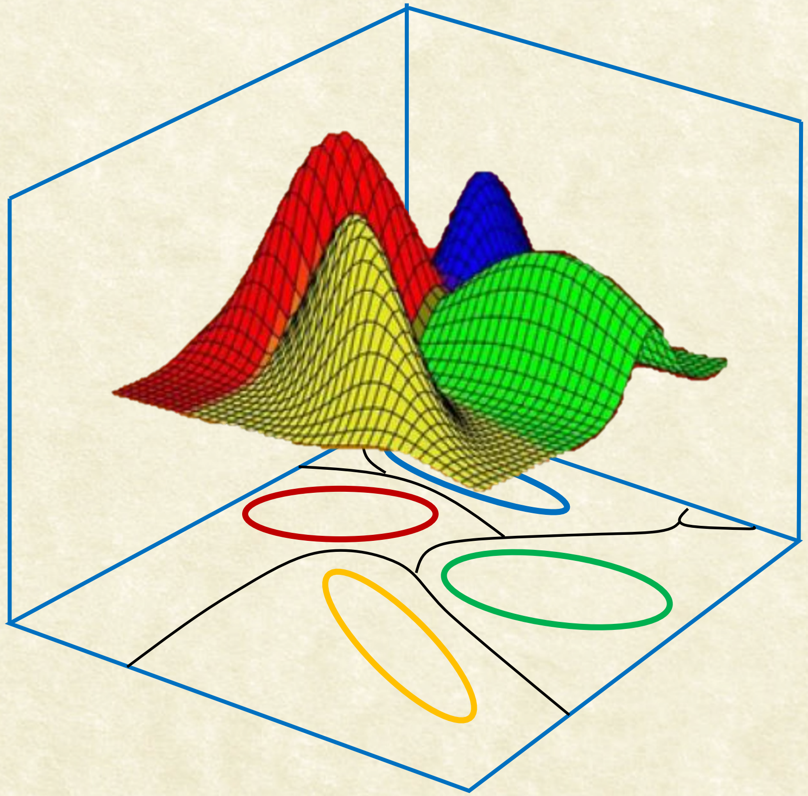
CS7.404: Digital Image Processing

Monsoon 2023: Linear Filters



Anoop M. Namboodiri

Biometrics and Secure ID Lab, CVIT,

IIIT Hyderabad

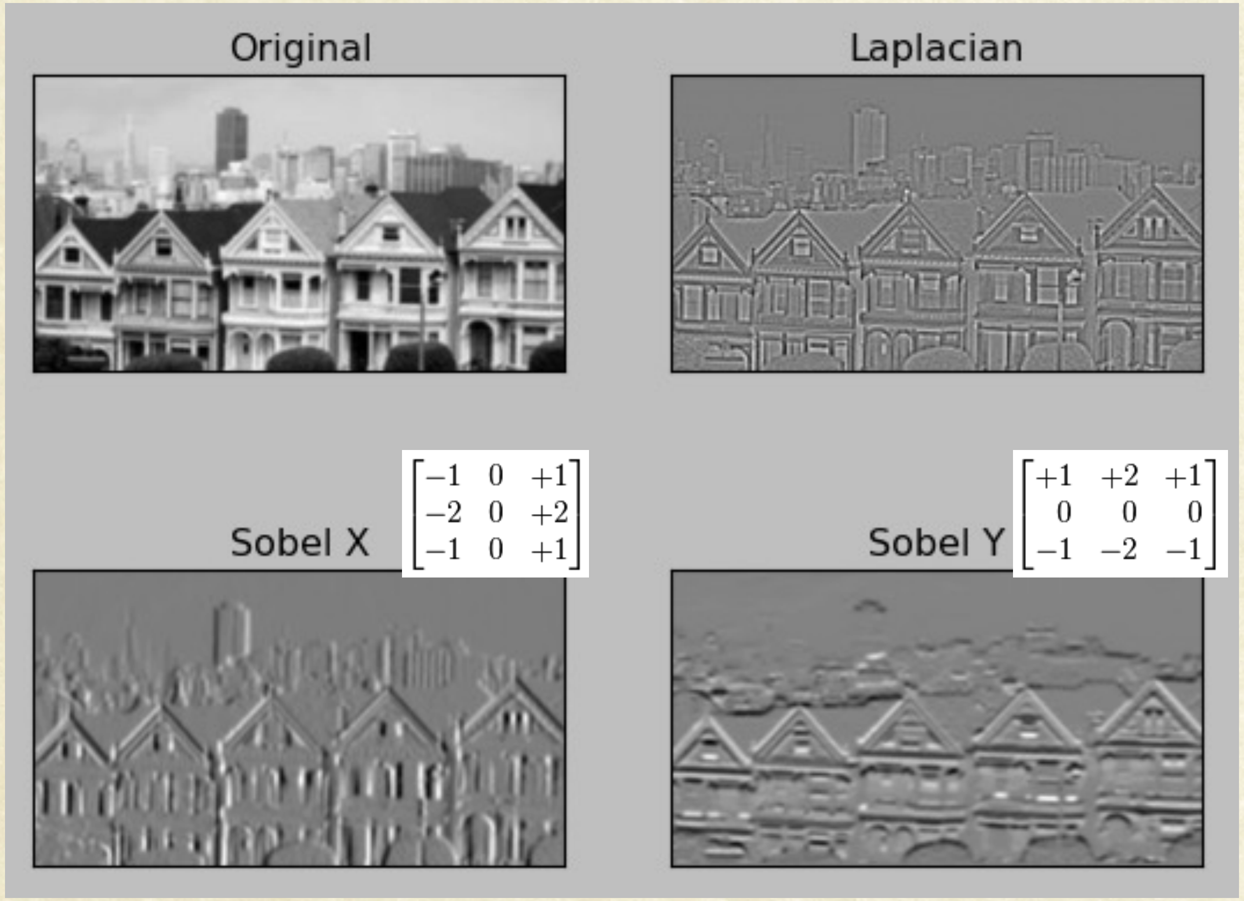
Recap: 2-D Laplacian Filter

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

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

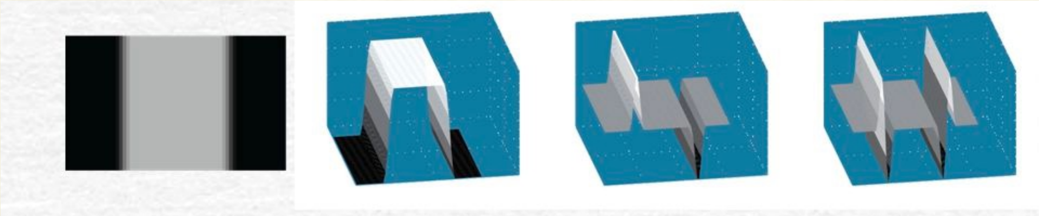
0	1	0
1	-4	1
0	1	0

Edge Masks: Sobel, Laplacian

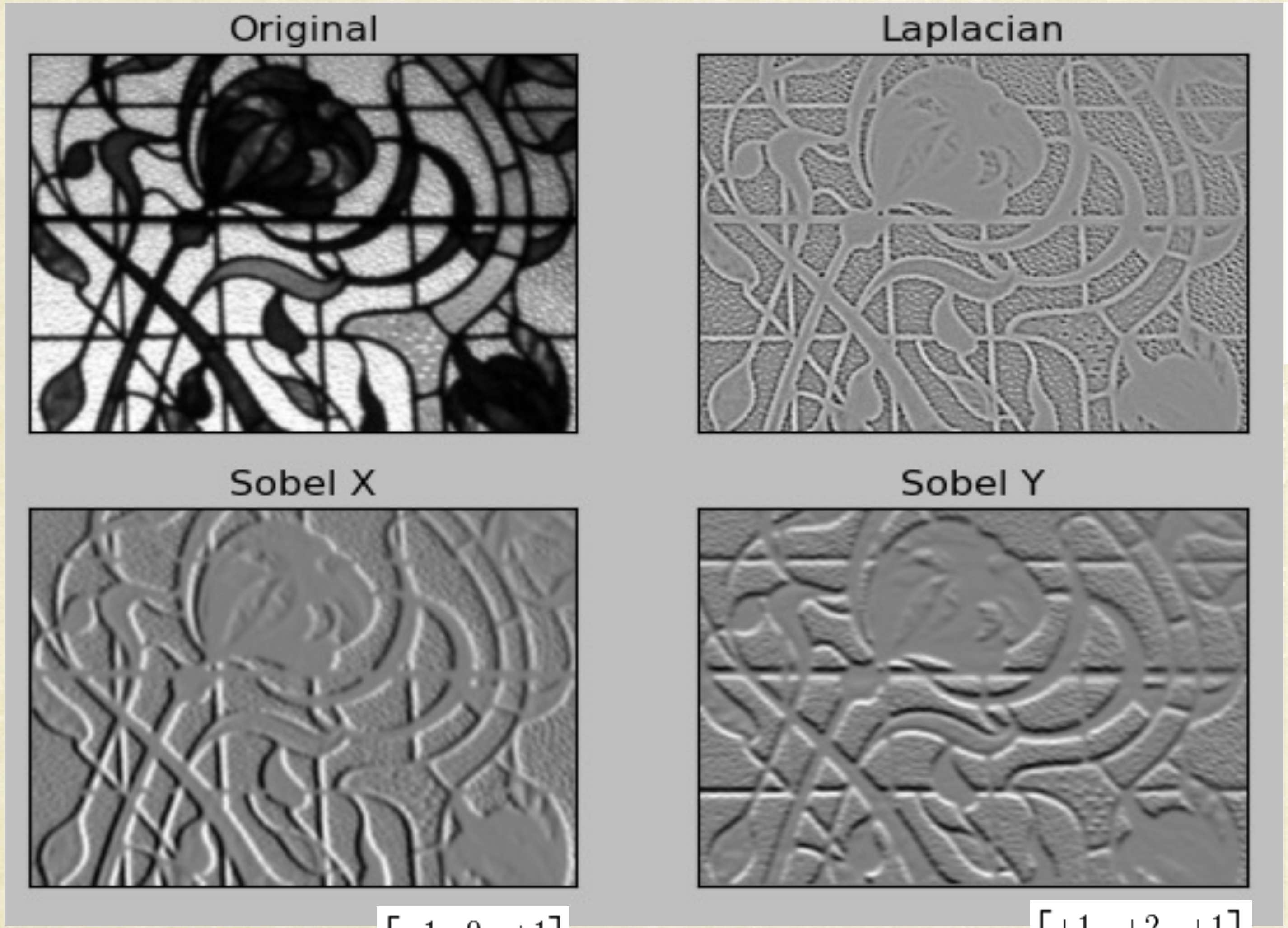


0	−1	0
−1	4	−1
0	−1	0

Note: Coefficients sum to 0



Edge Masks: Sobel, Laplacian



0	−1	0
−1	4	−1
0	−1	0

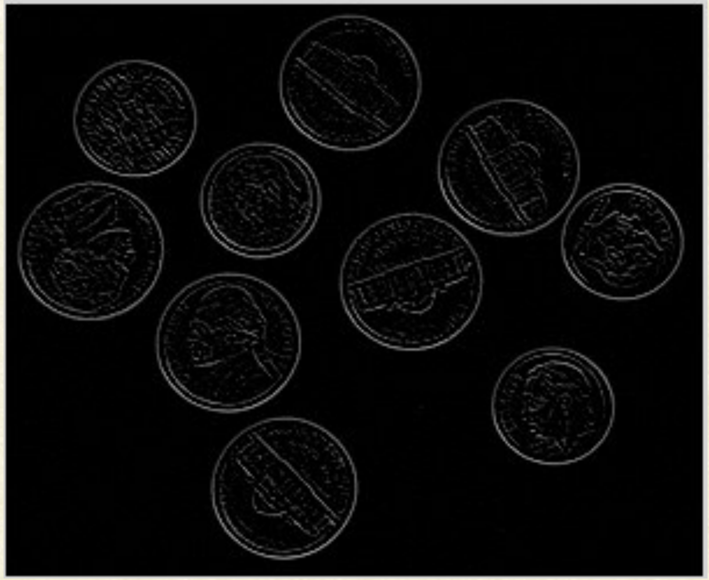
$$\begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$$

$$\begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Image Sharpening

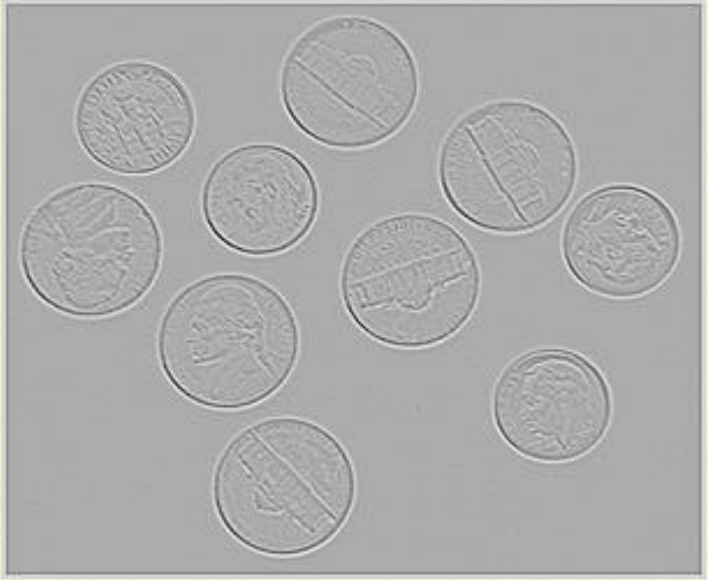
$I(u,v)$

$\nabla^2 I(u,v)$



$\nabla^2 I(u,v) + 128$

(For visualization)



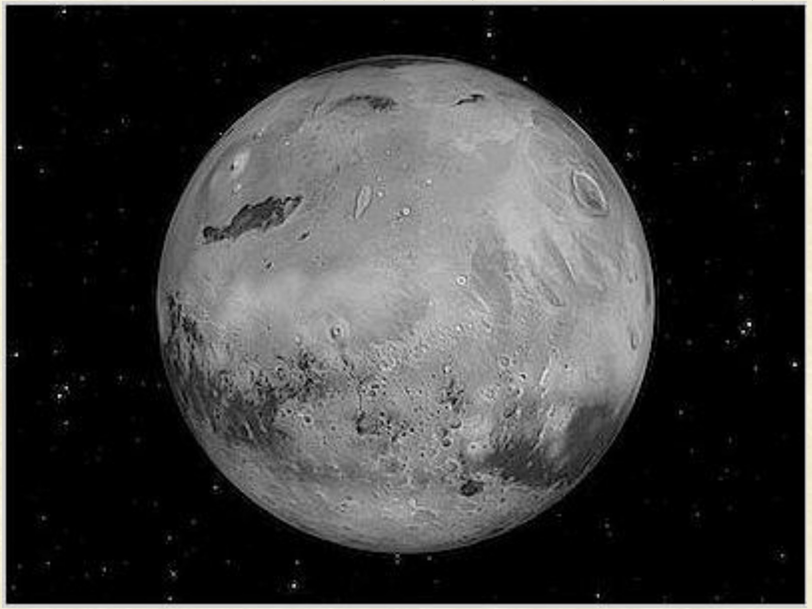
$I'(u,v) = I(u,v) + \nabla^2 I(u,v)$

Sharpening (Unsharp Masking)

$I(u,v)$

$\nabla\ I(u,v)$

$I'(u,v) = I(u,v) + \nabla^2 I(u,v)$



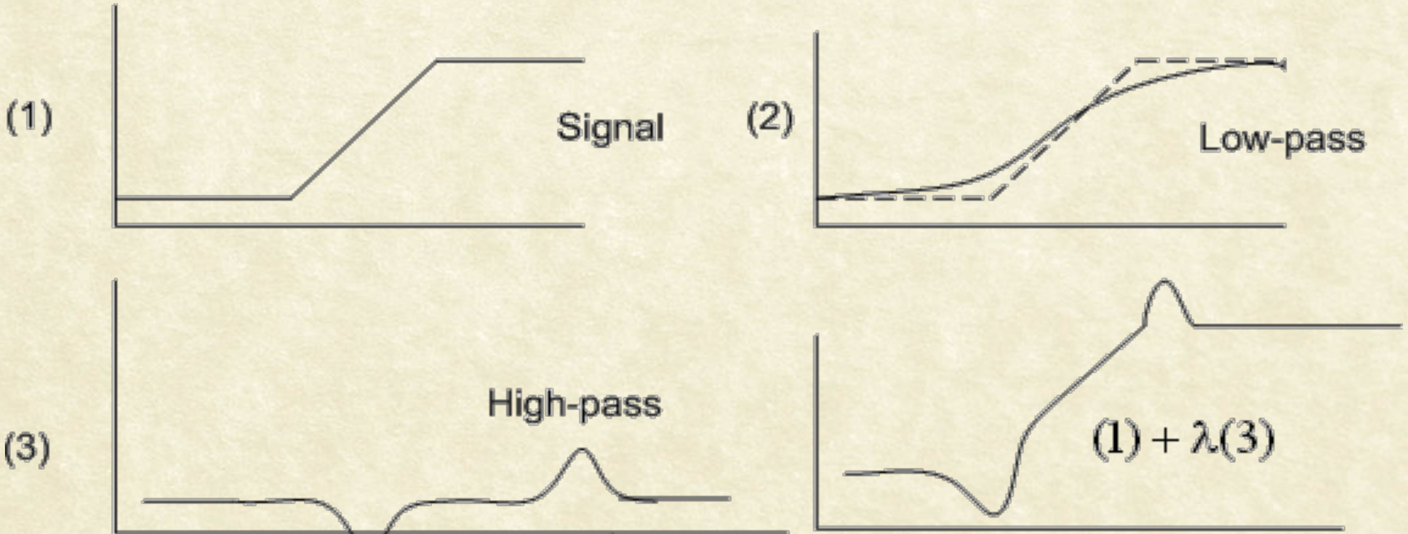
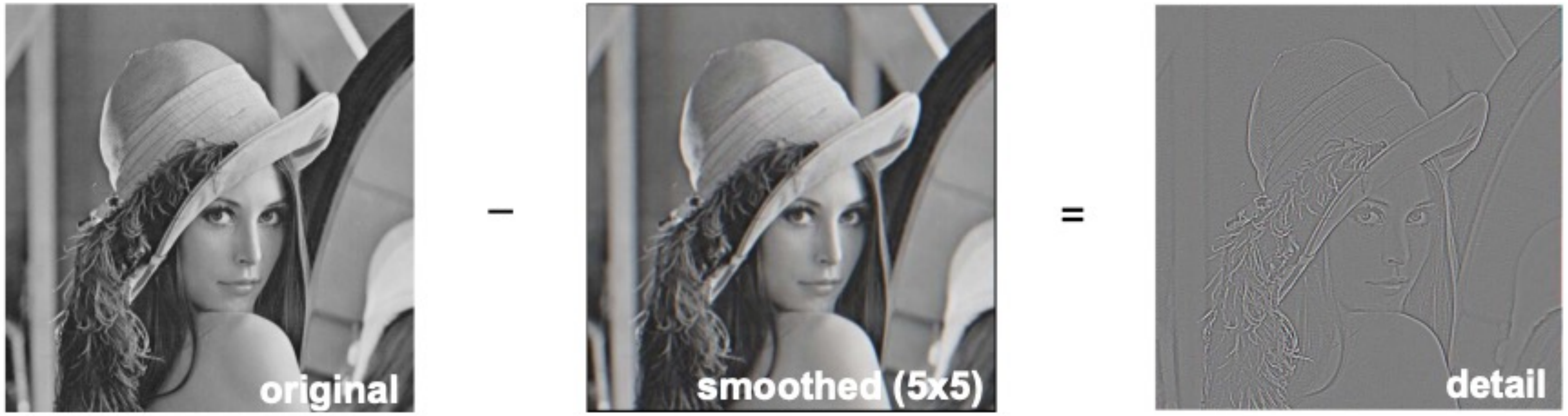


Image Courtesy:NASA

Highboost Filtering

- What does blurring take away?



- Let’s add it back:

Unsharp Masking vs Highboost Filtering

USM vs. HBF as Spatial Filters

A=1

w = 9A-1

-1	-1	-1
-1	w	-1
-1	-1	-1

A=2

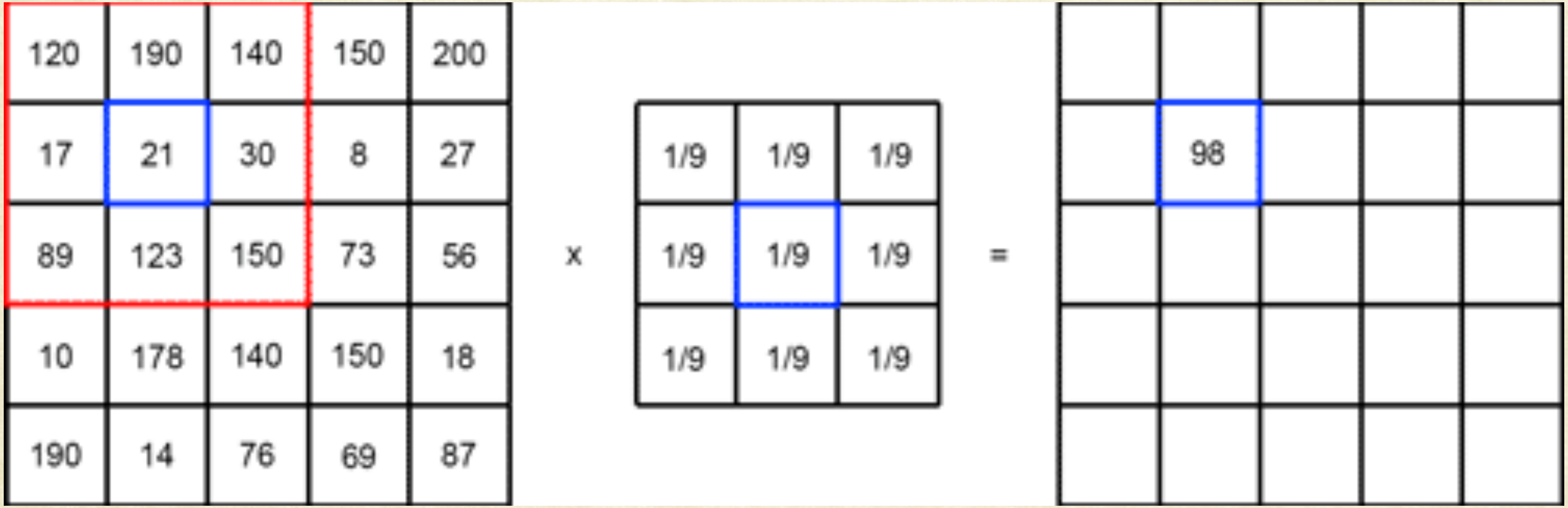
w = 17

-1	-1	-1
-1	17	-1
-1	-1	-1

- If **A=1**, we get unsharp masking. $I'(u,v) = I(u,v) + \nabla^2 I(u,v)$
- If **A>1**, original image is **added back** to detail image (HBF).

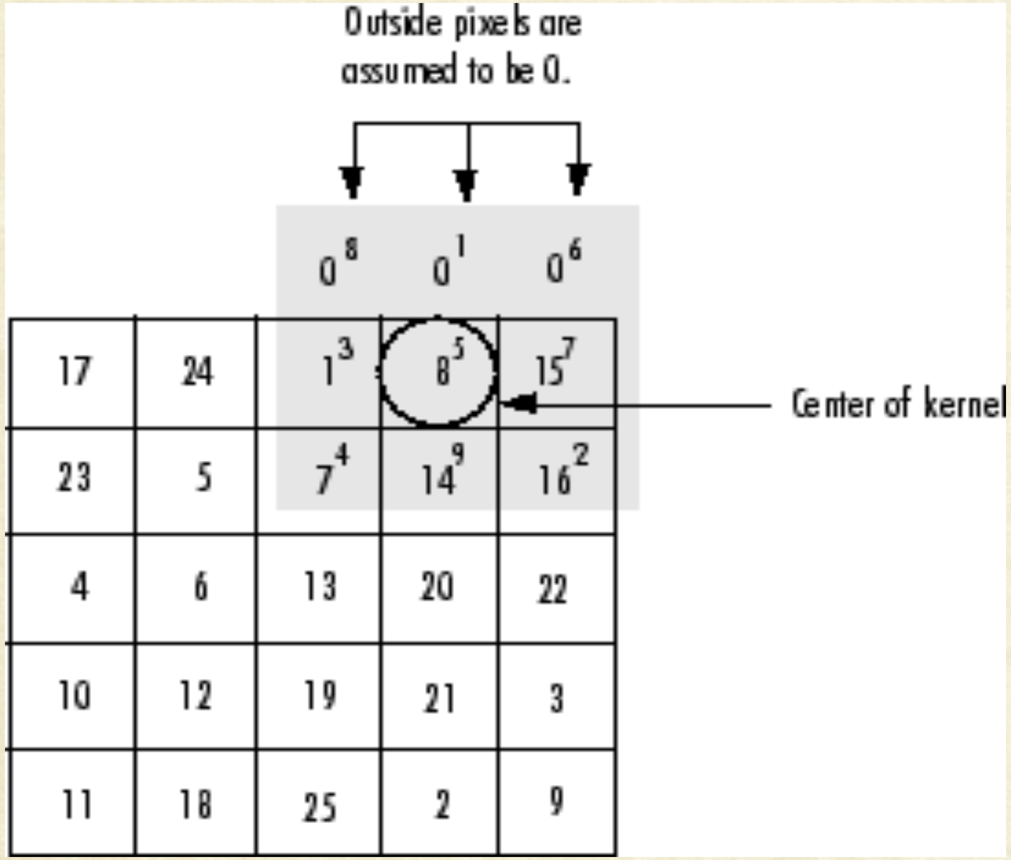
Corner Cases: Padding

```
M = 3
For each valid location [x,y] in S
        a ← Average of intensities in a M x M neighborhood centered on [x,y]
        D[x,y] = round(a)
```

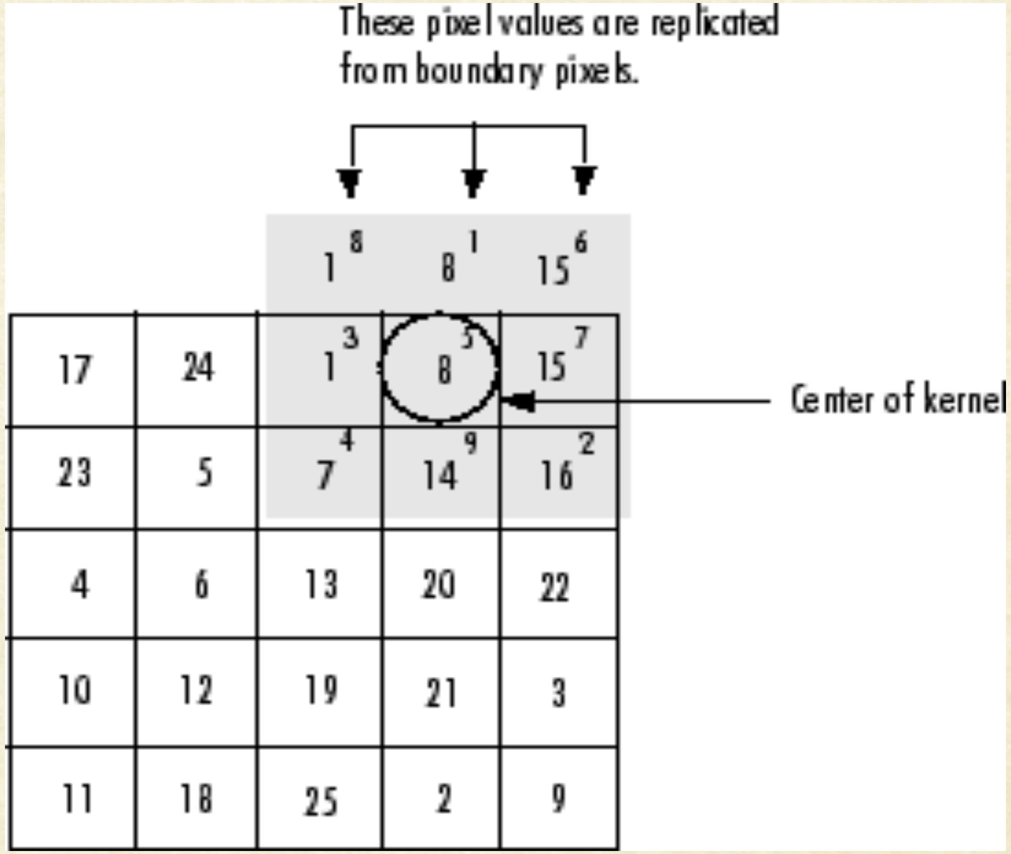


http://199.116.233.101/index.php?title=File:Animated-kernels.gif

Image Padding



Zero/Constant

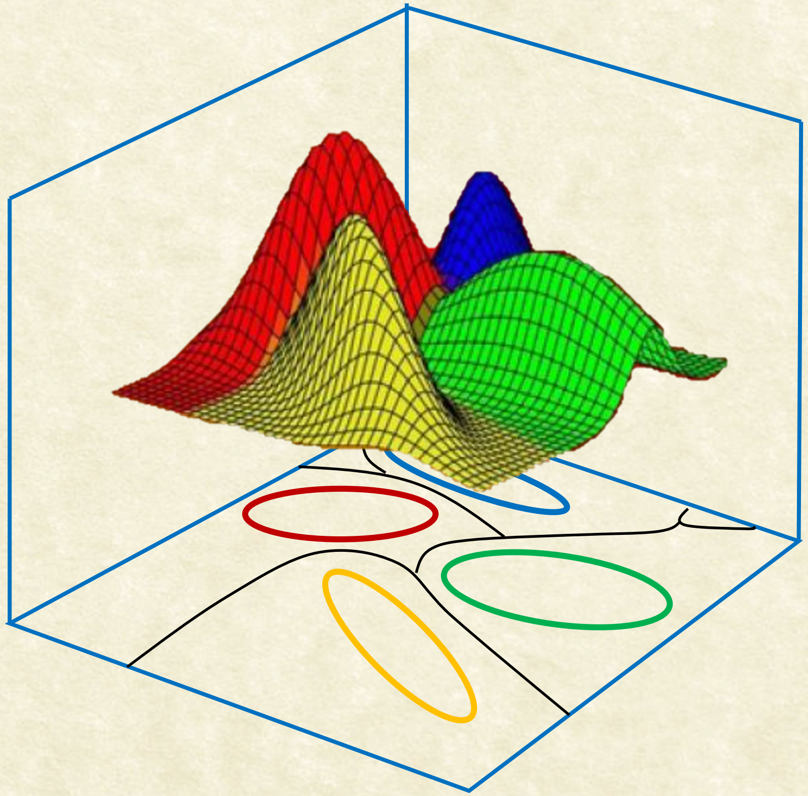


Replicate/Reflect

Questions?

CS7.404: Digital Image Processing

Monsoon 2023: Linear Shift Invariant Systems



Anoop M. Namboodiri

Biometrics and Secure ID Lab, CVIT,

IIIT Hyderabad

Linear Systems

$$g(x) = T_H(f(x))$$

$$f_1(x) \longrightarrow \boxed{T_H(\,)} \longrightarrow g_1(x) \qquad f_2(x) \longrightarrow \boxed{T_H(\,)} \longrightarrow g_2(x)$$

$$\alpha f_1(x) + \beta f_2(x) \longrightarrow \boxed{T_H(\,)} \longrightarrow \alpha g_1(x) + \beta g_2(x)$$

Shift Invariant Systems

$$f(x) \longrightarrow \boxed{T_H(\,)} \longrightarrow g(x)$$

$$f(x+\delta) \longrightarrow \boxed{T_H(\,)} \longrightarrow g(x+\delta)$$

Linear Shift Invariant Systems

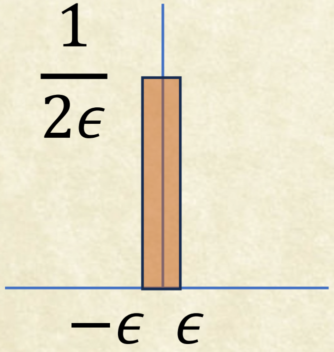
- Systems that are both Linear and Shift Invariant
- All convolution operations are Linear Shift Invariant
- All Linear Shift Invariant Systems can be expressed as Convolutions
- Characterized by Impulse Response or PSF
- Additional Properties
  - Separability

Impulse Response in Practice

$$f(x) \longrightarrow \boxed{T_H(\,)} \longrightarrow g(x)$$

$$f(x+\delta) \longrightarrow \boxed{T_H(\,)} \longrightarrow g(x+\delta)$$

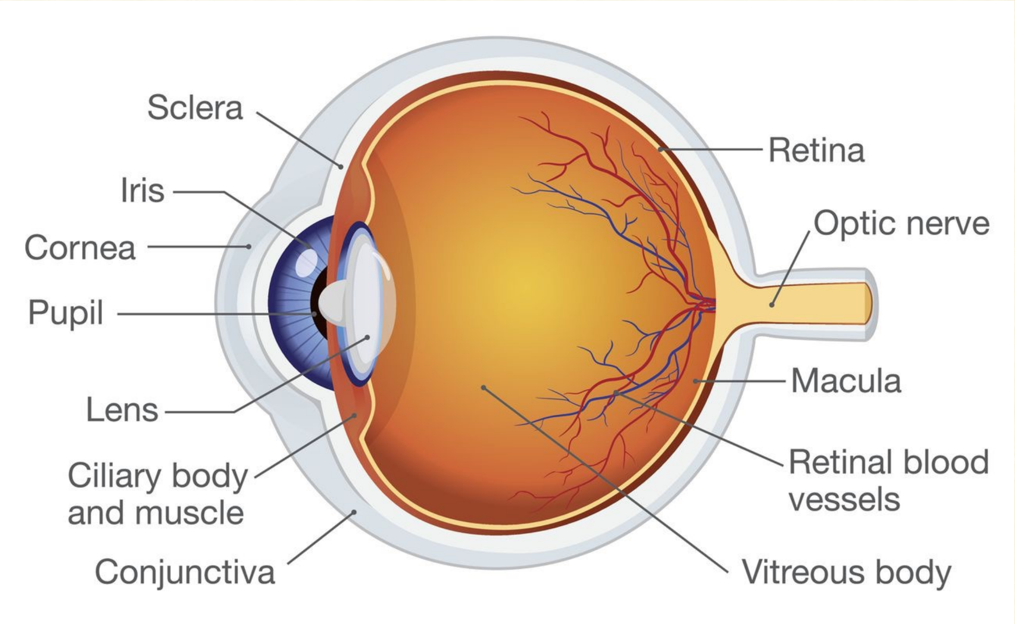
Impulse Response in Practice



$$I(x) \longrightarrow \boxed{T_H(\,)} \longrightarrow H(x)$$: IR or PSF

Impulse Response of Human Eye

- Input: Point Light Source ?
  - A bright (distant) star
- System: Cornea, Eye Lens, Vitreous, Retina
- Output: PSF



Homework

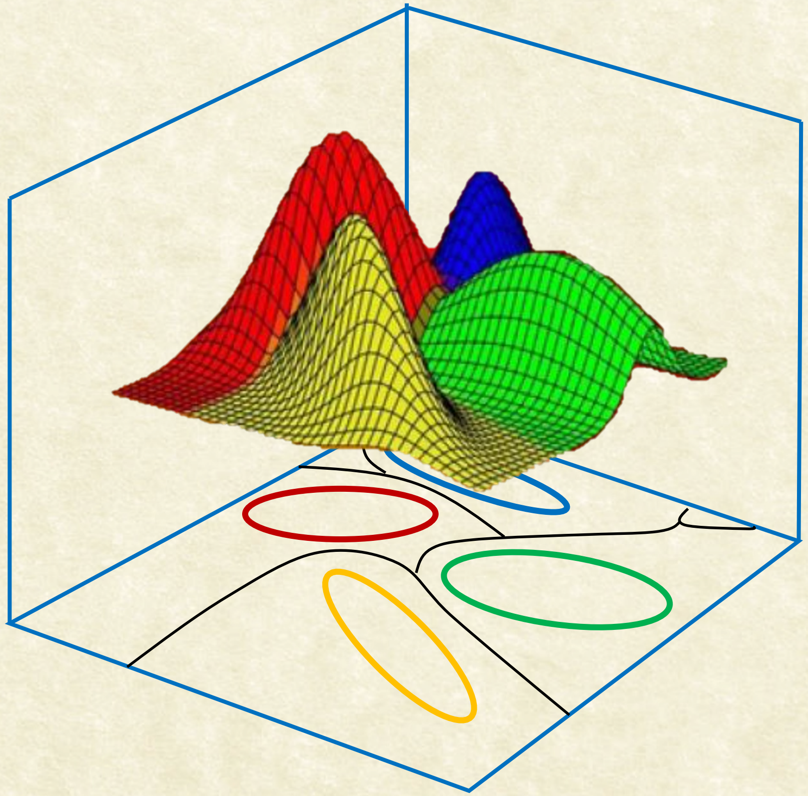
1. Show that Gaussian Convolutions are both LSI and Separable

2. Show that Impulse Response can fully characterize a LSIS

Questions?

CS7.404: Digital Image Processing

Monsoon 2023: Non-Linear Filters

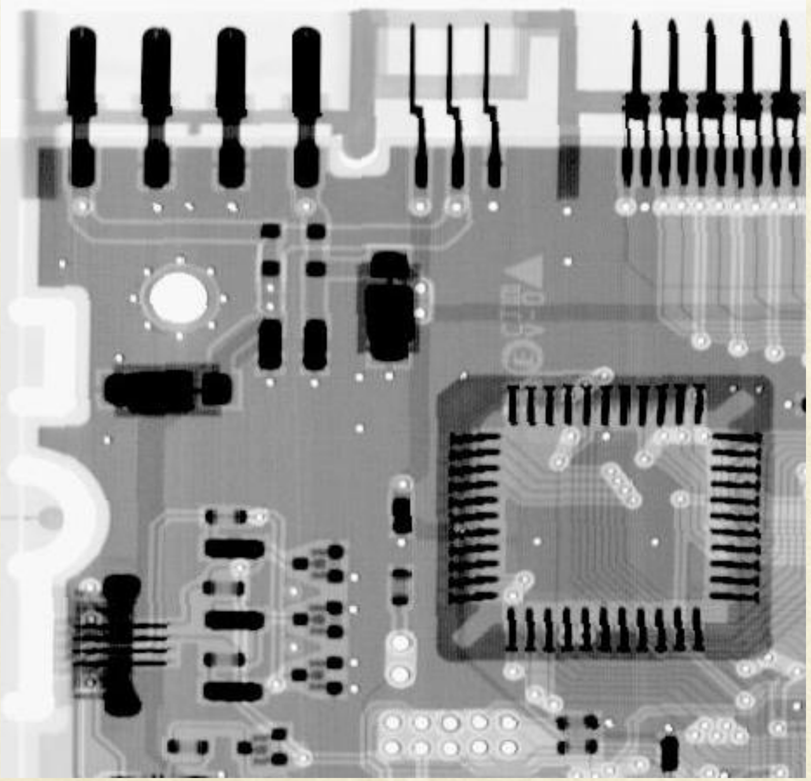


Anoop M. Namboodiri

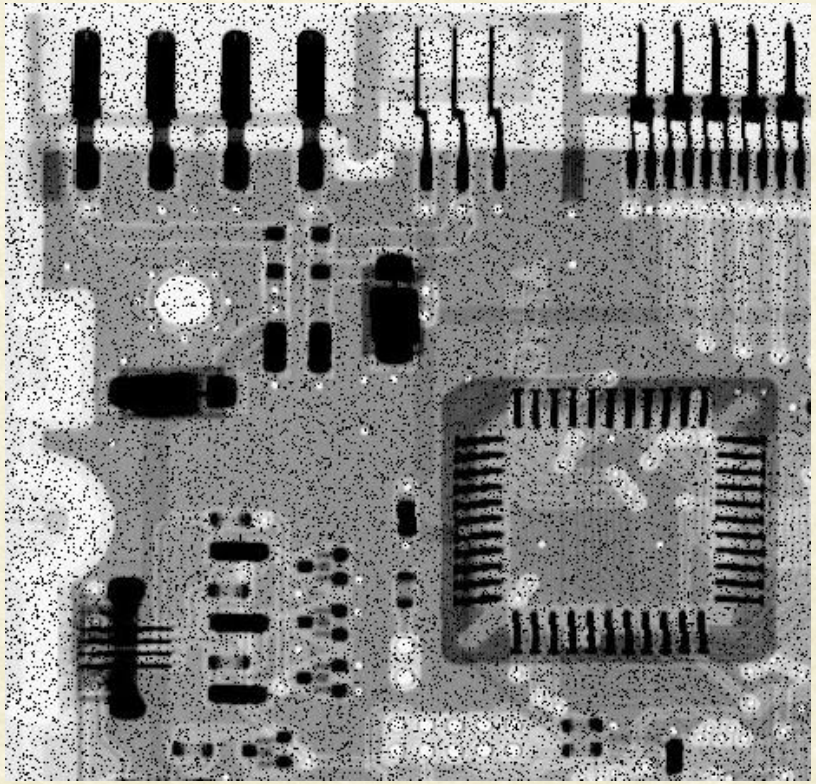
Biometrics and Secure ID Lab, CVIT,

IIIT Hyderabad

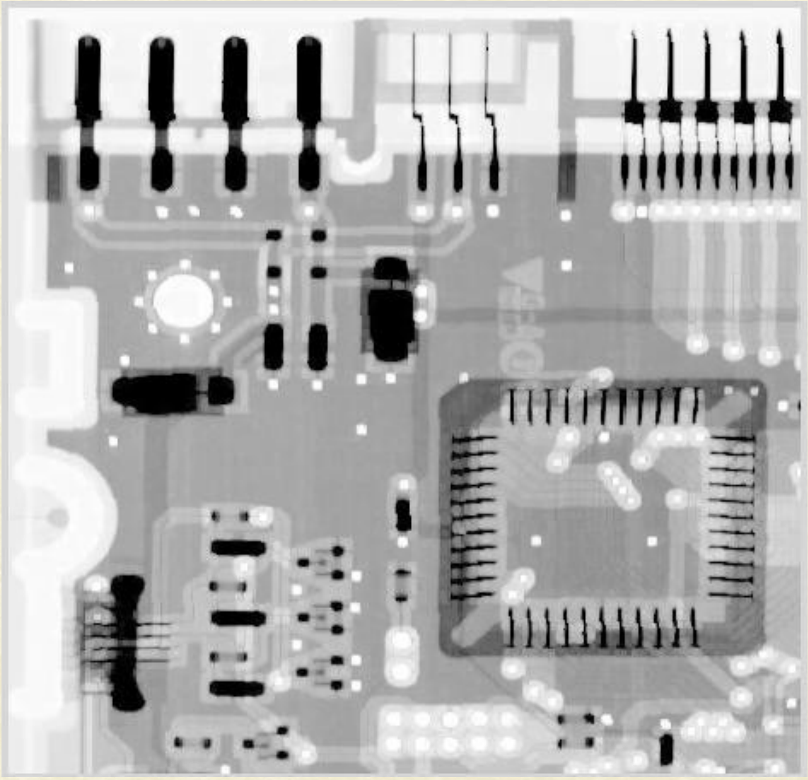
Non-linear Spatial Filters (max)



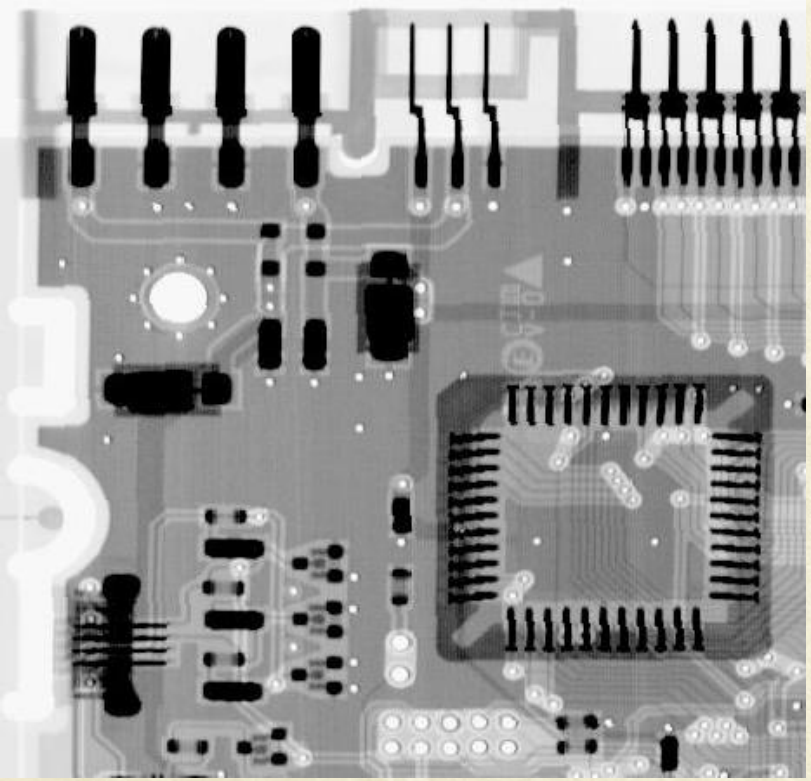
Pepper noise



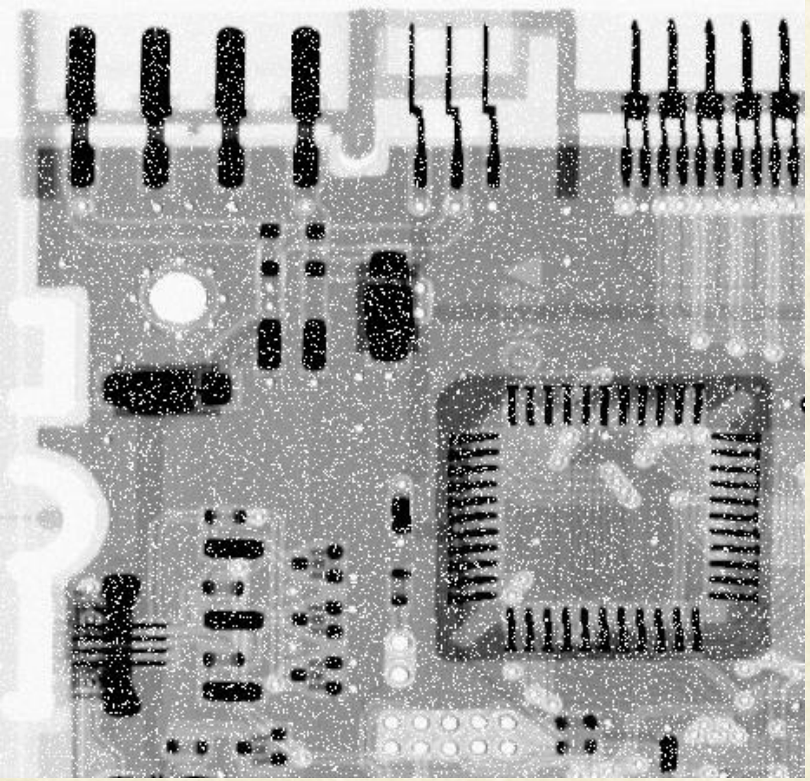
After applying max filter



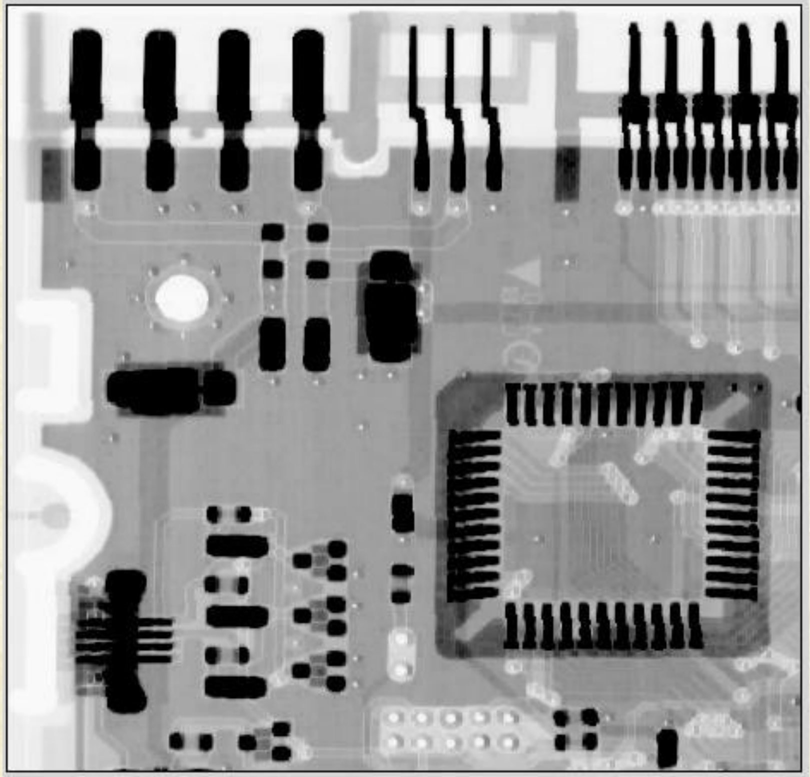
Non-linear Spatial Filters (min)



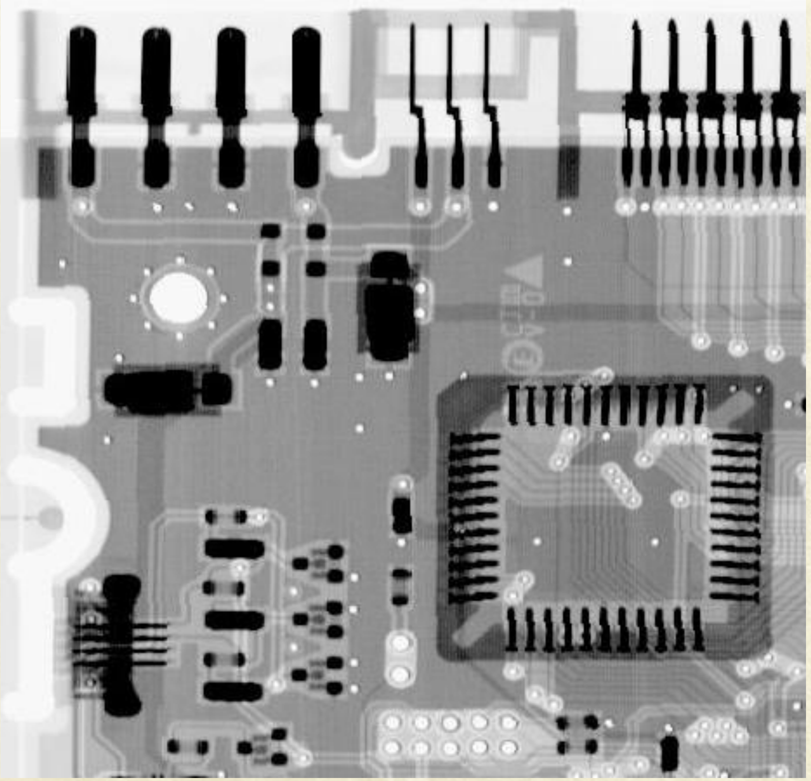
Salt noise



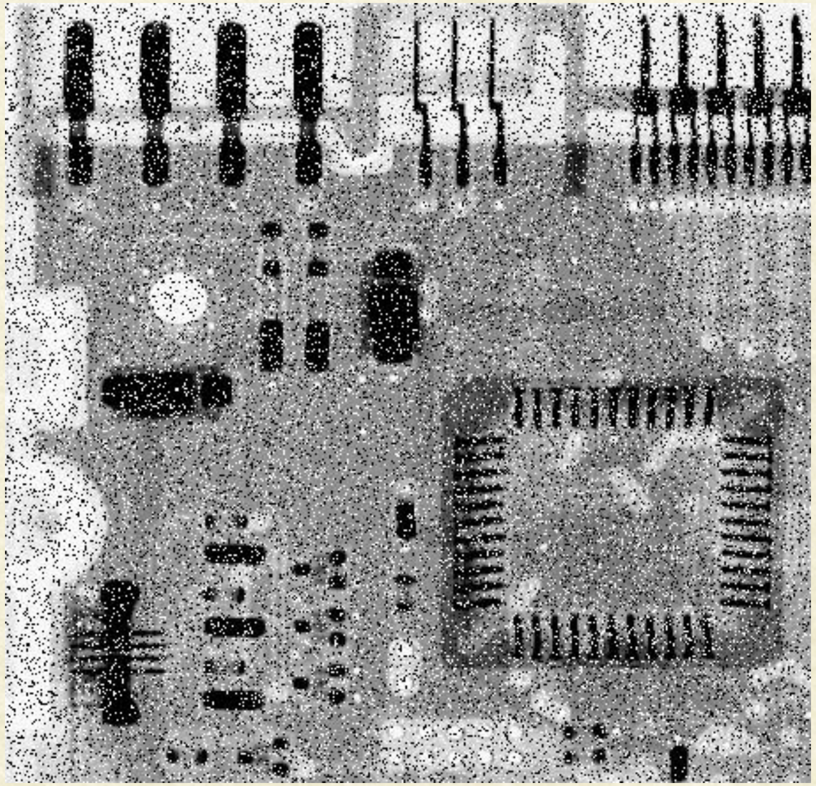
After applying min filter



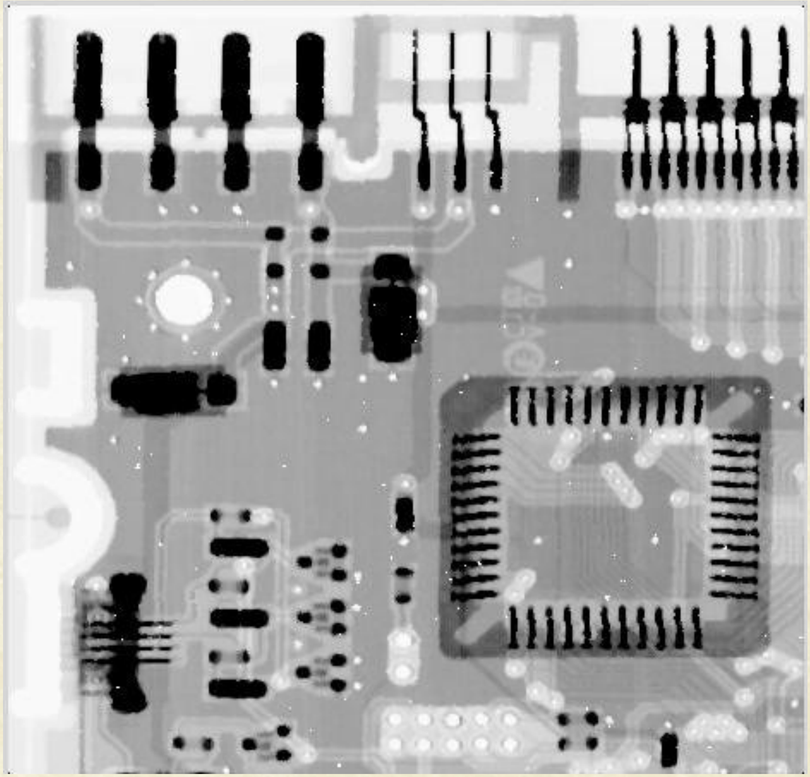
Non-linear Spatial Filters (median)



Salt & Pepper noise



After applying median filter

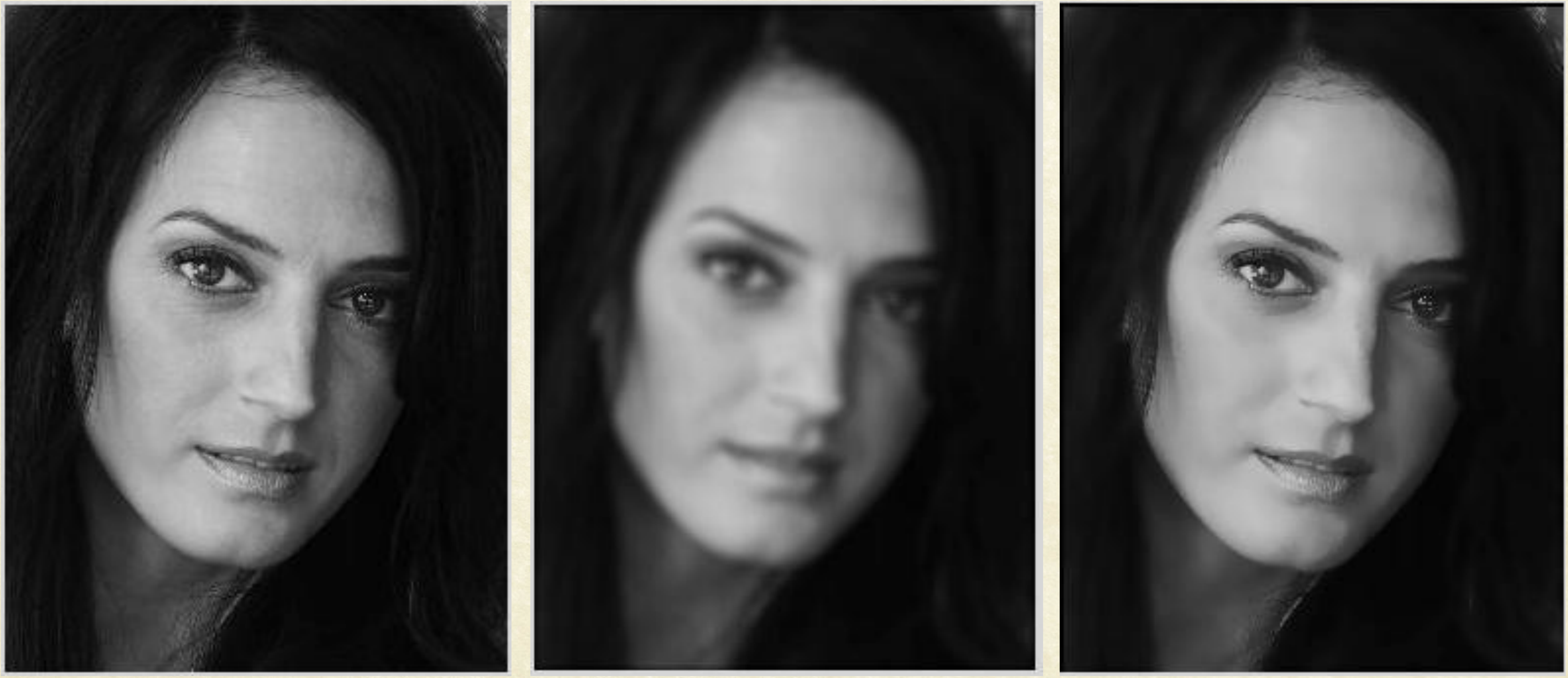


max, min, median → also known as rank / order statistic filters

Other Spatial Filters

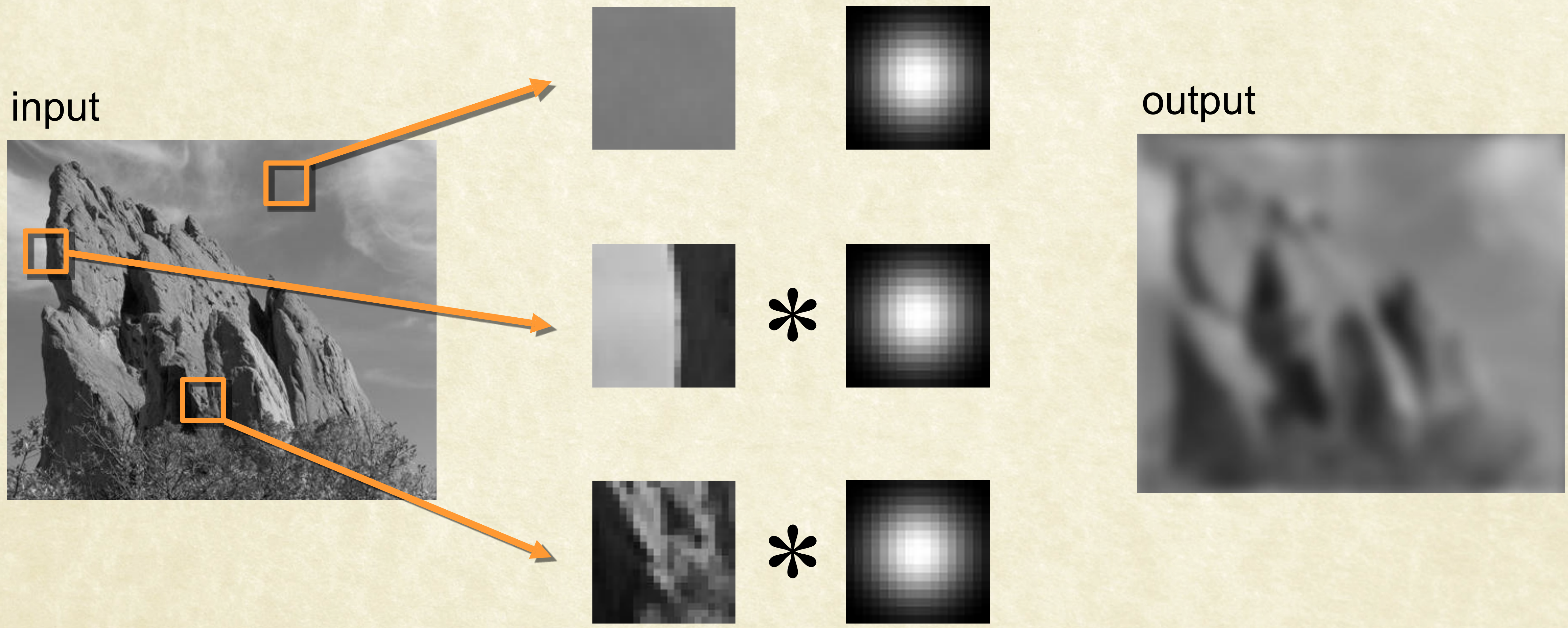
- Geometric mean
- Harmonic mean
- Contra harmonic mean
- Mid Point filter
- Alpha trimmed mean filter
- …..

Bilateral Filtering (Edge preserving smoothing)



Original image taken from cs.cityu.edu.hk

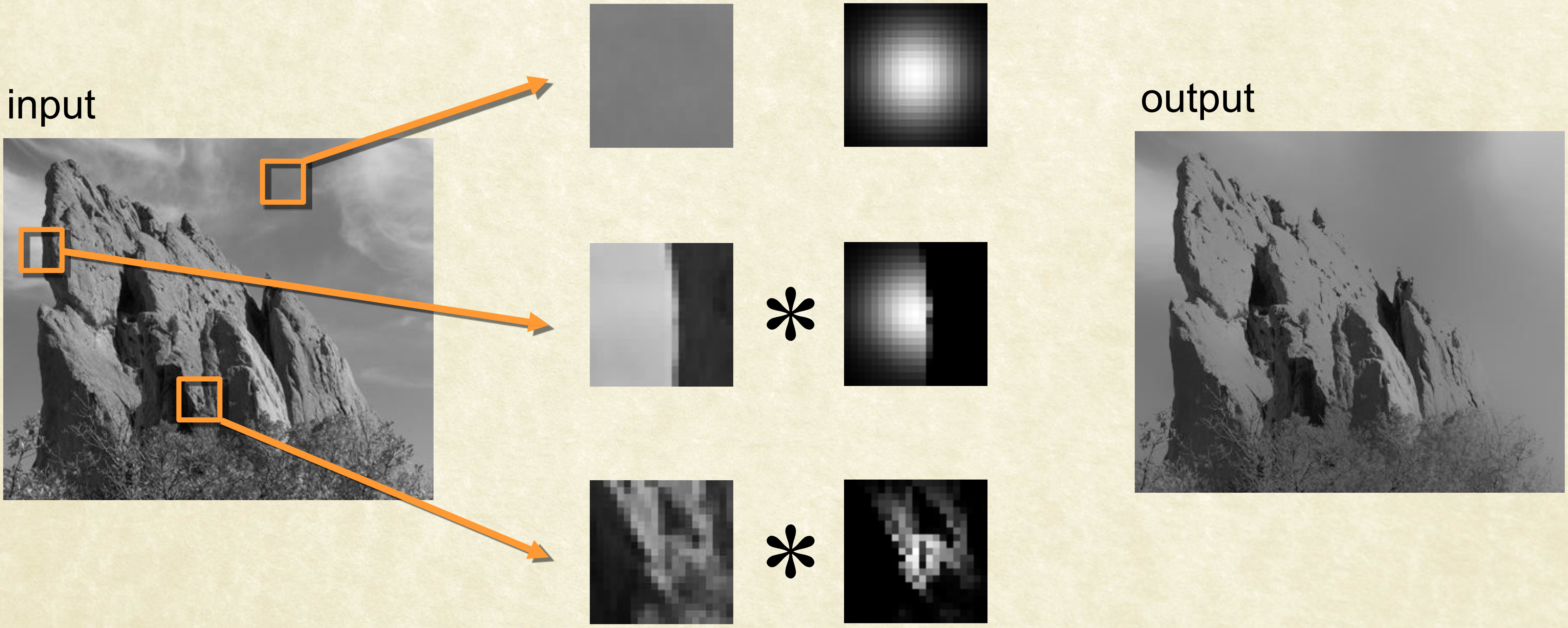
*Usual Gaussian Filtering



Same Gaussian kernel everywhere.

Source: Sylvain Paris

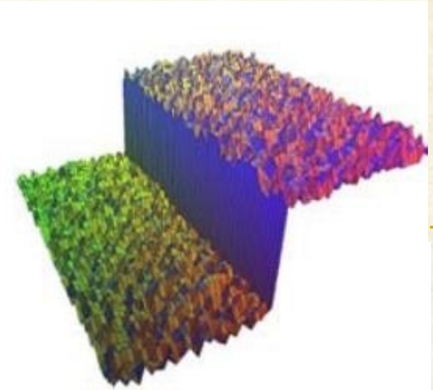
Bilateral Filtering =



The kernel shape depends on the image content.

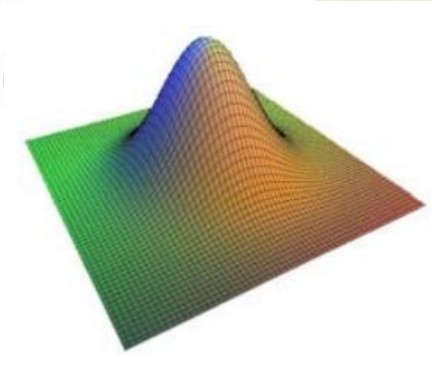
Source: Sylvain Paris

Bilateral Filter



$$g(i,j) = \frac{\sum_{k,l} f(k,l)w(i,j,k,l)}{\sum_{k,l} w(i,j,k,l)}.$$

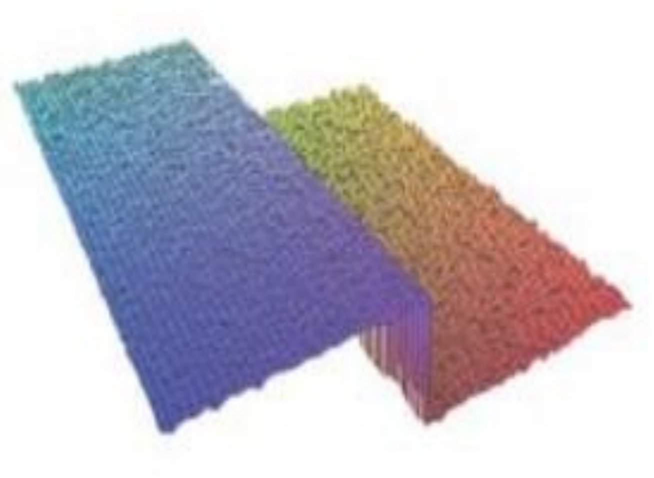
The weighting coefficient $w(i,j,k,l)$ depends on the product of a *domain kernel*

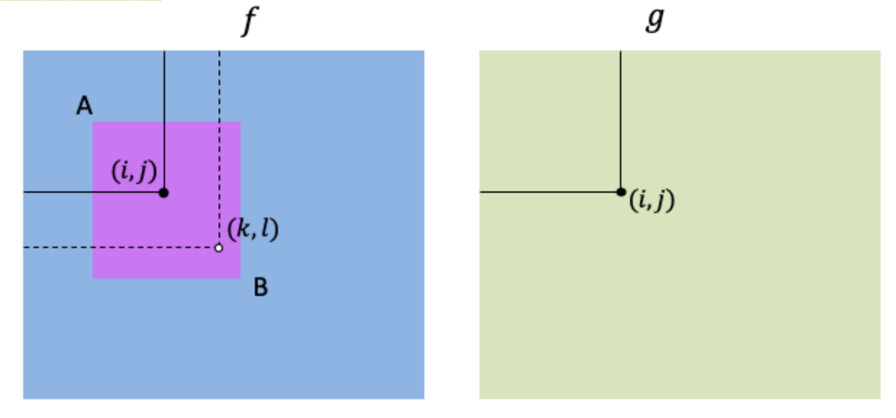


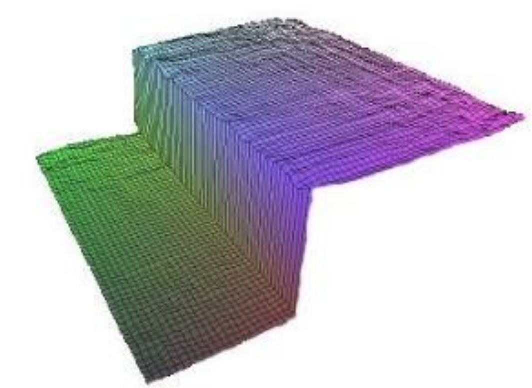
$$d(i,j,k,l) = \exp\left(-\frac{(i-k)^2+(j-l)^2}{2\sigma_d^2}\right),$$

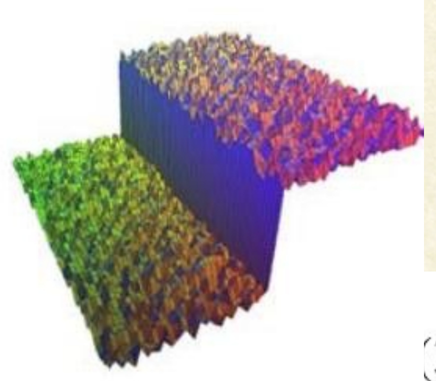
and a data-dependent *range kernel* (Figure 3.19d),

$$r(i,j,k,l) = \exp\left(-\frac{\|f(i,j)-f(k,l)\|^2}{2\sigma_r^2}\right).$$

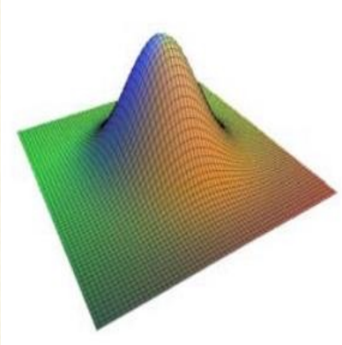








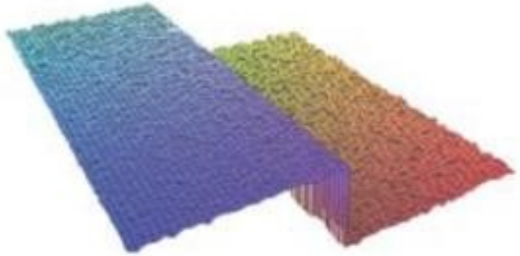
$$g(i,j) = \frac{\sum_{k,l} f(k,l)w(i,j,k,l)}{\sum_{k,l} w(i,j,k,l)}. \tag{3.34}$$



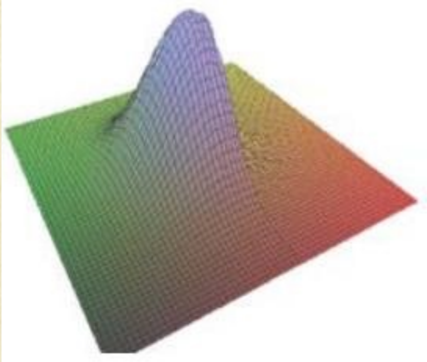
The weighting coefficient $w(i,j,k,l)$ depends on the product of a *domain kernel* (Figure 3.19c),

$$d(i,j,k,l) = \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2}\right), \tag{3.35}$$

and a data-dependent *range kernel* (Figure 3.19d),



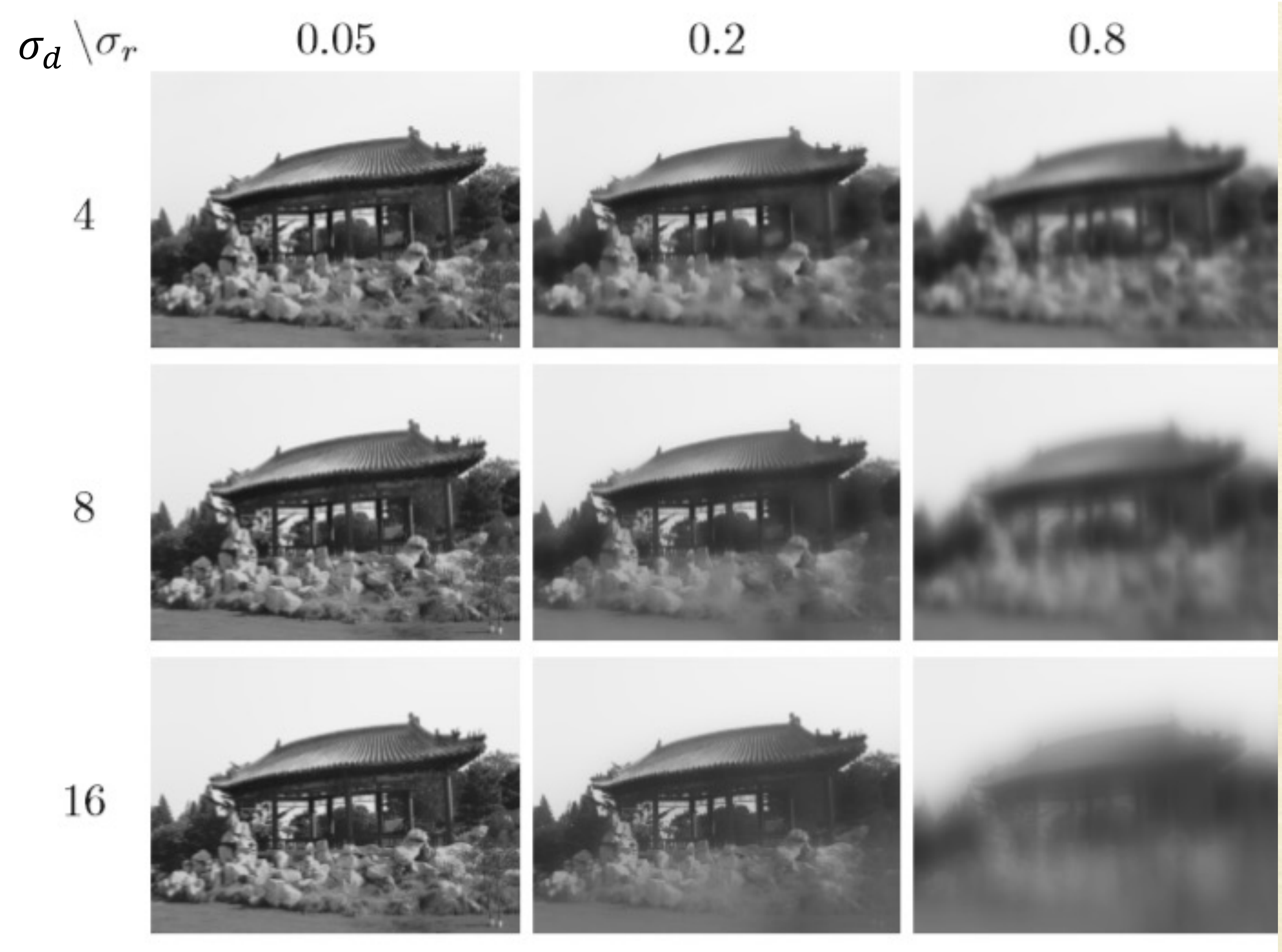
$$r(i,j,k,l) = \exp\left(-\frac{\|f(i,j) - f(k,l)\|^2}{2\sigma_r^2}\right).$$



When multiplied together, these yield the data-dependent *bilateral weight function*

$$w(i,j,k,l) = \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2} - \frac{\|f(i,j) - f(k,l)\|^2}{2\sigma_r^2}\right). \tag{3.37}$$

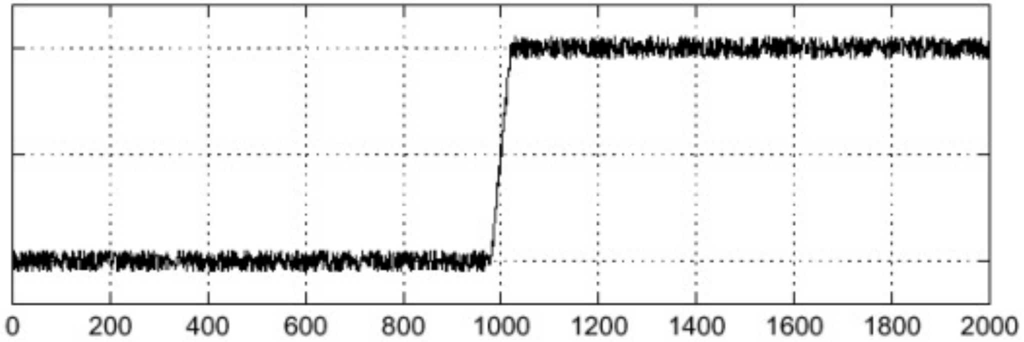
$$w(i,j,k,l) = \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2} - \frac{\|f(i,j) - f(k,l)\|^2}{2\sigma_r^2}\right)$$



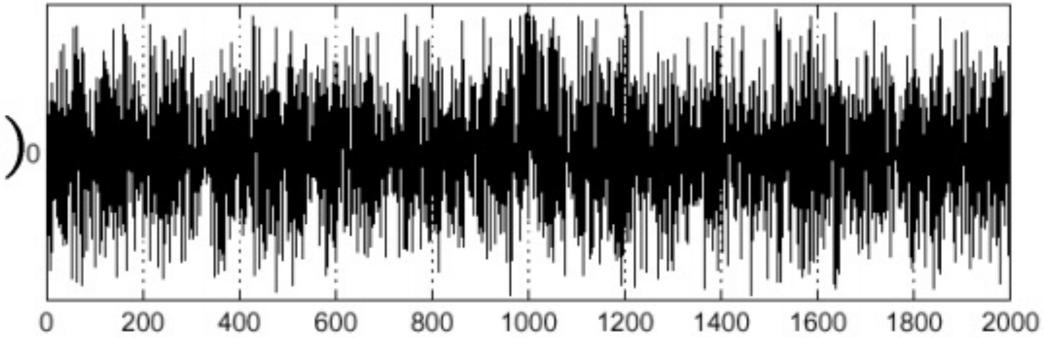
Edge Detection under Noise

Consider a single row or column of the image

- Plotting intensity as a function of position gives a signal



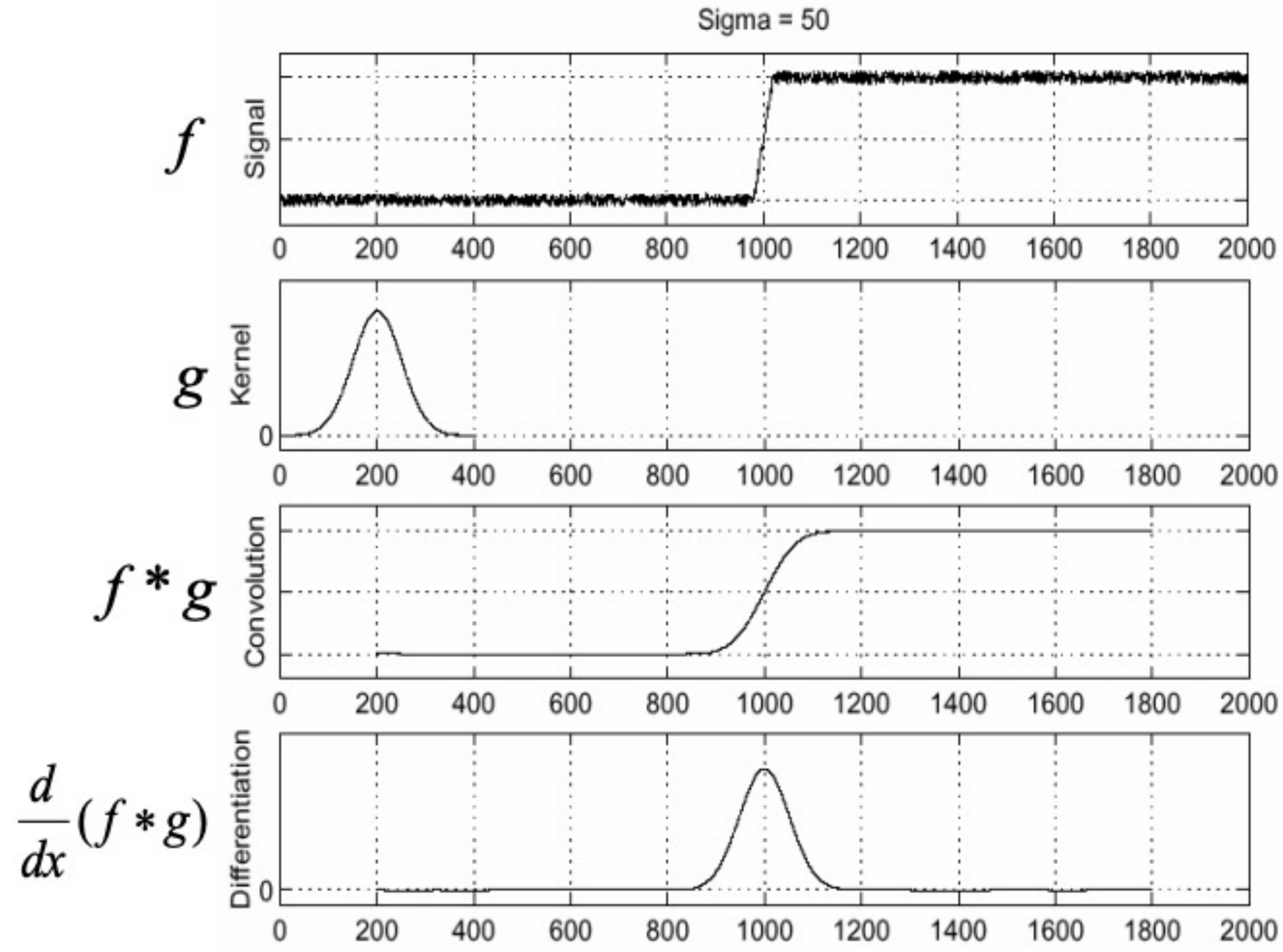
$\frac{d}{dx} f(x)$



Where is the edge?

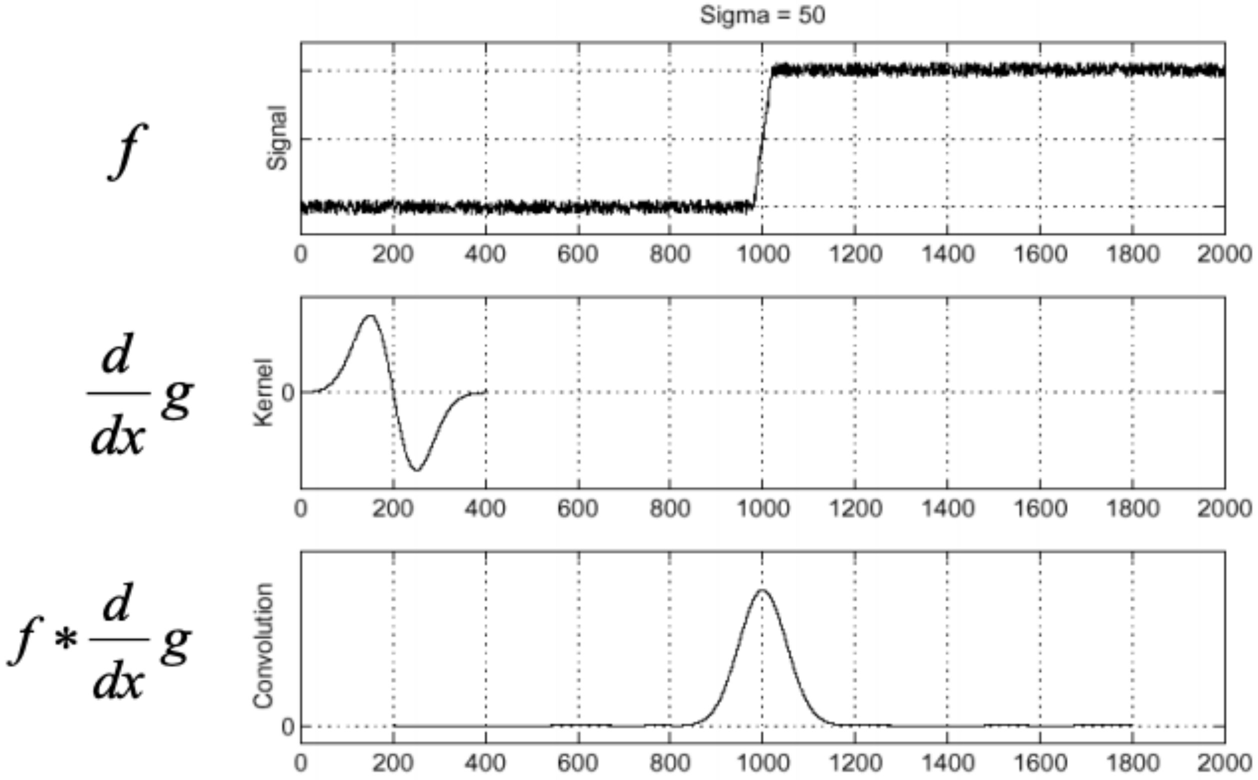
Source: S. Seitz

Solution: Smooth First



Derivative Theorem of Convolution

- Differentiation is convolution, and convolution is associative: $\frac{d}{dx}(f * g) = f * \frac{d}{dx} g$
- This saves us one operation:



Source: S. Seitz

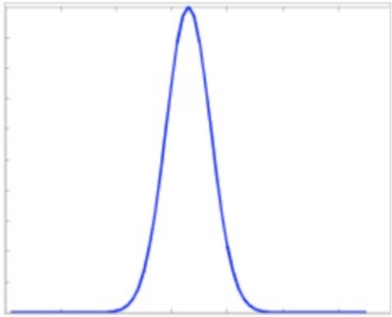
Other Important Filters

- Laplacian of Gaussian (LoG)

Robert Collins
CSE486

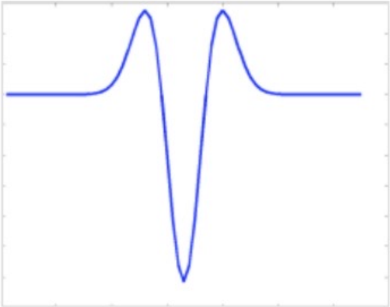
1D Gaussian and Derivatives

$$g(x) = e^{-\frac{x^2}{2\sigma^2}}$$



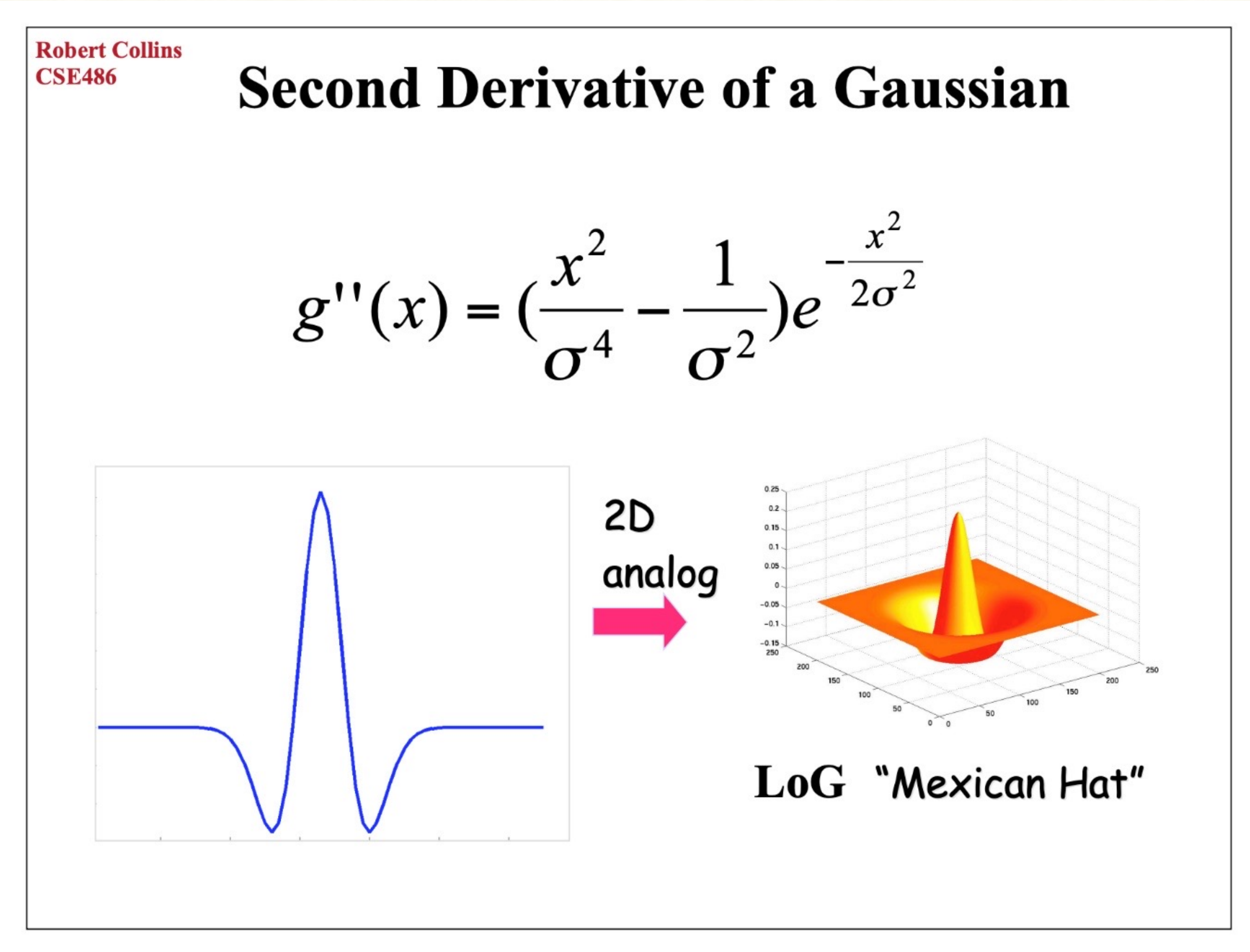
$$g'(x) = -\frac{1}{2\sigma^2} 2x e^{-\frac{x^2}{2\sigma^2}} = -\frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}}$$

$$g''(x) = \left(\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right) e^{-\frac{x^2}{2\sigma^2}}$$



Other Important Filters

- Laplacian of Gaussian (in 2D)



Other Important Filters

- Laplacian of Gaussian (in 2D)
- Difference of Gaussian (DoG)

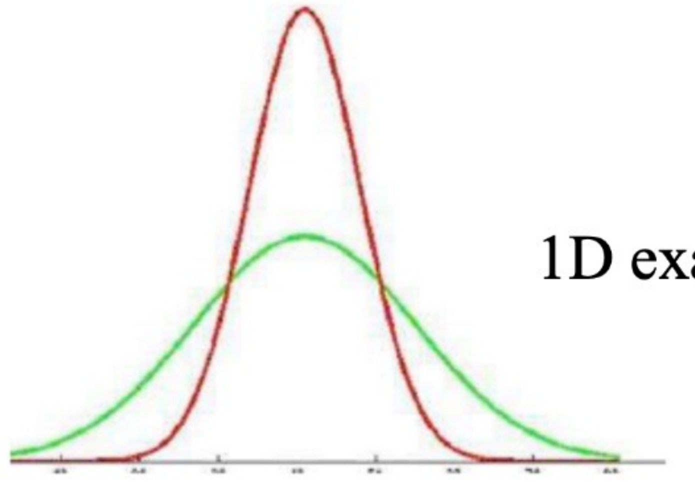
Robert Collins
CSE486

Efficient Implementation
Approximating LoG with DoG

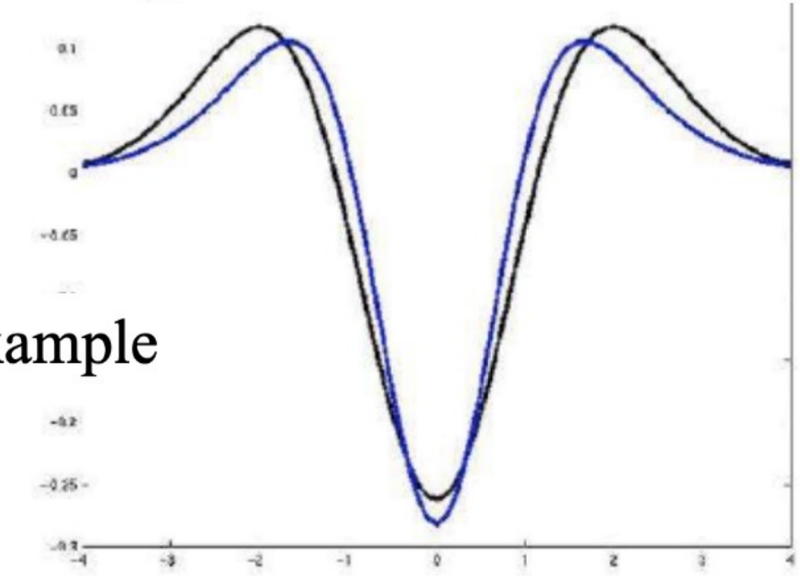
LoG can be approximate by a Difference of two Gaussians (DoG) at different scales

$$\nabla^2 G_\sigma \approx G_{\sigma_1} - G_{\sigma_2}$$

Best approximation when:
$\sigma_1 = \frac{\sigma}{\sqrt{2}}$, $\sigma_2 = \sqrt{2}\sigma$



1D example



Questions?