CS4670/5670: Computer Vision

Kavita Bala

Lecture 2: Filtering

Announcements

- PA 1 will be out early next week (Monday)
  - due in 2 weeks
  - to be done in groups of two – please form your groups ASAP
- We will grade in demo sessions

Linear filtering

- One simple version: linear filtering
  - Replace each pixel by a linear combination (a weighted sum) of its neighbors
  - Simple, but powerful
  - Cross-correlation, convolution
- The prescription for the linear combination is called the "kernel" (or "mask", "filter")

10	5	3
4	6	1
1	1	8

Local image data

0	0	0
0	0.5	0
0	1	0.5

kernel

	8	

Modified image data

Source: L. Zhang

Filter Properties

- Linearity
  - Weighted sum of original pixel values
  - Use same set of weights at each point
  - $S[f + g] = S[f] + S[g]$
  - $S[p\, f + q\, g] = p\, S[f] + q\, S[g]$
- Shift-invariance
  - If $f[m,n] \overset{S}{\Rightarrow} g[m,n]$, then $f[m-p,n-q] \overset{S}{\Rightarrow} g[m-p, n-q]$
  - The operator behaves the same everywhere

Cross-correlation

Let F be the image, H be the kernel (of size 2*k*+1 x 2*k*+1), and G be the output image

$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v]F[i+u,j+v]$$

This is called a **cross-correlation** operation:

$$G = H \otimes F$$

- Can think of as a "dot product" between local neighborhood and kernel for each pixel

Convolution

- Same as cross-correlation, except that the kernel is "flipped" (horizontally and vertically)

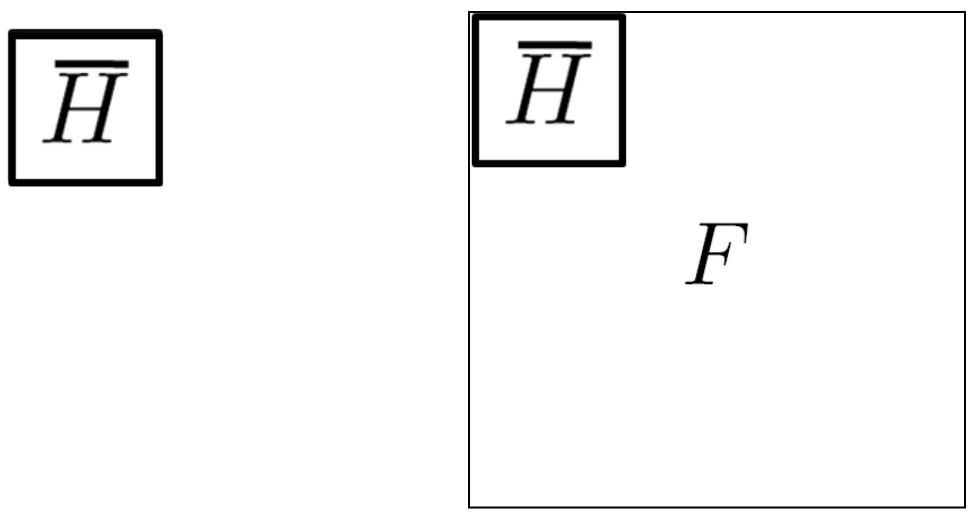
$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v]F[i-u,j-v]$$

This is called a **convolution** operation:

$$G = H * F$$

- Convolution is **commutative** and **associative**

Convolution



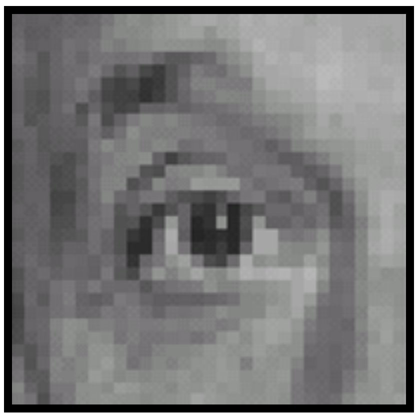
Adapted from F. Durand

Pseudo-code

```
function convolve(sequence a, sequence b, int r, int i )
    s = 0
    for j = −r to r
        s = s + a[j]b[i − j]
    return s
```

CS 4620 and 5625 cover topic in detail

Linear filters: examples

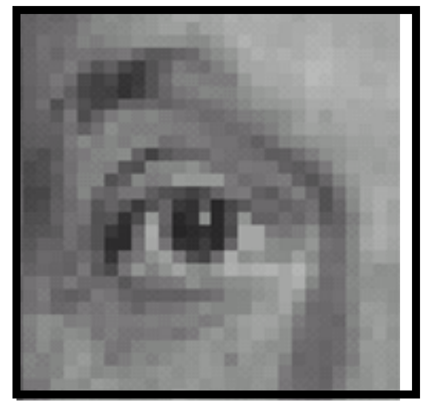


Original

*

0	0	0
1	0	0
0	0	0

=

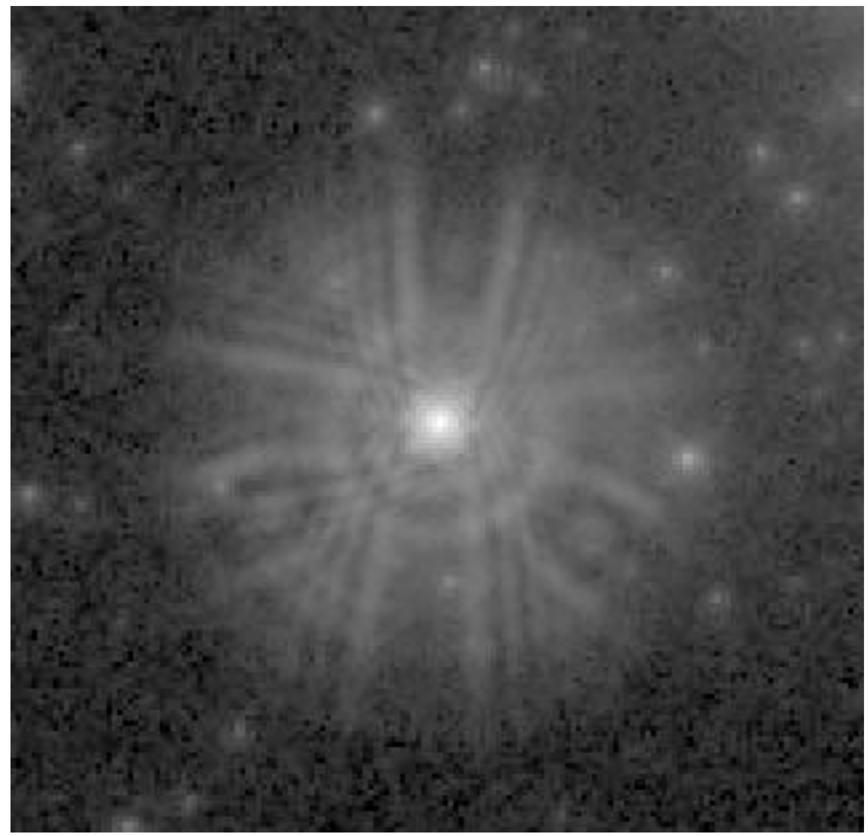


Shifted left
By 1 pixel

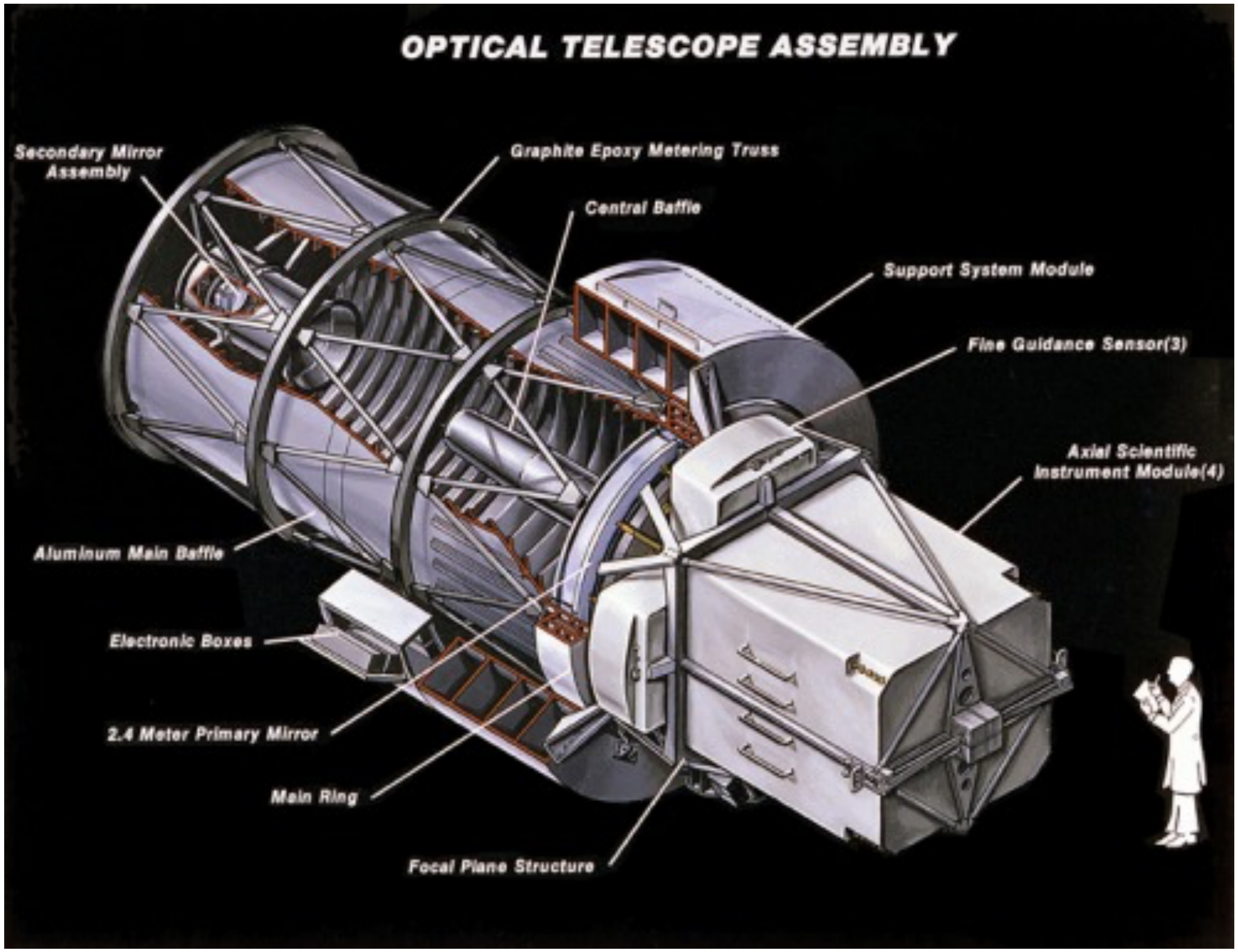
Source: D. Lowe

Convolution

- Point spread function, impulse response function

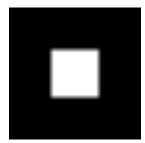


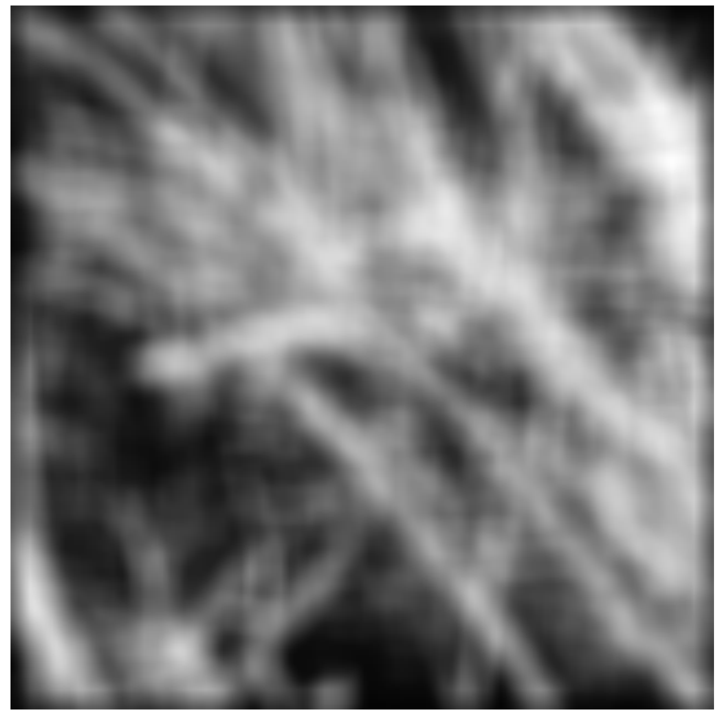
PSF of Hubble Telescope



http://aetherforce.com/wp-content/uploads/2013/10/8707342.jpg

Smoothing with box filter

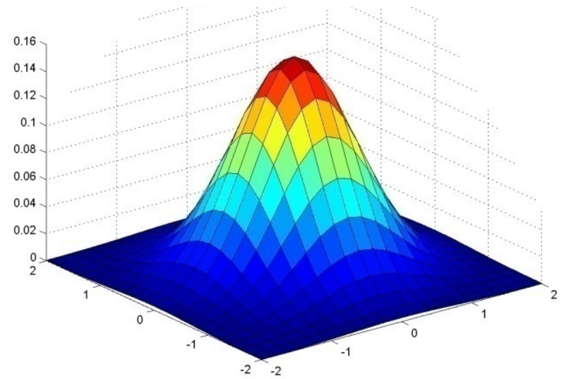


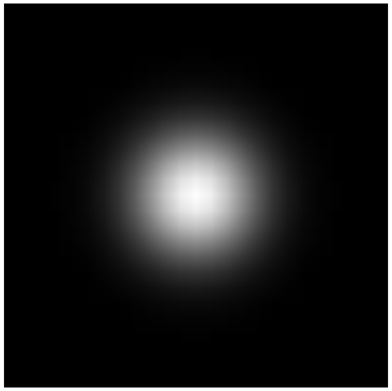


$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Source: D. Forsyth

Gaussian Kernel



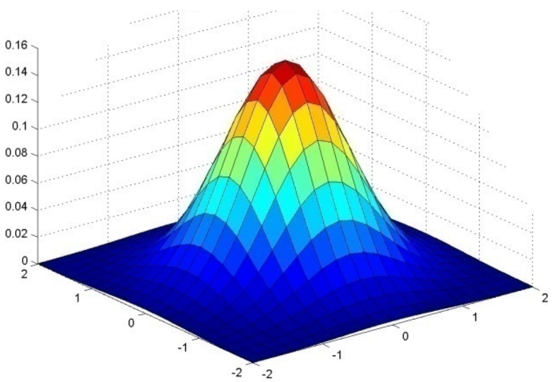


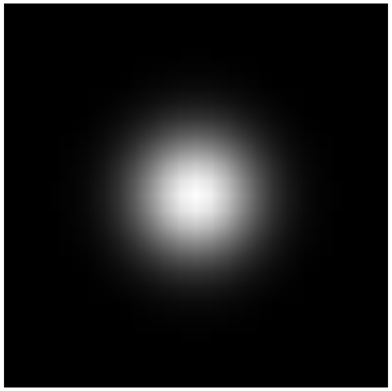
$$G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Source: C. Rasmussen

Gaussian Kernel

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



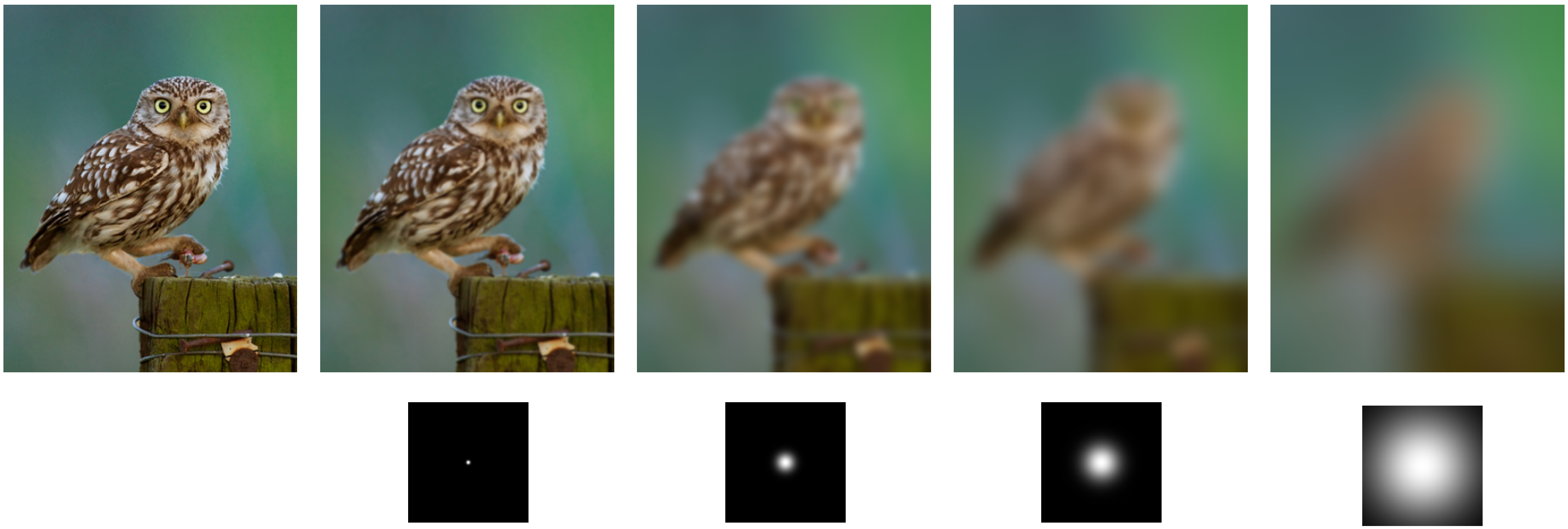


0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

5 x 5, σ = 1

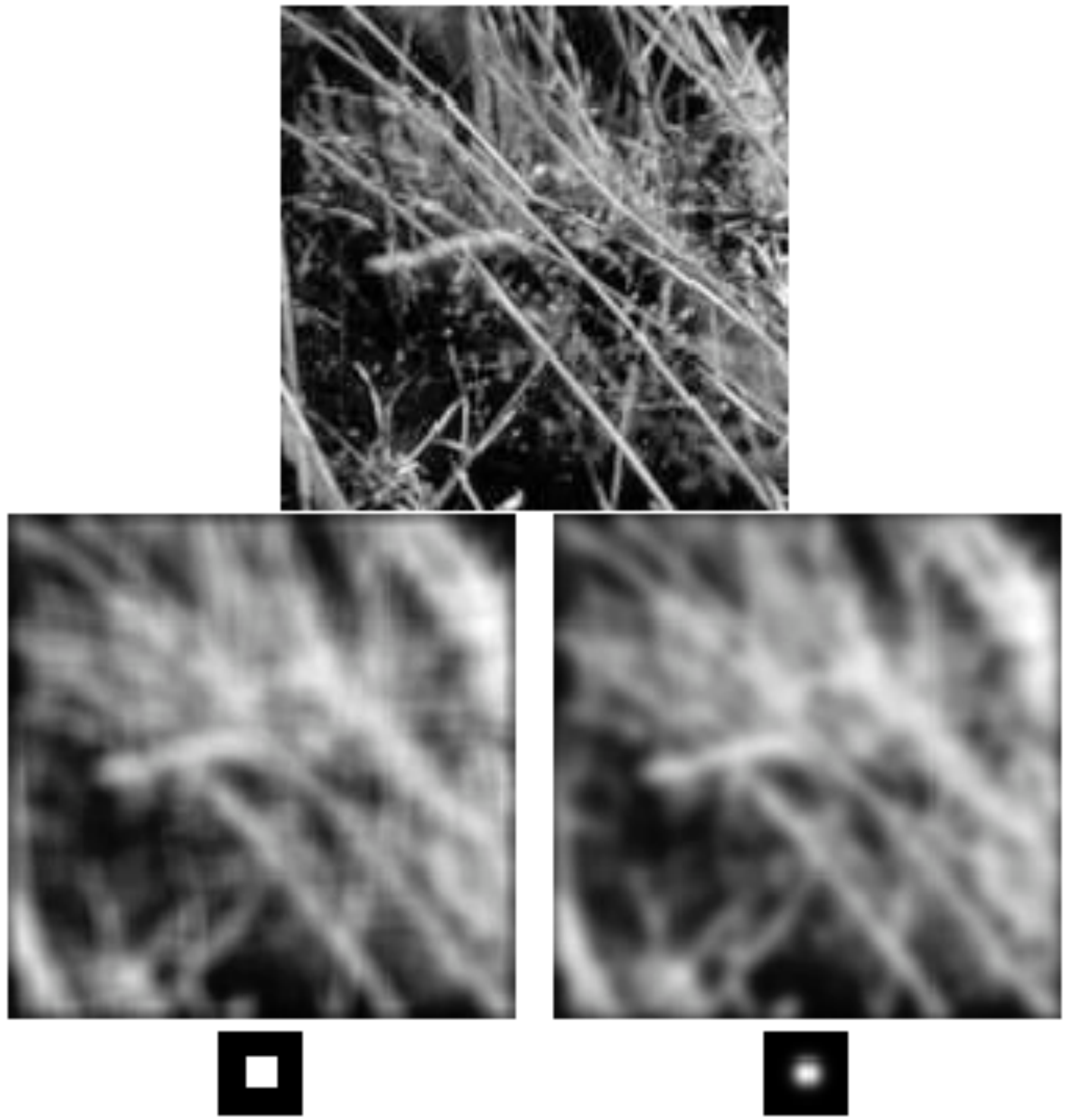
Source: C. Rasmussen

Gaussian filters



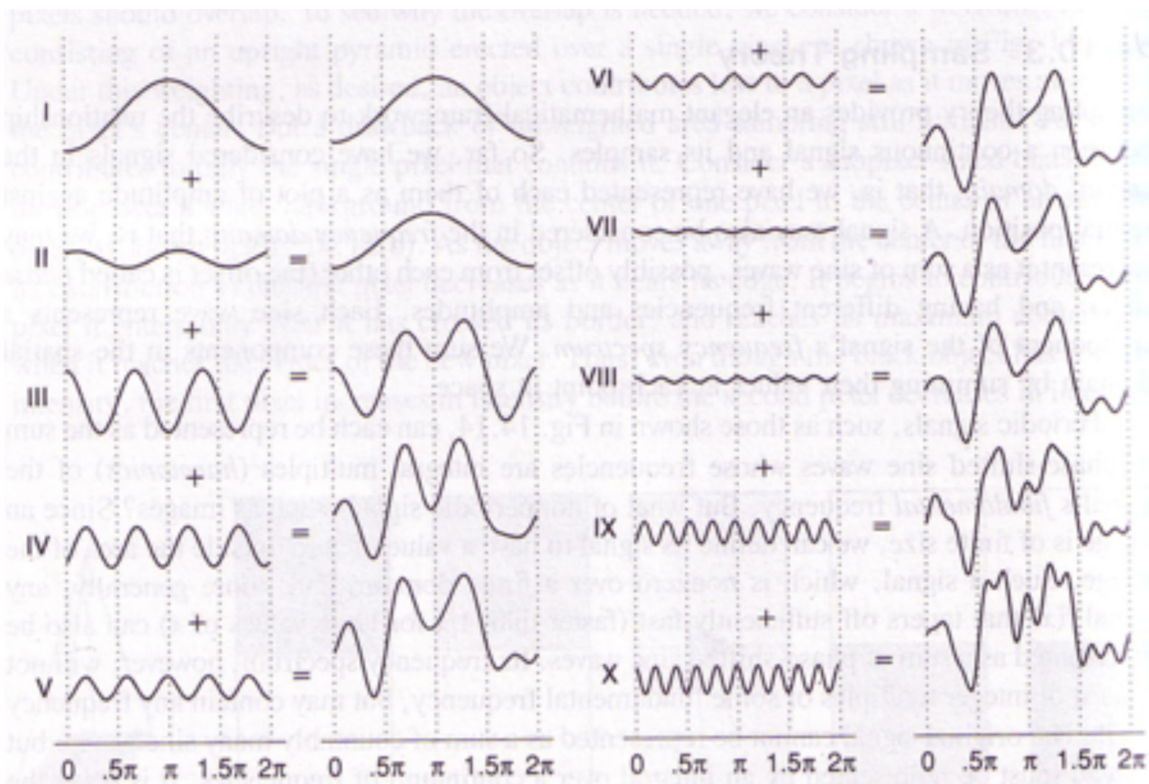
σ = 1 pixel　σ = 5 pixels　σ = 10 pixels　σ = 30 pixels

Mean vs. Gaussian filtering

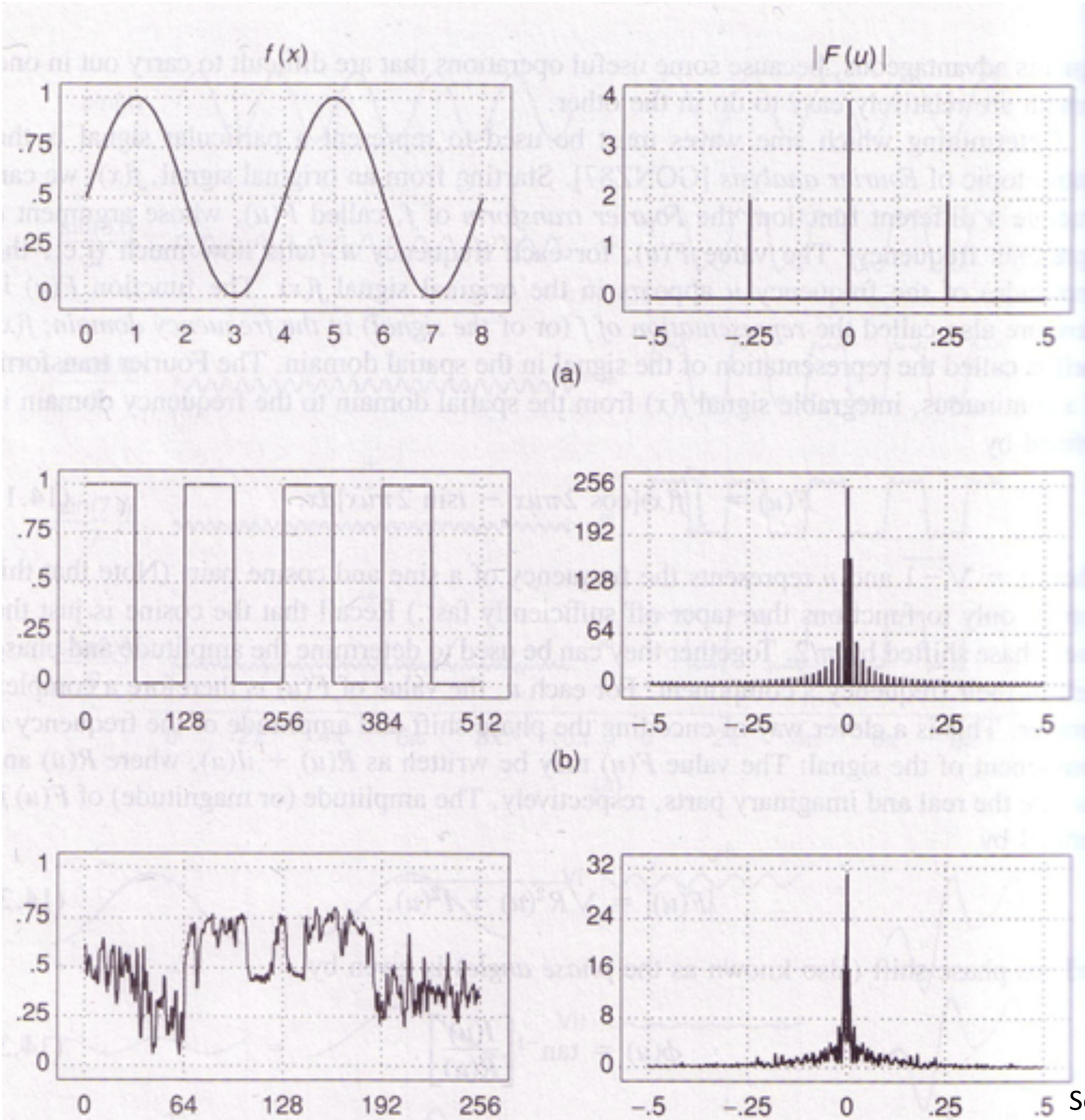


Detour: Fourier Analysis

- Every signal has some frequency
- Fourier analysis finds frequencies of a signal
  - Sum of sine/cosine waves



Source: Foley, van Dam



Source: Foley, van Dam

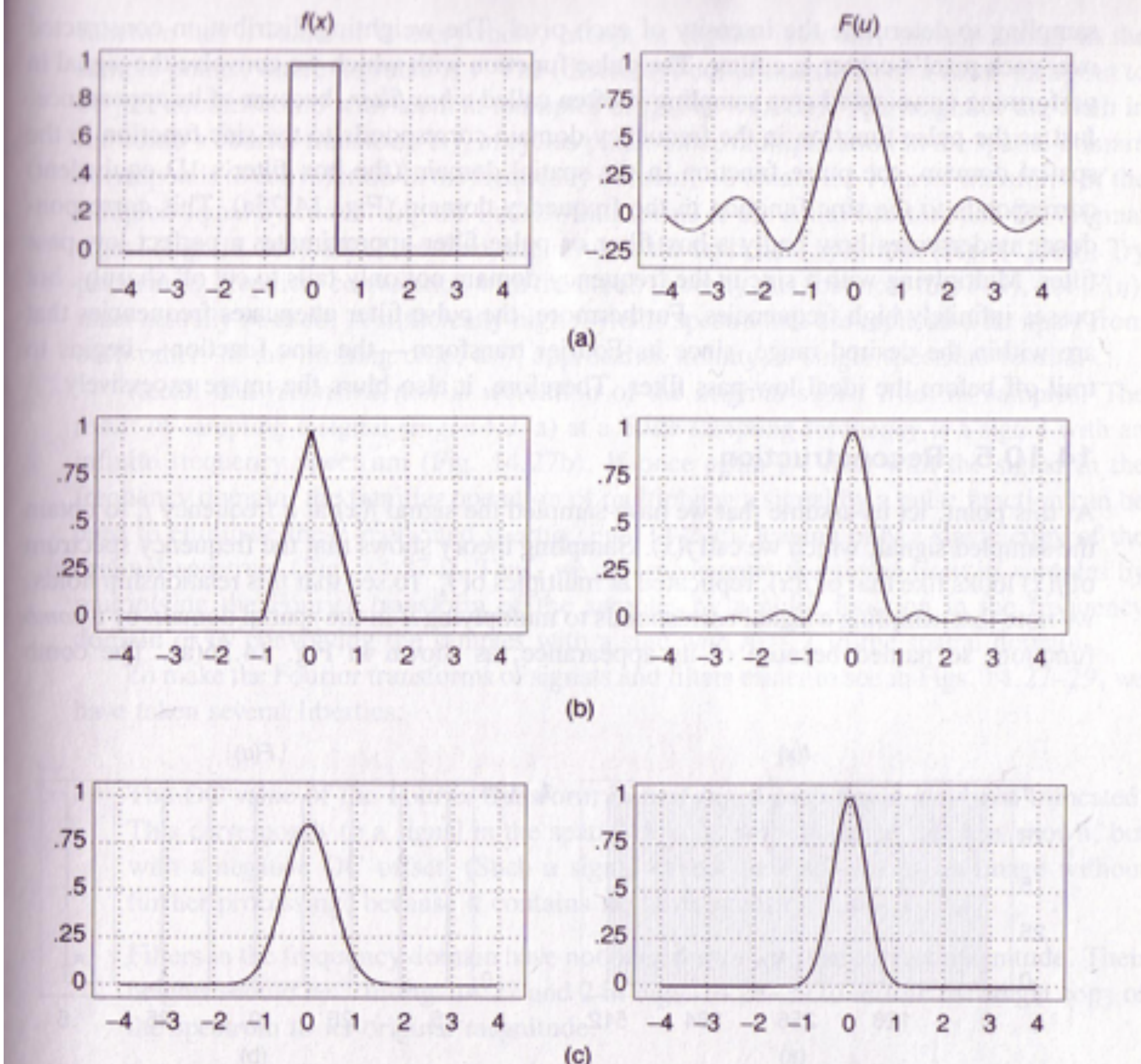


Fig. 14.25 Filters in spatial and frequency domains. (a) Pulse—sinc. (b) Triangle—sinc^2. (c) Gaussian—Gaussian. (Courtesy of George Wolberg, Columbia University.)

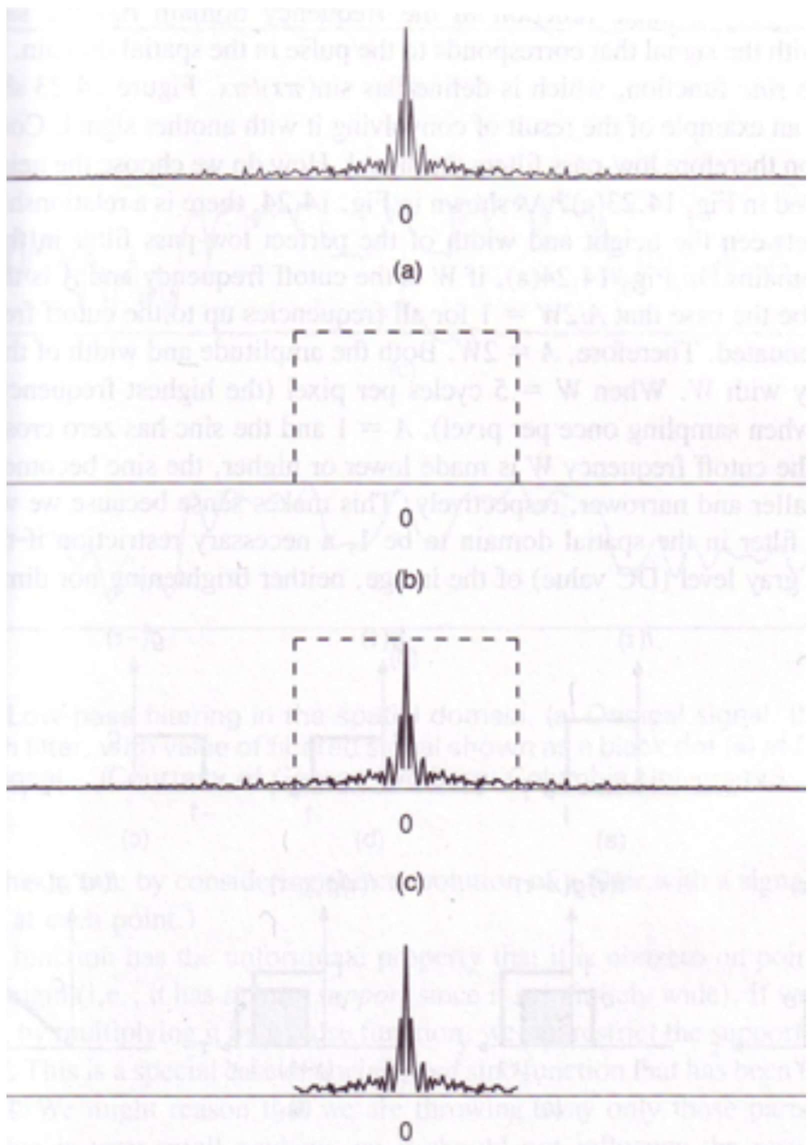
Source: Foley, van Dam

Convolution is special

- Convolution in image space
  - Multiplication in Fourier space

- Box filter -> sinc in Fourier space
- Gaussian filter -> Gaussian in Fourier space

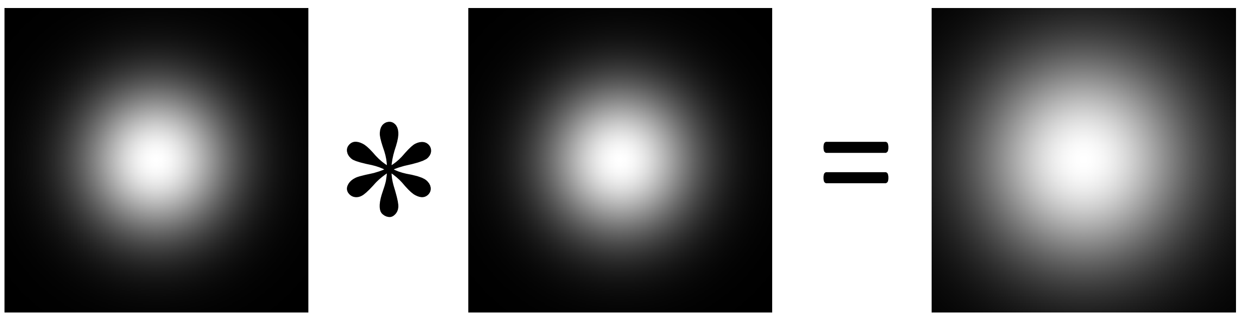
Low pass filtering



Source: Foley, van Dam

Gaussian filter

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



  – Convolving twice with Gaussian kernel of width σ = convolving once with kernel of width $\sigma\sqrt{2}$

Source: K. Grauman

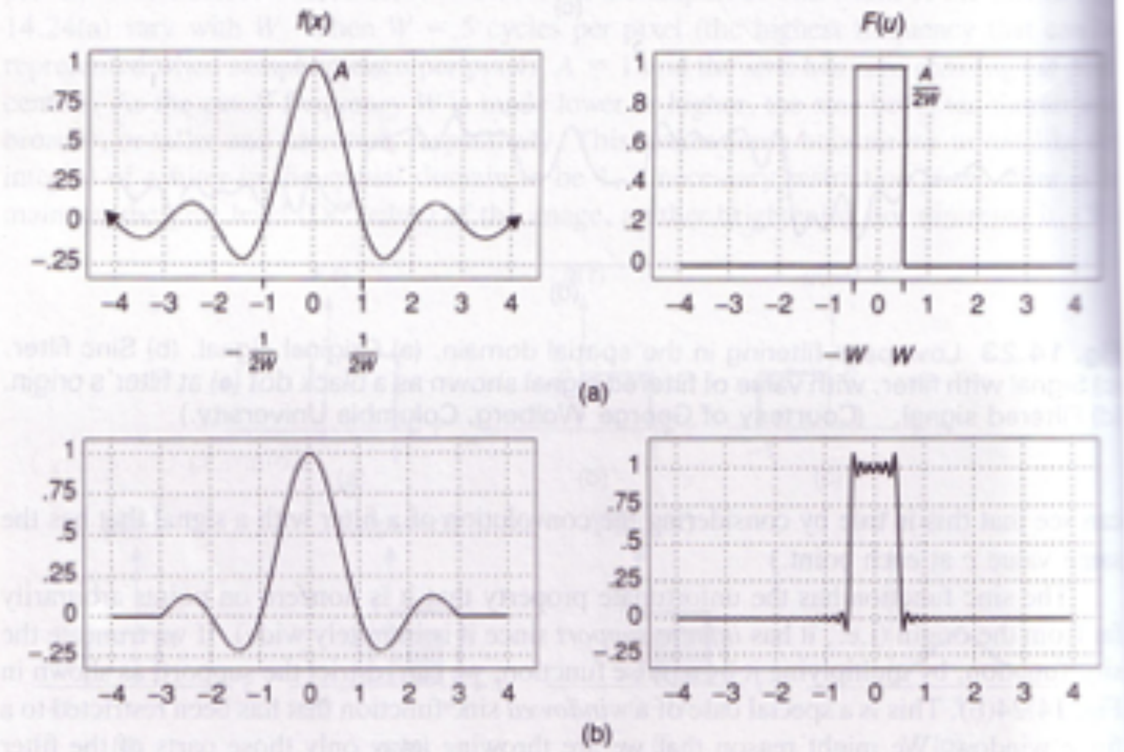


Fig. 14.24 (a) Sinc in spatial domain corresponds to pulse in frequency domain. (b) Truncated sinc in spatial domain corresponds to ringing pulse in frequency domain. (Courtesy of George Wolberg, Columbia University.)

Source: Foley, van Dam

Sharpening

- What does blurring take away?

–

=

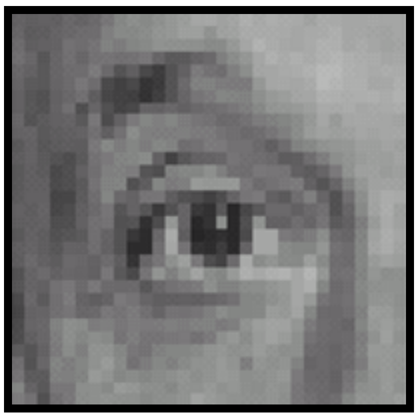
Let’s add it back:

+ α

=

Source: S. Lazebnik

Linear filters: examples



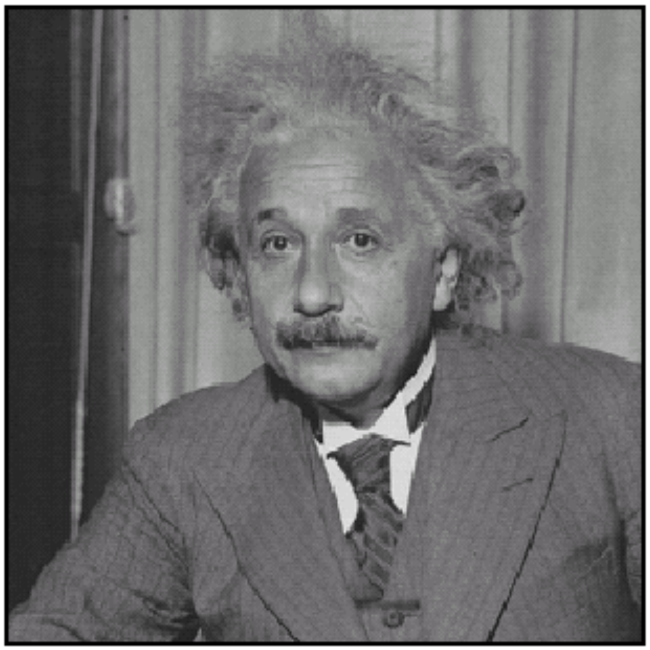
Original

$$* \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \right) =$$

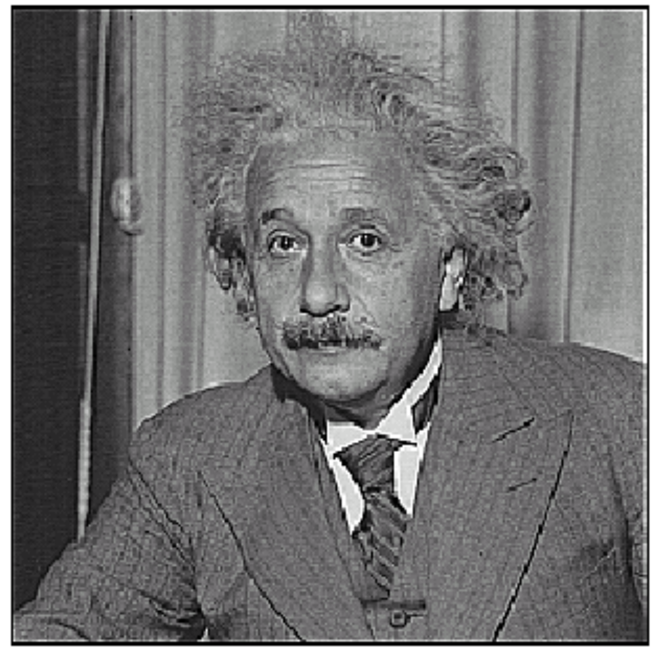
Sharpening filter
(accentuates edges)

Source: D. Lowe

Sharpening



before



after

Source: D. Lowe

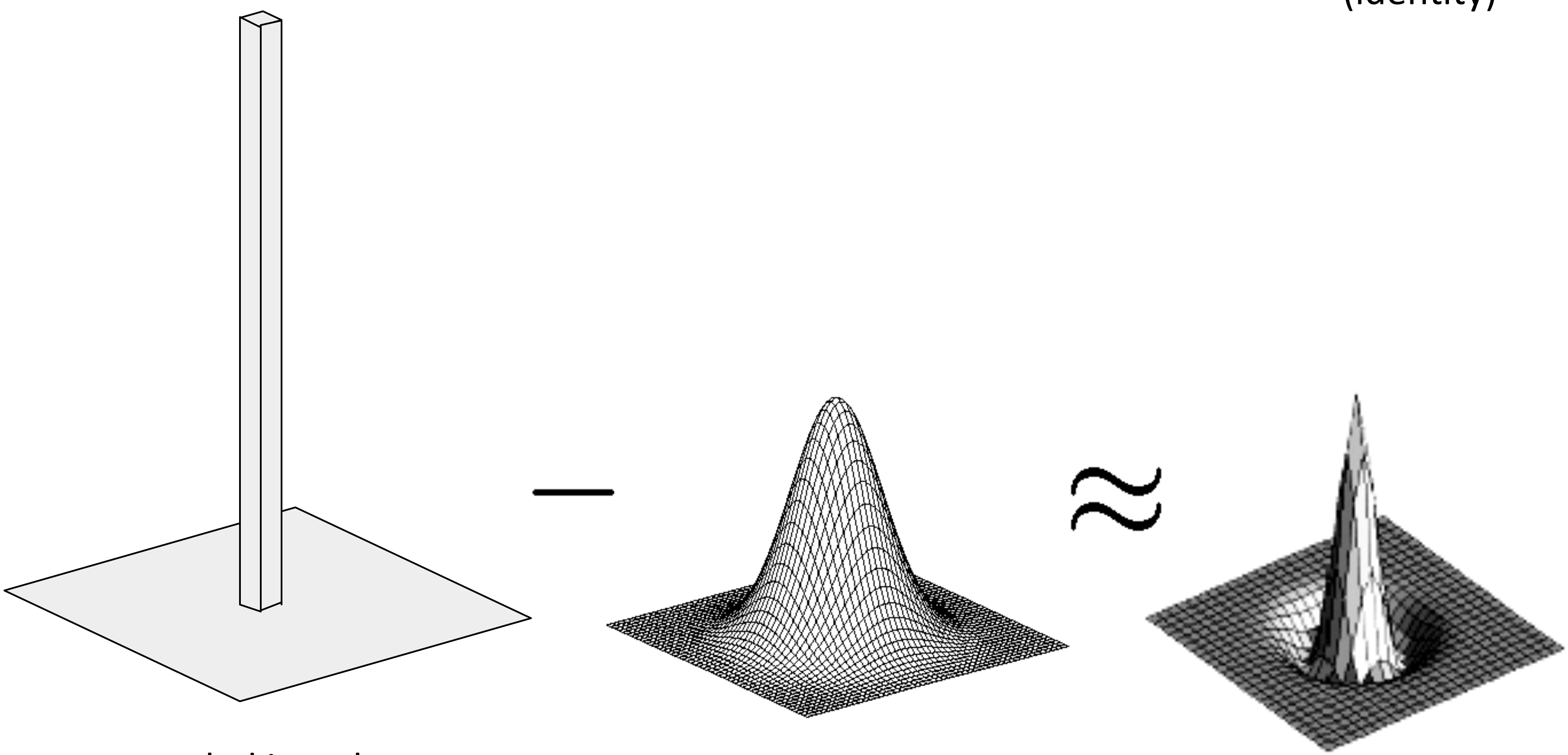
Sharpen filter

$$F + \alpha\,(F - F * H) = (1+\alpha)\,F - \alpha\,(F * H) = F * ([1+\alpha]e - \alpha H)$$

image (F)

blurred image (F * H)

unit impulse (identity) (e)



scaled impulse − Gaussian ≈ Laplacian of Gaussian

Sharpen filter

“Optical” Convolution

Camera shake

Source: Fergus, *et al.* “Removing Camera Shake from a Single Photograph”, SIGGRAPH 2006

Bokeh: Blur in out-of-focus regions of an image.

Source: http://lullaby.homepage.dk/diy-camera/bokeh.html