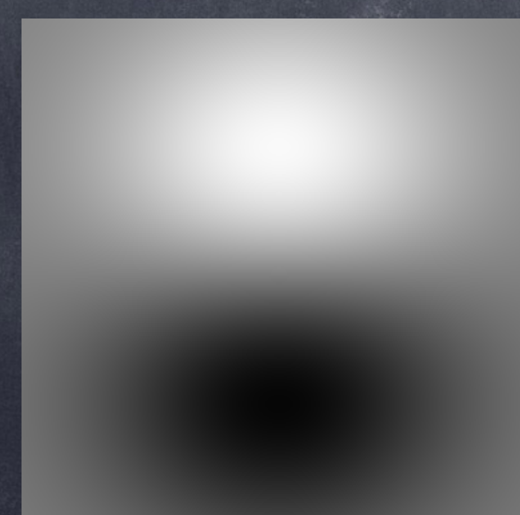


x-direction





y-direction



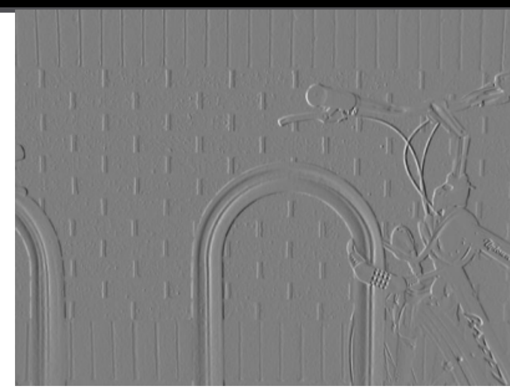
THE SOBEL OPERATOR

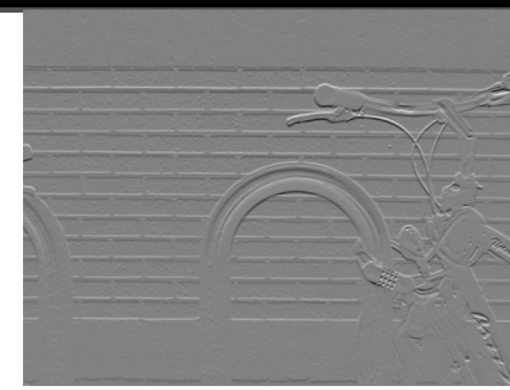
Common approximation of derivative of Gaussian

$$\frac{1}{8}\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad \frac{1}{8}\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

s_x $\qquad$ s_y

SOBEL OPERATOR: EXAMPLE





Source: Wikipedia

EXAMPLE

original image (Lena)

FINDING EDGES

gradient magnitude

Finding edges

Thresholding and thinning

CANNY EDGE DETECTOR

1. Filter image with derivative of Gaussian
2. Find magnitude and orientation of gradient
3. Non-maximum suppression
4. Linking and thresholding

Source: D. Lowe, L. Fei-Fei

CANNY EDGE DETECTOR

Still one of the most widely used edge detectors in computer vision

J. Canny, *A Computational Approach To Edge Detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986.

Depends on several parameters:

σ: width of the Gaussian blur

threshold

CANNY EDGE DETECTOR

original | Canny with $\sigma = 1$ | Canny with $\sigma = 2$

- The choice of σ depends on desired behavior
  - large σ detects "large-scale" edges
  - small σ detects fine edges

Source: S. Seitz

COMBINING SPATIAL ENHANCEMENT METHODS

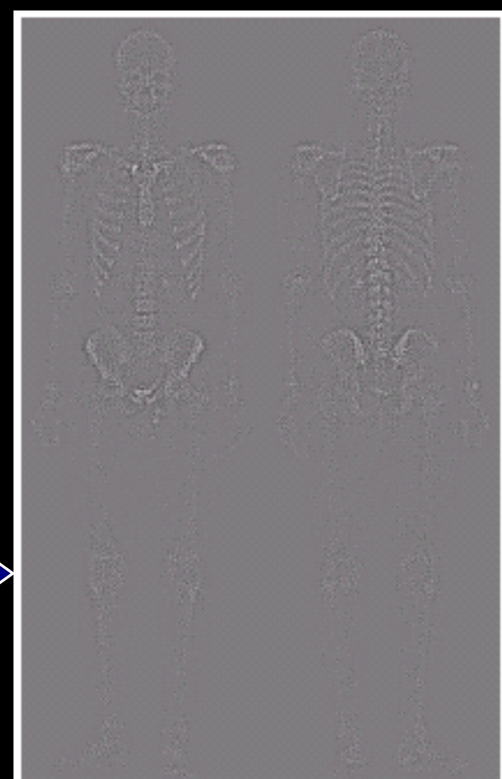
Successful image enhancement is typically not achieved using a single operation

Rather we combine a range of techniques in order to achieve a final result

COMBINING SPATIAL ENHANCEMENT METHODS

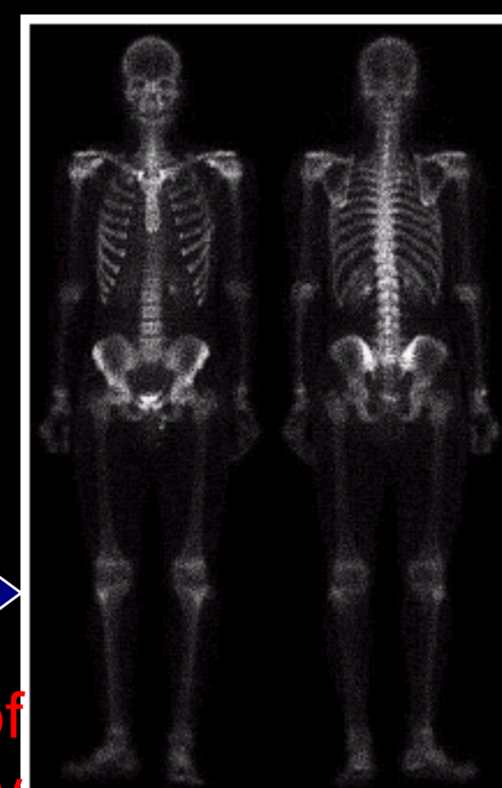
(a)

Laplacian filter of bone scan (a)



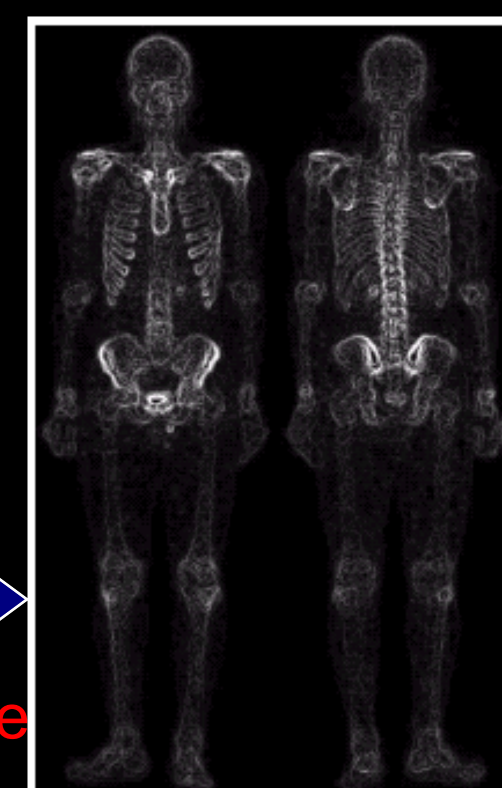
(b)

Sharpened version of bone scan achieved by subtracting (a) and (b)



(c)

Sobel filter of bone scan (a)



(d)

COMBINING SPATIAL ENHANCEMENT METHODS

(e)

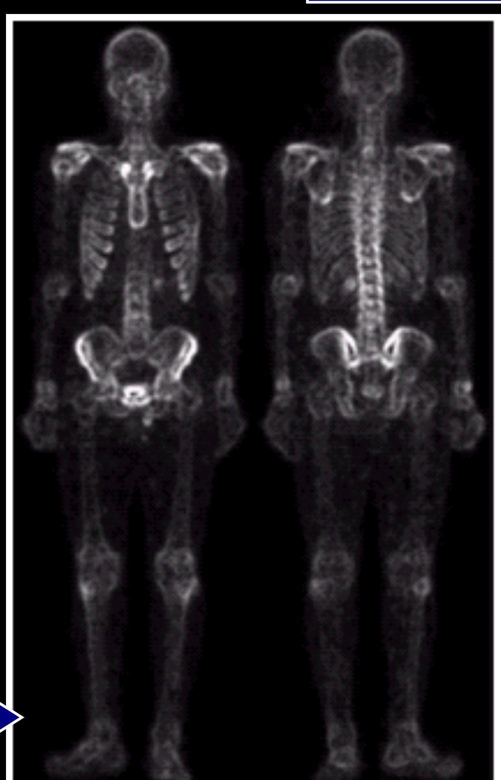
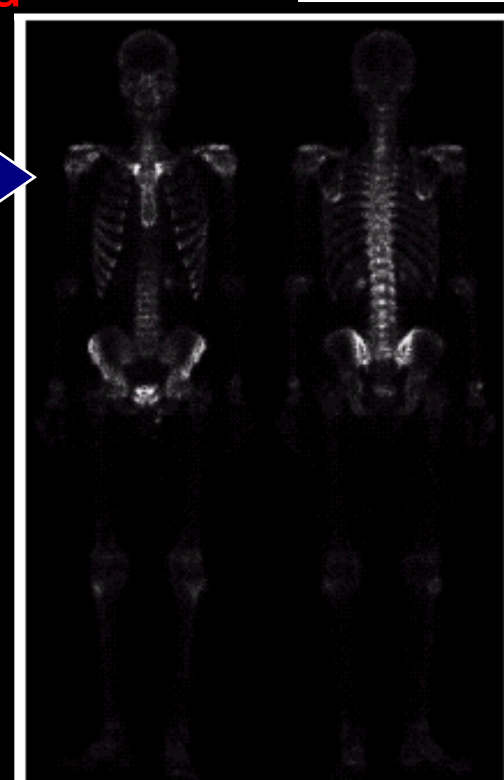


Image (d) smoothed with a 5*5 averaging filter

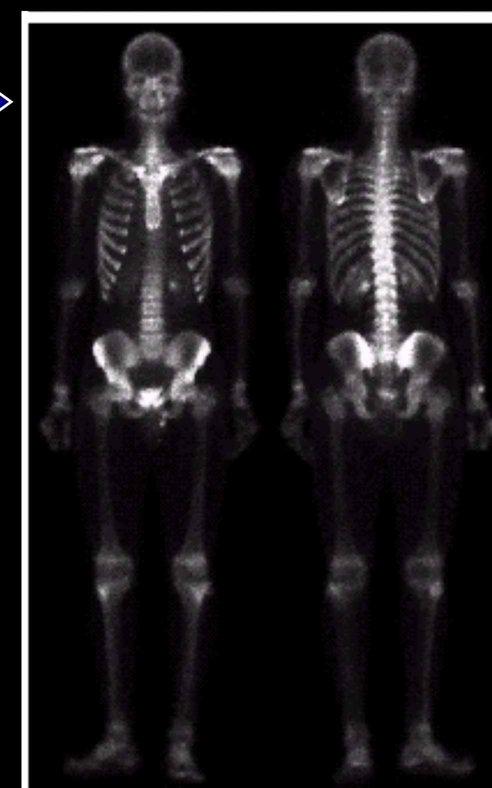
The product of (c) and (e) which will be used as a mask

(f)



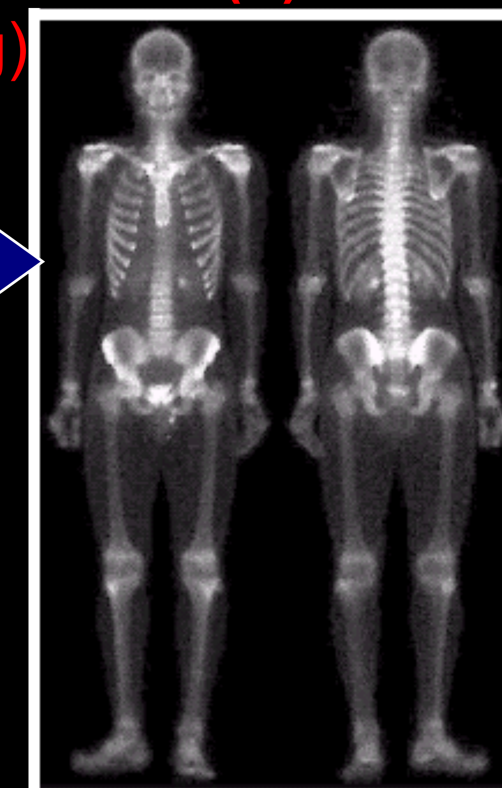
Sharpened image which is sum of (a) and (f)

(g)



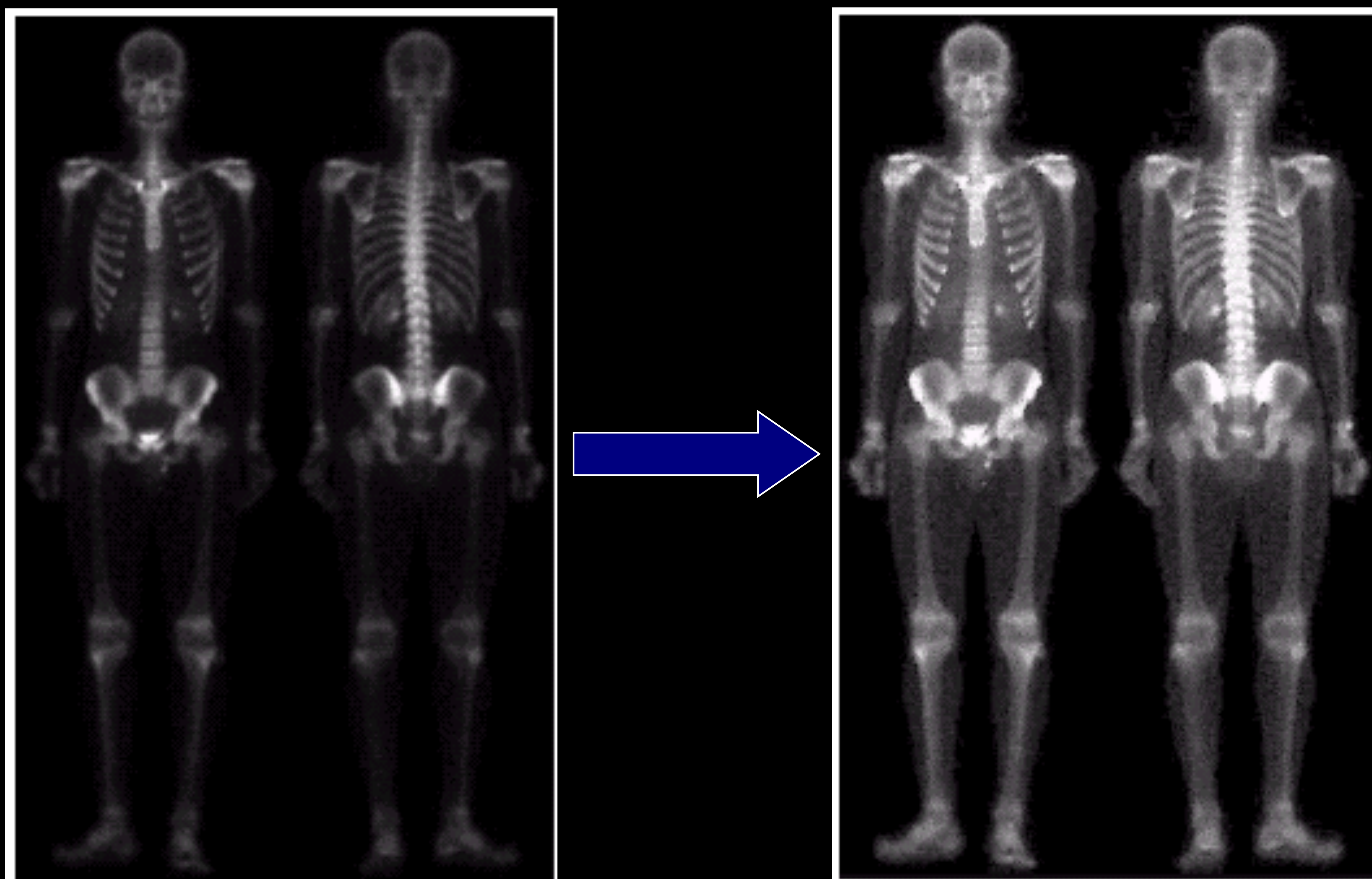
Result of applying a power-law trans. to (g)

(h)



COMBINING SPATIAL ENHANCEMENT METHODS

Compare the original and final images



OTHER SPATIAL FILTERS

- Read from the book (will not cover all in lectures but may come in the exam

Image Pyramids

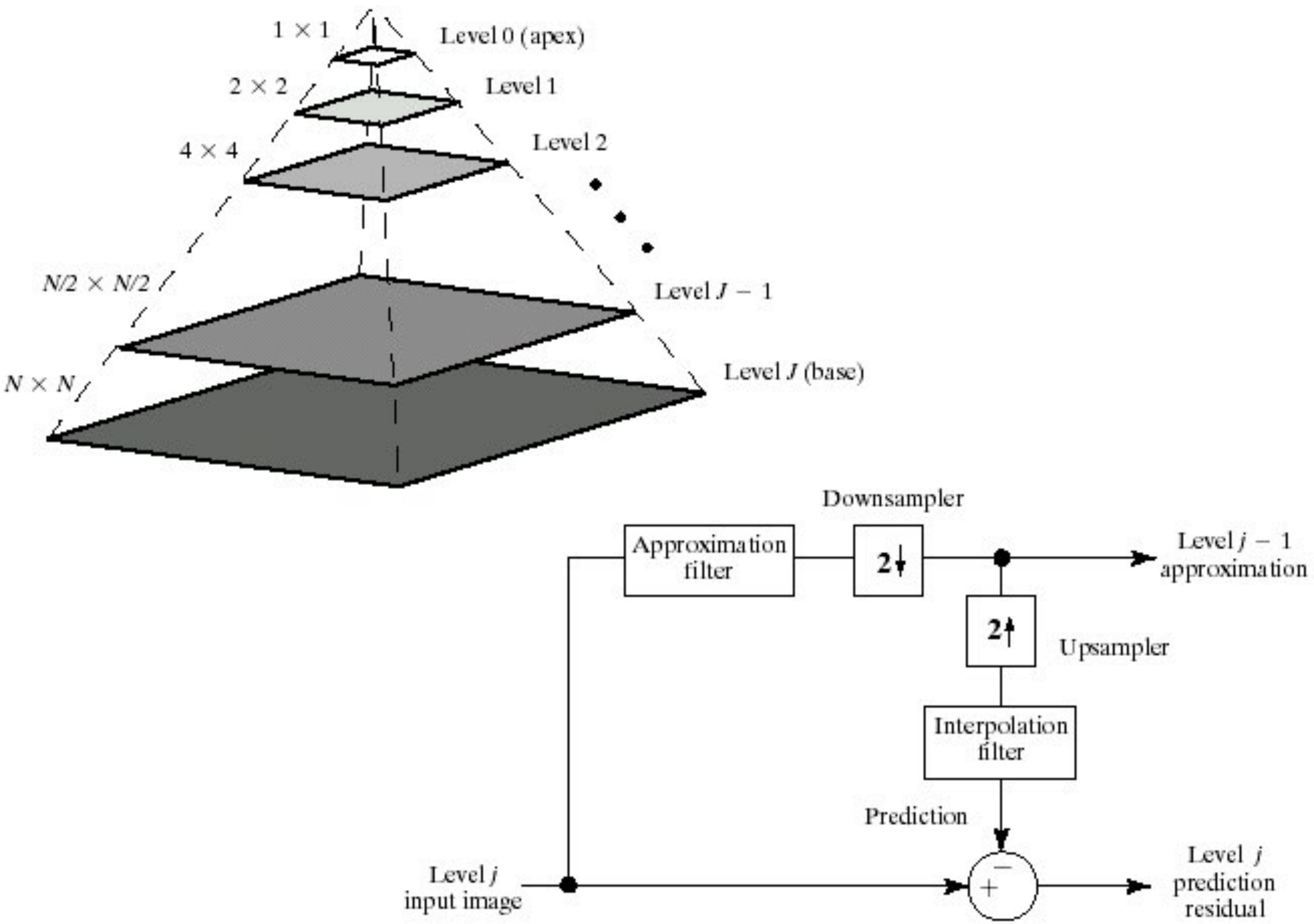
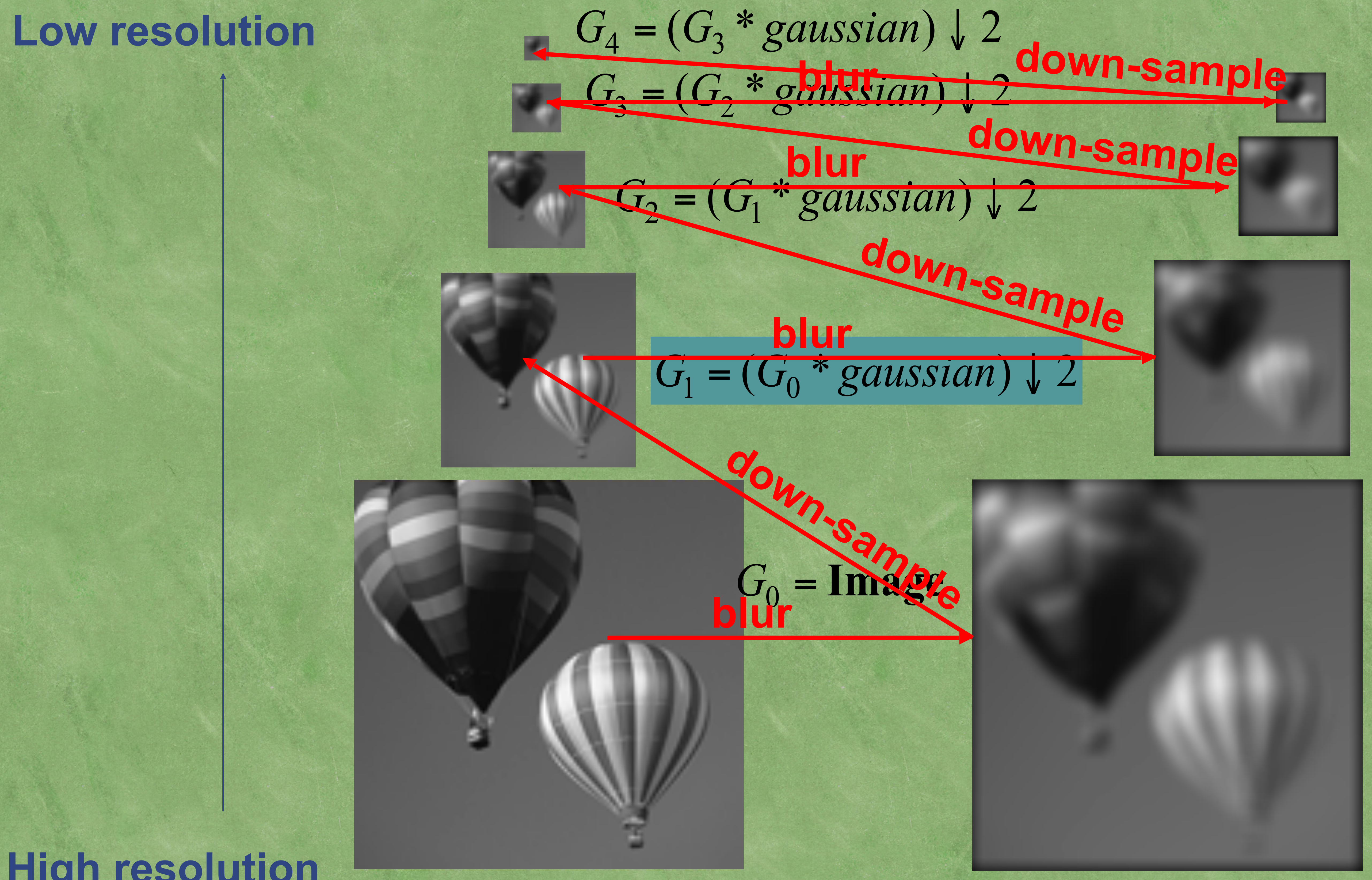


FIGURE 7.2 (a) A pyramidal image structure and (b) system block diagram for creating it.

The Gaussian Pyramid



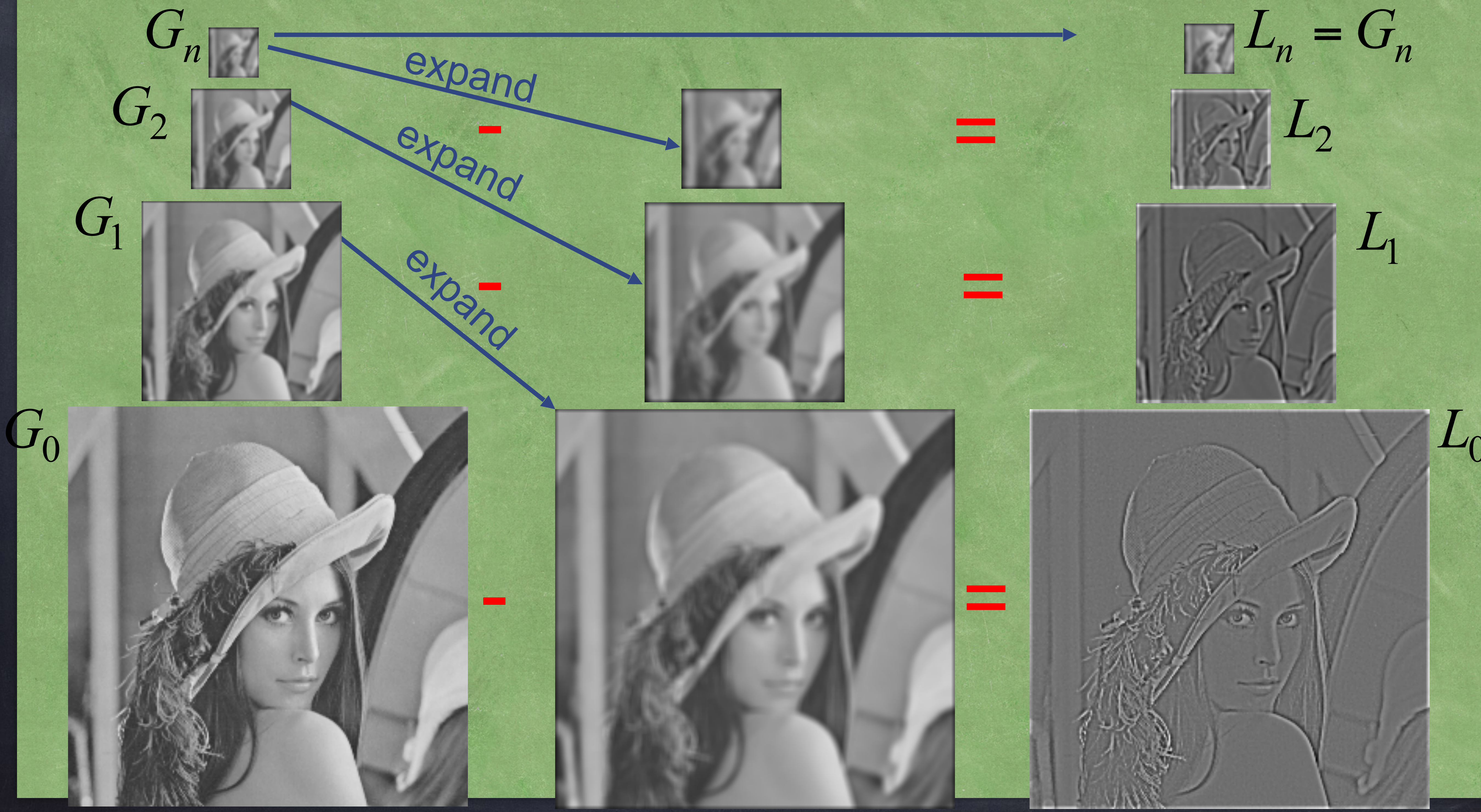
The Laplacian Pyramid

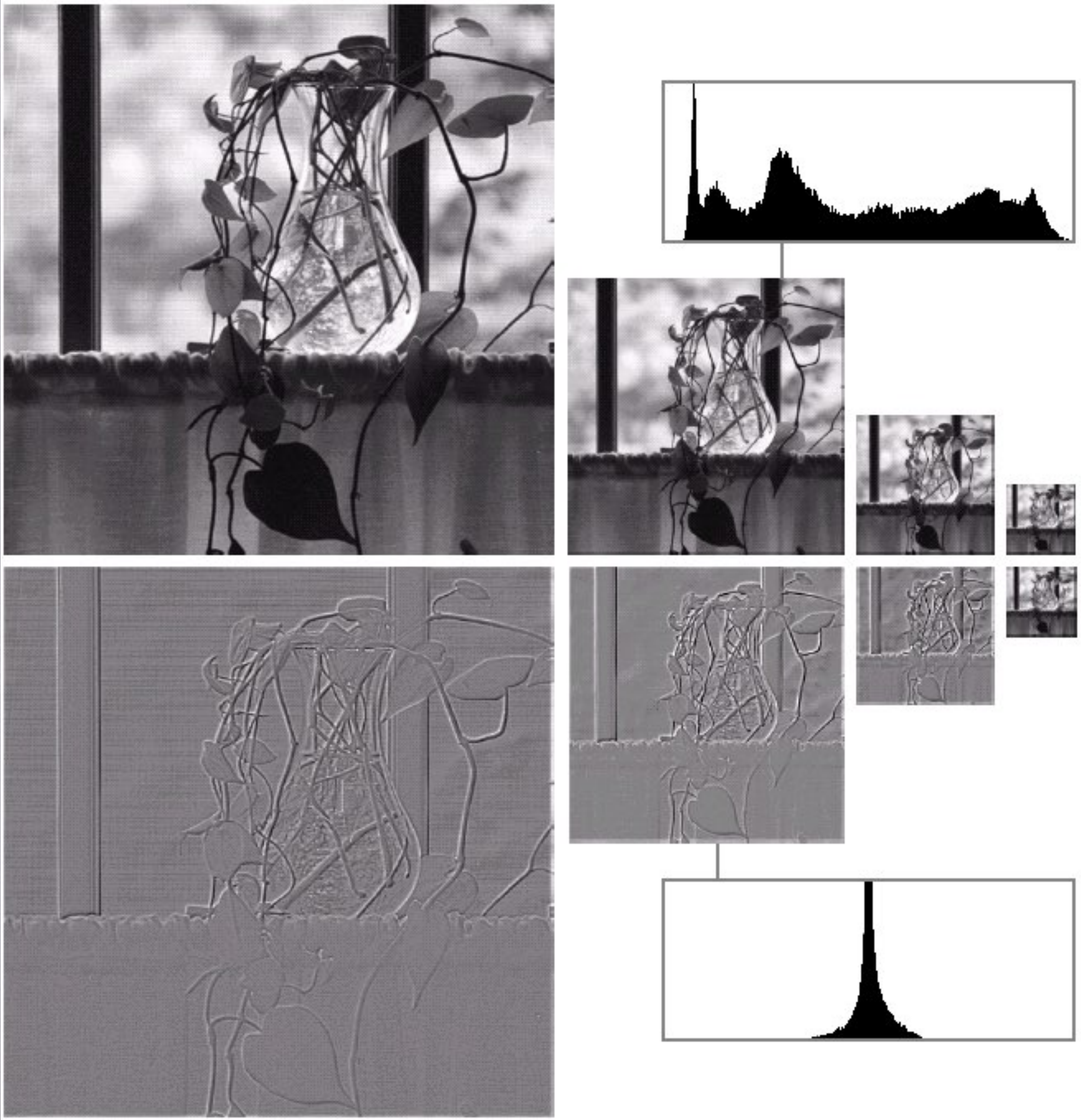
$$L_i = G_i - \text{expand}(G_{i+1})$$

$$G_i = L_i + \text{expand}(G_{i+1})$$

Gaussian Pyramid

Laplacian Pyramid





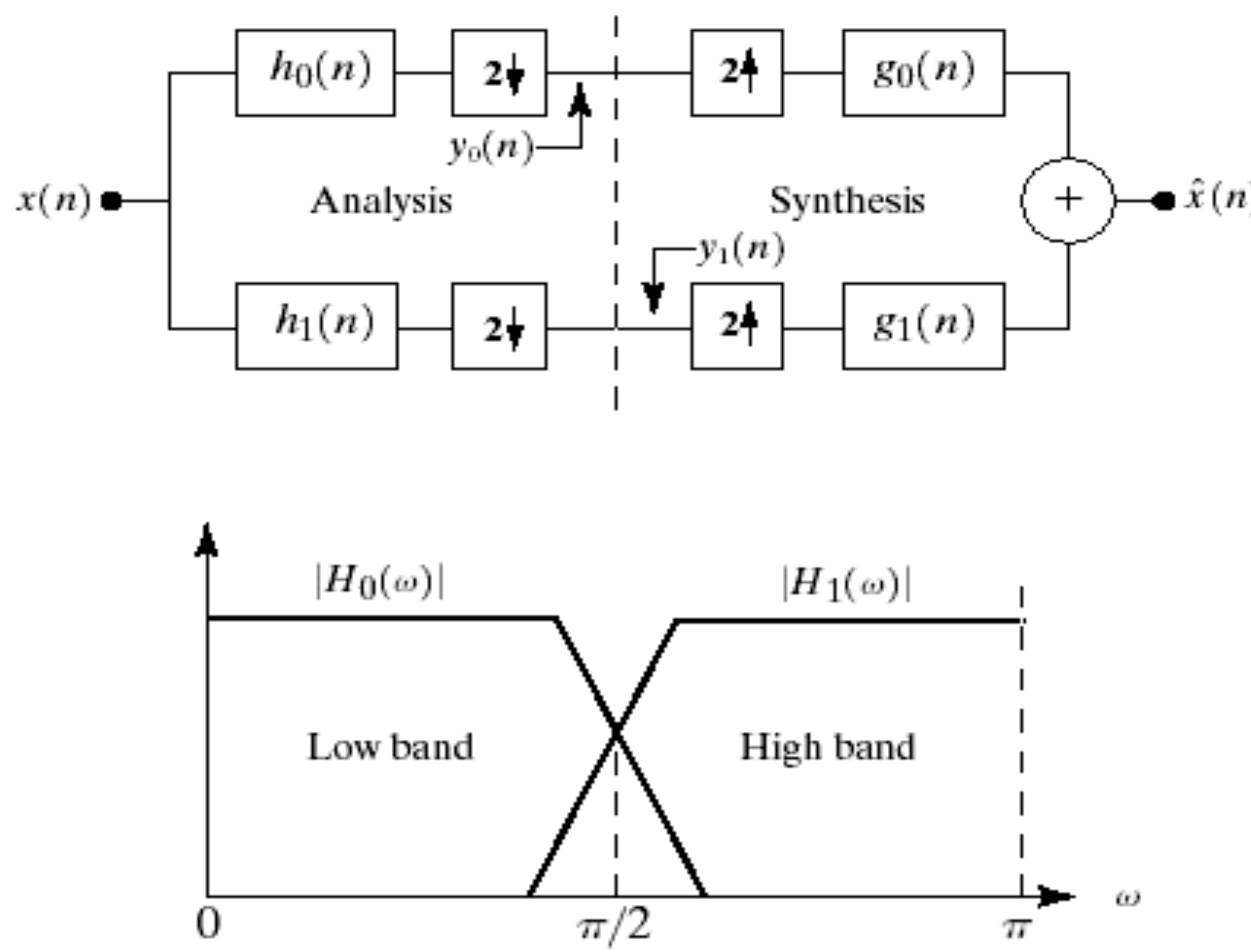
a
b

FIGURE 7.3 Two image pyramids and their statistics: (a) a Gaussian (approximation) pyramid and (b) a Laplacian (prediction residual) pyramid.

From Image Pyramids to Wavelet Transforms

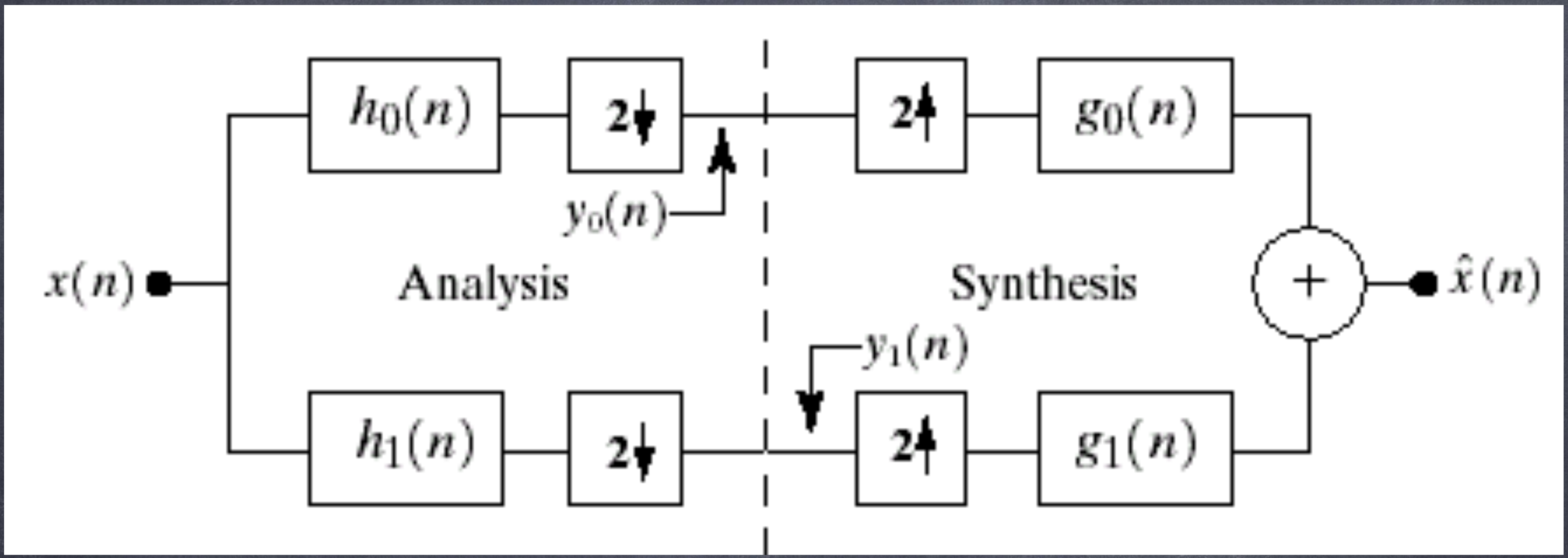
Subband coding

- In subband coding, an image is decomposed into a set of bandlimited components, called subbands.
- Since the bandwidth of the resulting subbands is smaller than that of the original image, the subbands can be downsampled without loss of information.



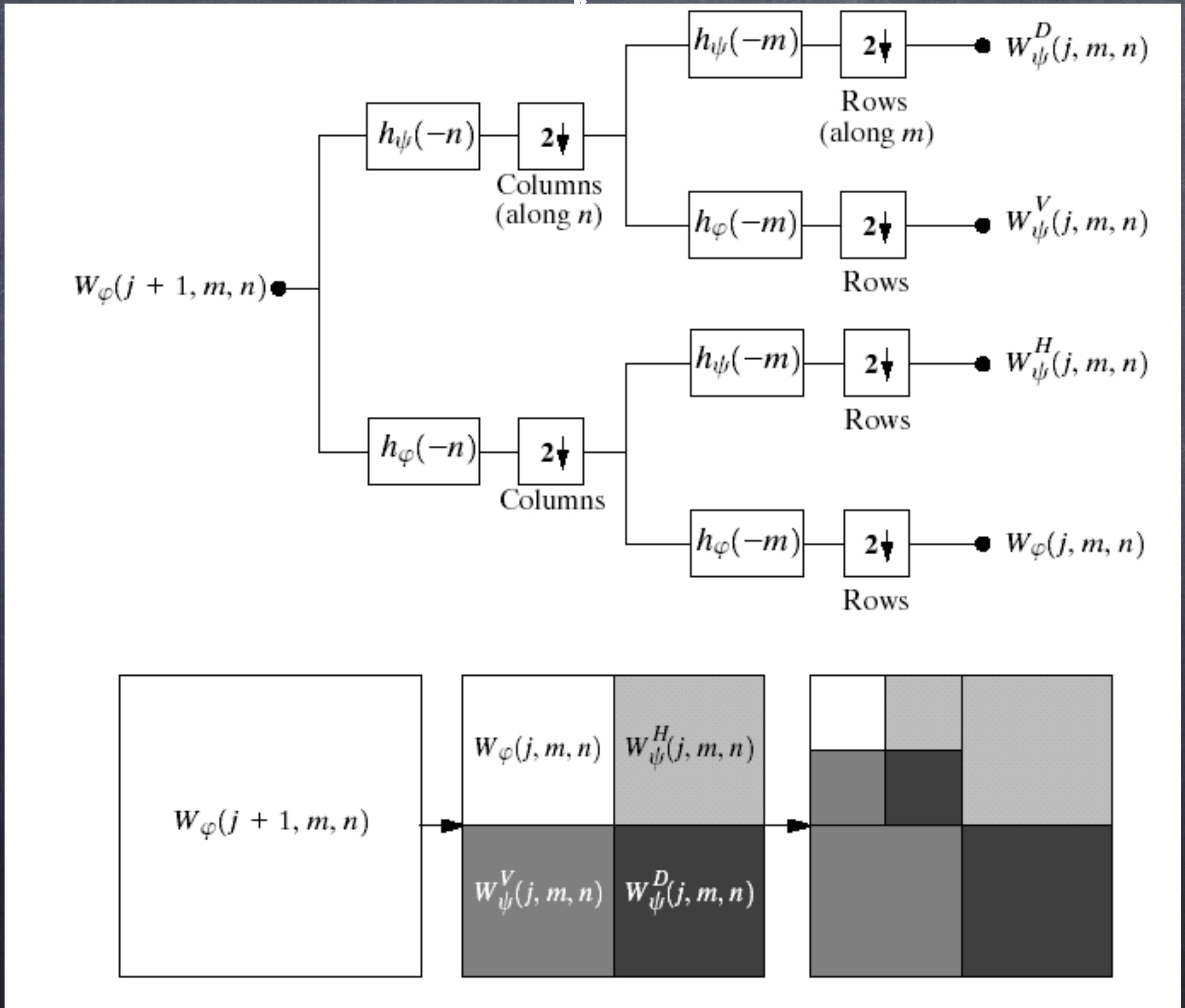
a
b
FIGURE 7.4 (a) A two-band filter bank for one-dimensional subband coding and decoding, and (b) its spectrum splitting properties.

Perfect Reconstruction Filter

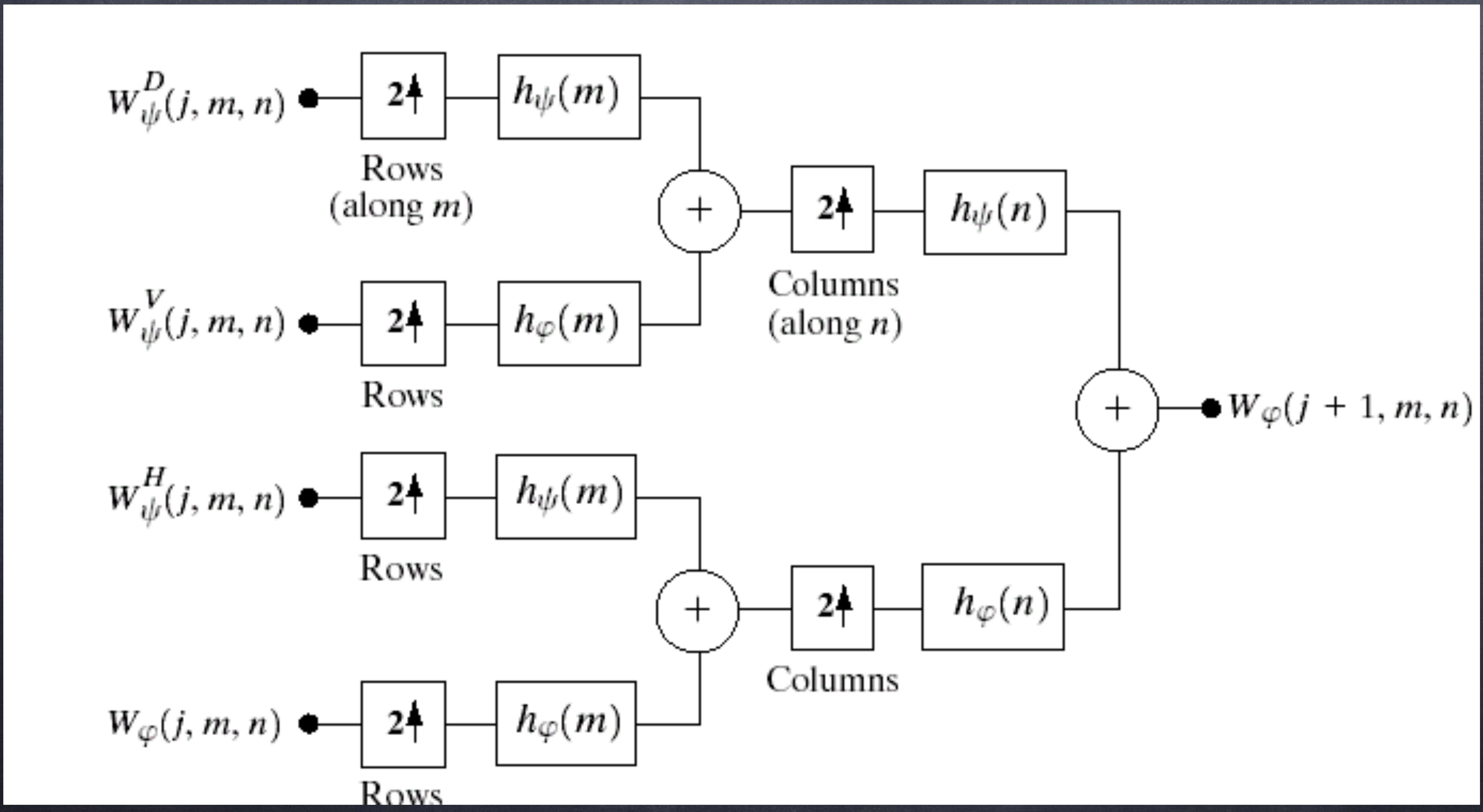


Goal: find filters so that $x(n) = \hat{x}(n)$

2-D Wavelet Transform: Decomposition

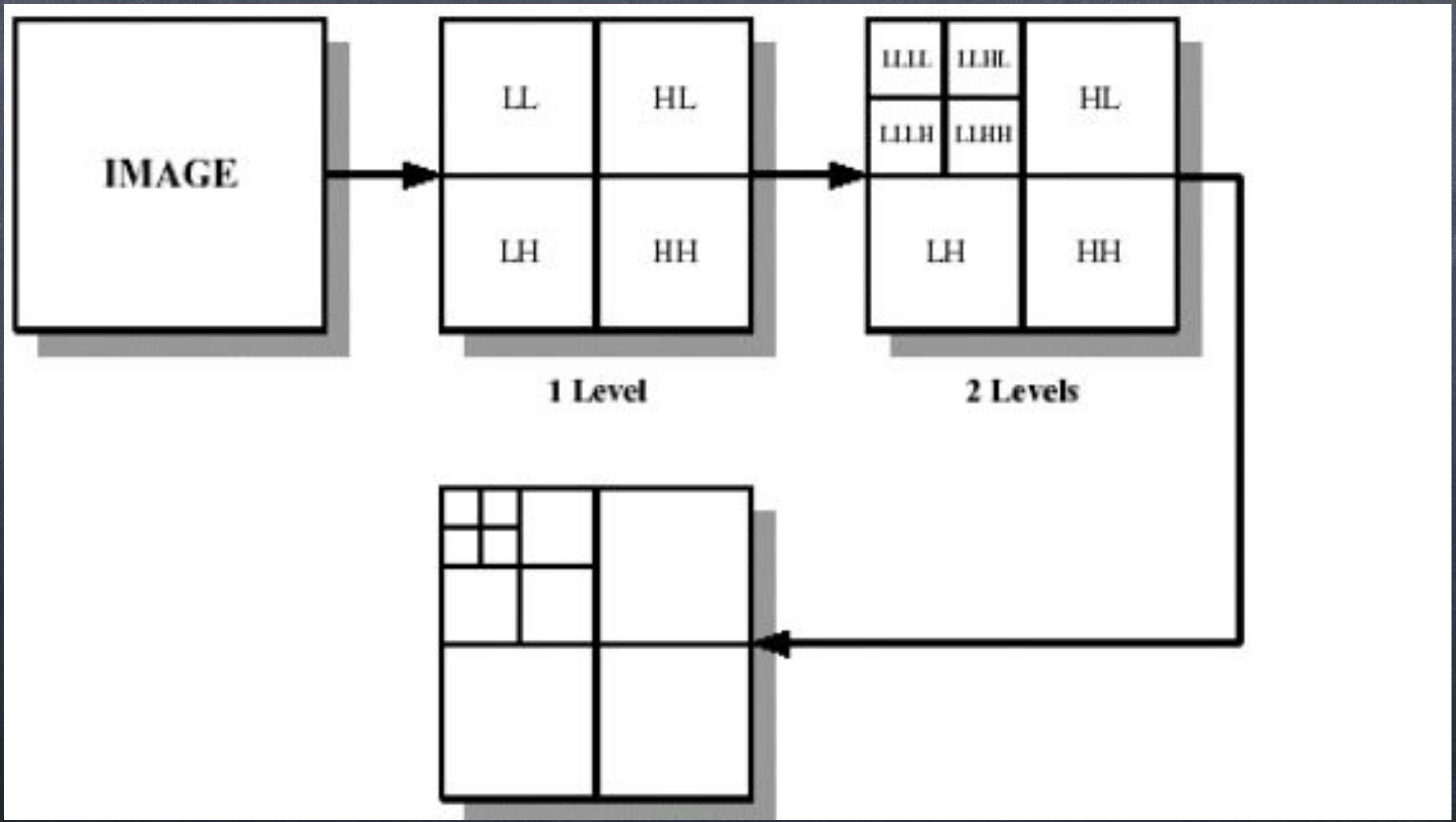


2-D Wavelet Transform: Reconstruction (IDWT)



Discrete Wavelet Transform

- 2-D DWT for Image



Common Wavelet Filters

- Haar: simplest, orthogonal, not very good
- Daubechies 8/8: orthogonal
- Daubechies 9/7: bi-orthogonal
  most commonly used if numerical reconstruction errors are acceptable
- LeGall 5/3: bi-orthogonal, integer operation, can be implemented with integer operations only, used for lossless image coding

Example of DWT (Haar Basis)

Let's consider a 1D 4-pixel Image [9 7 3 5]

[9 7 3 5]

Average (smoothing)

(9 + 7)/2 (3 + 5)/2

[8 4]

(8 + 4)/2

6

(8 – 4)/2

2

Detail coefficients (edge detection)

(9 - 7)/2 (3 - 5)/2

[1 -1]

WT

[6 2 1 -1]

Example

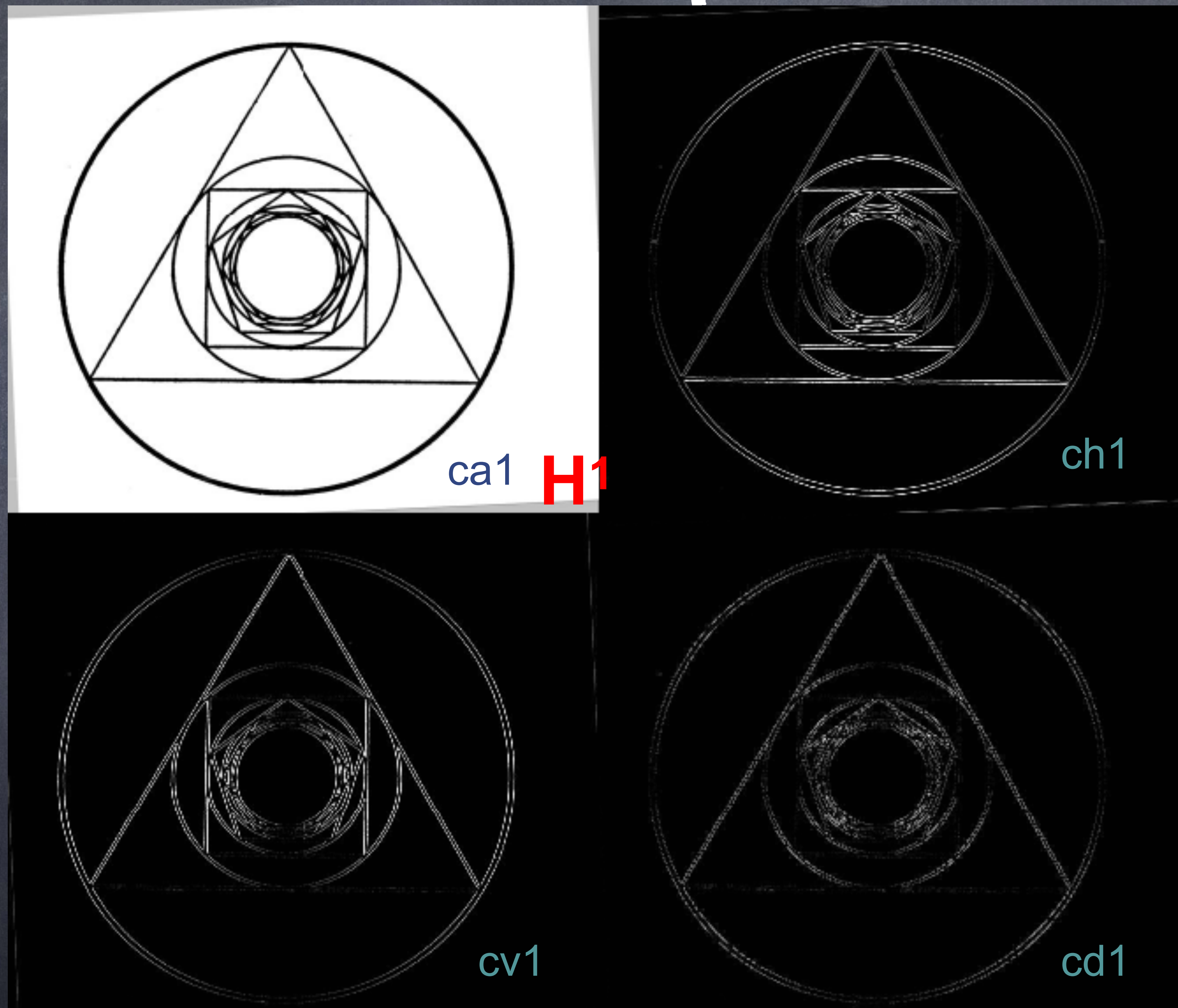
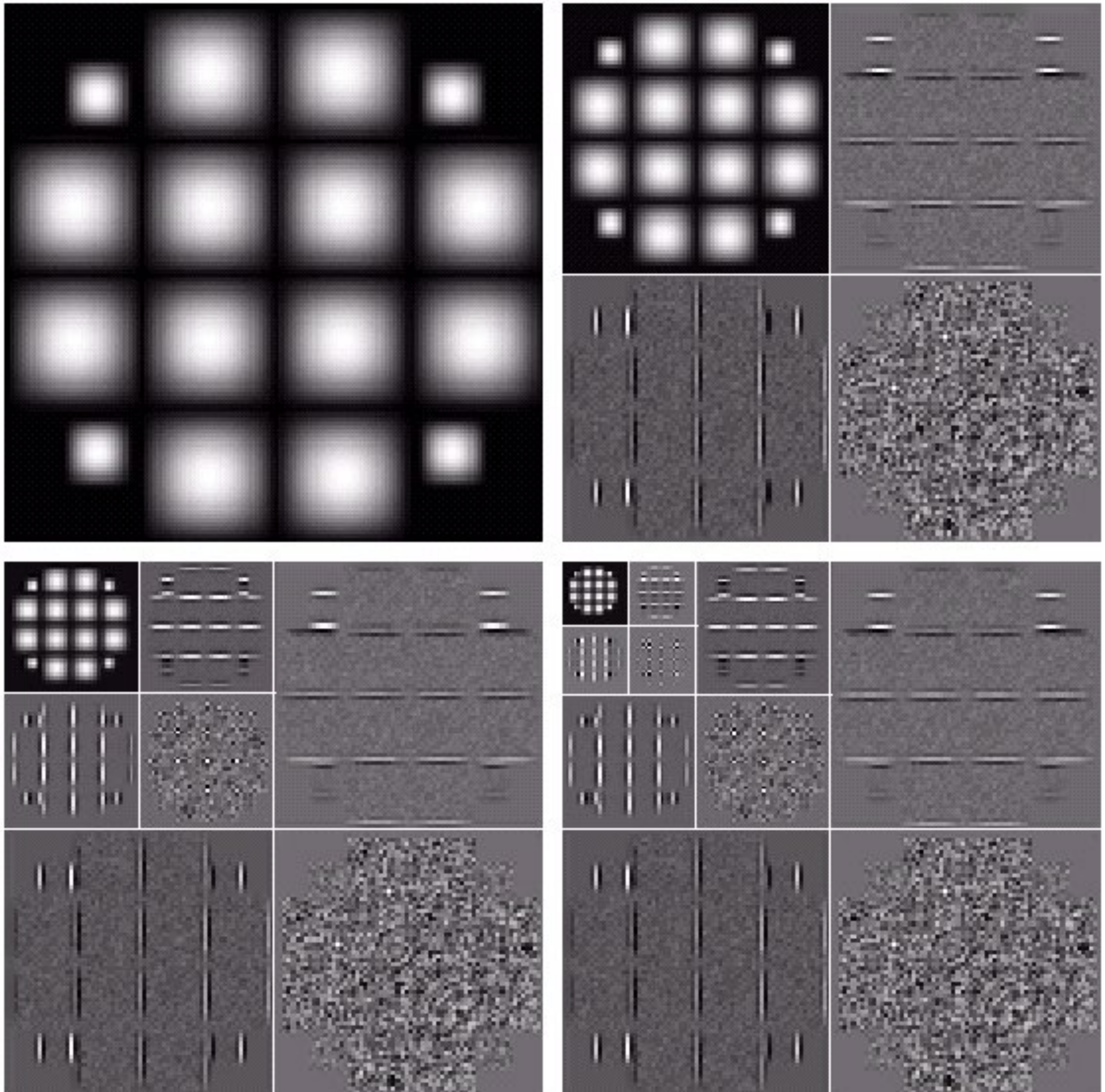


Image taken from : http://www.cs.ucla.edu/~klinger/circle.jpg



a	b
c	d

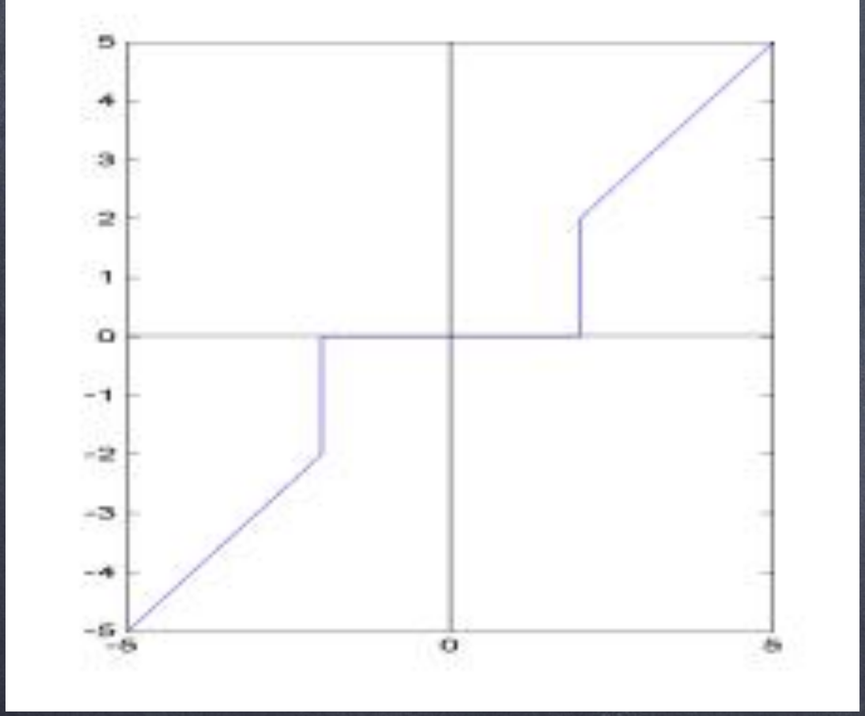
FIGURE 7.23 A three-scale FWT.

Image Denoising Using Wavelets

- Calculate DWT of the image.
- Threshold the wavelet coefficients. The threshold may be universal or subband adaptive.
- Compute the IDWT to get the denoised estimate.
- Soft thresholding is used in the different thresholding methods. Visually more pleasing images.

Threshold techniques

Hard threshold



Soft threshold

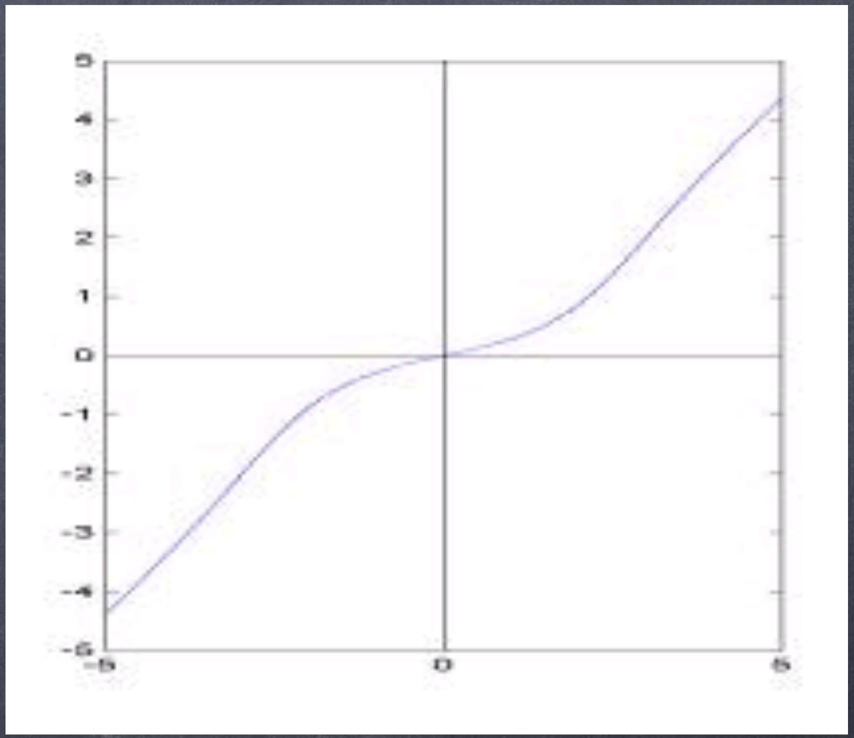


Image Enhancement

- Image contrast enhancement with wavelets, especially important in medical imaging
- Make the small coefficients very small and the large coefficients very large.
- Apply a nonlinear mapping function to the coefficients.

Enhancement

(a) Original Image

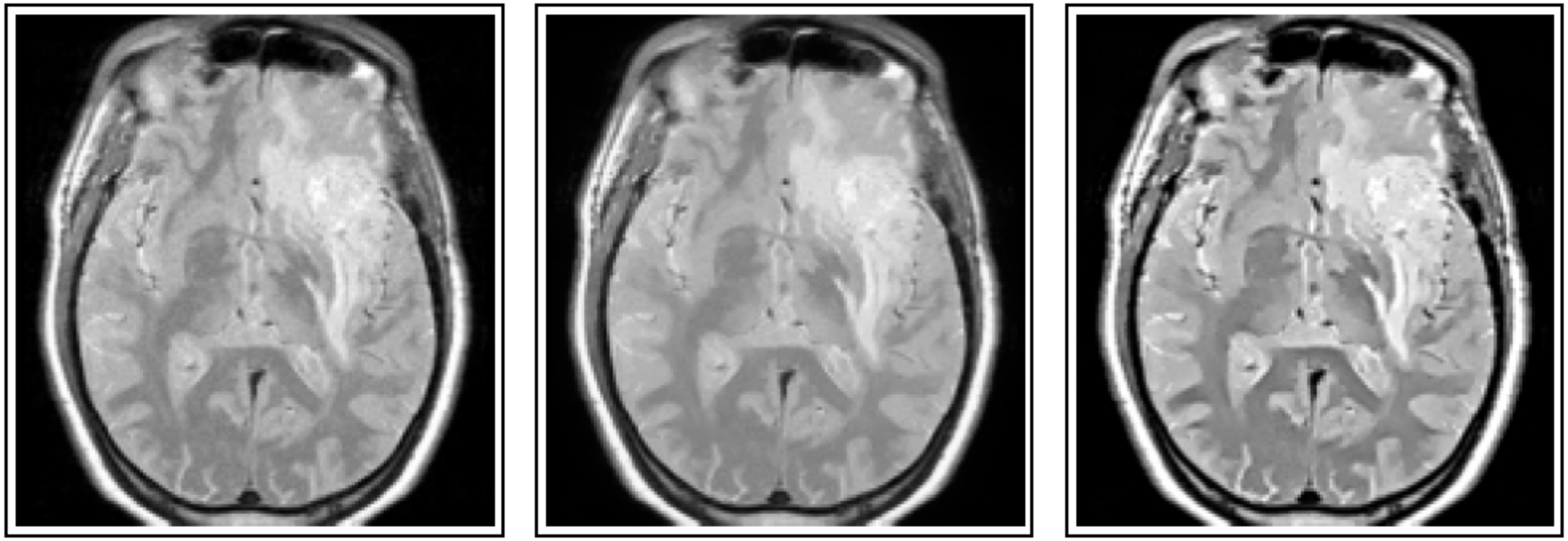
(c) Proposed Method

Denoising and Enhancement

- Apply DWT
- Shrink transform coefficients in finer scales to reduce the effect of noise
- Emphasize features within a certain range using a nonlinear mapping function
- Perform IDWT to reconstruct the image.

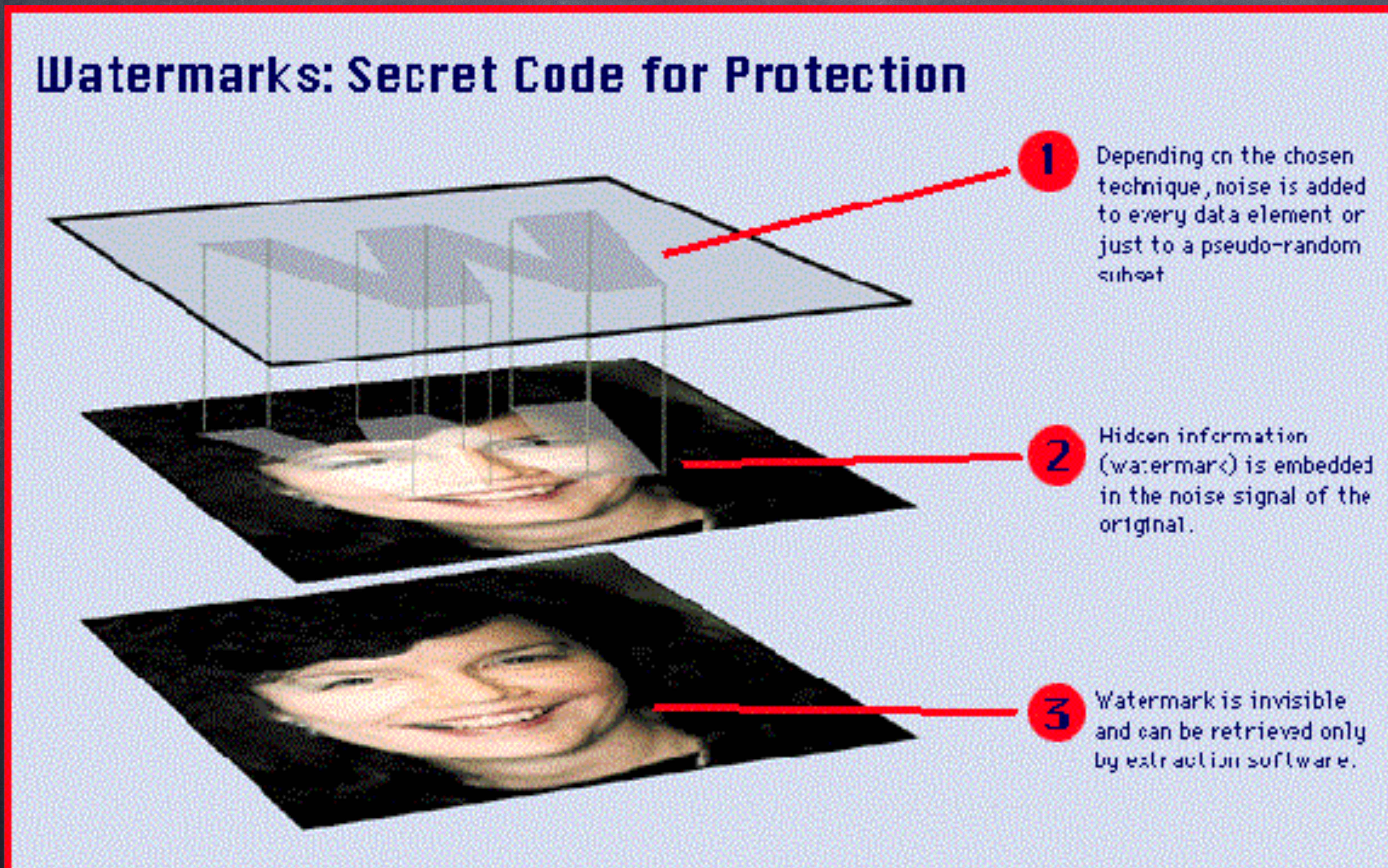
Examples

Original Denoised Denoising with Enhancement



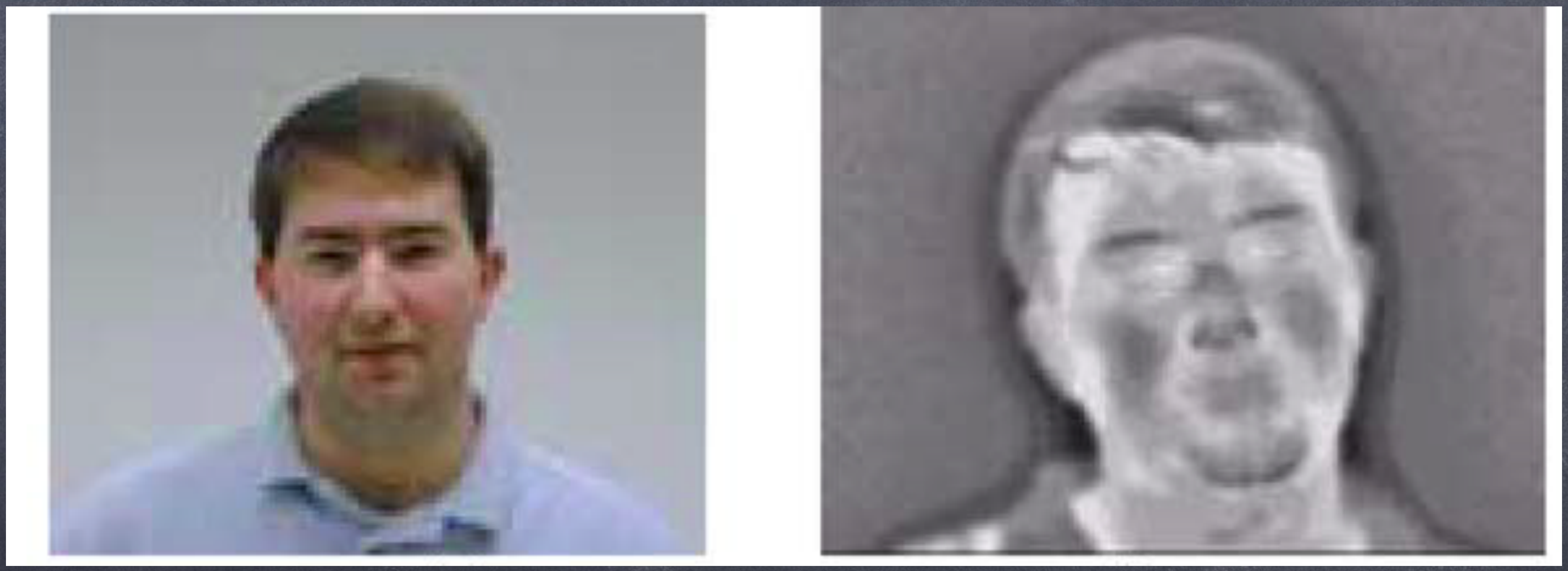
(a) (b) (c)

Watermarking



Watermarking

Fusion: How do we combine two images?



Assignment 1

1. Spatial filtering
2. Image pyramid
3. Edge detection
4. Wavelet processing