CS6670: Computer Vision
Noah Snavely

Lecture 1: Images and image filtering

Hybrid Images, Oliva et al., http://cvcl.mit.edu/hybridimage.htm

CS6670: Computer Vision
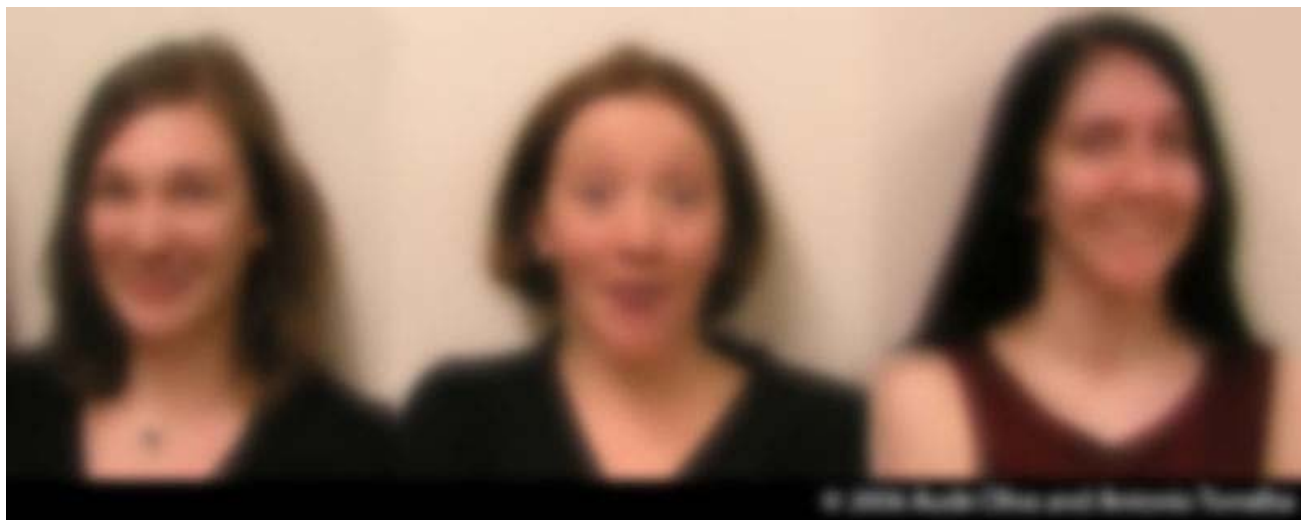Noah Snavely

Lecture 1: Images and image filtering

Hybrid Images, Oliva et al., http://cvcl.mit.edu/hybridimage.htm

Reading

- Szeliski, Chapter 3.1-3.2

What is an image?

Illumination (energy) source

Imaging system

(Internal) image plane

Scene element

Digital Camera

We'll focus on these in this class

(More on this process later)

Zonule fibers

Vitt

Fovea

Visual axis

Lens

Optic axis

Disk

liary body

Retina

The Eye
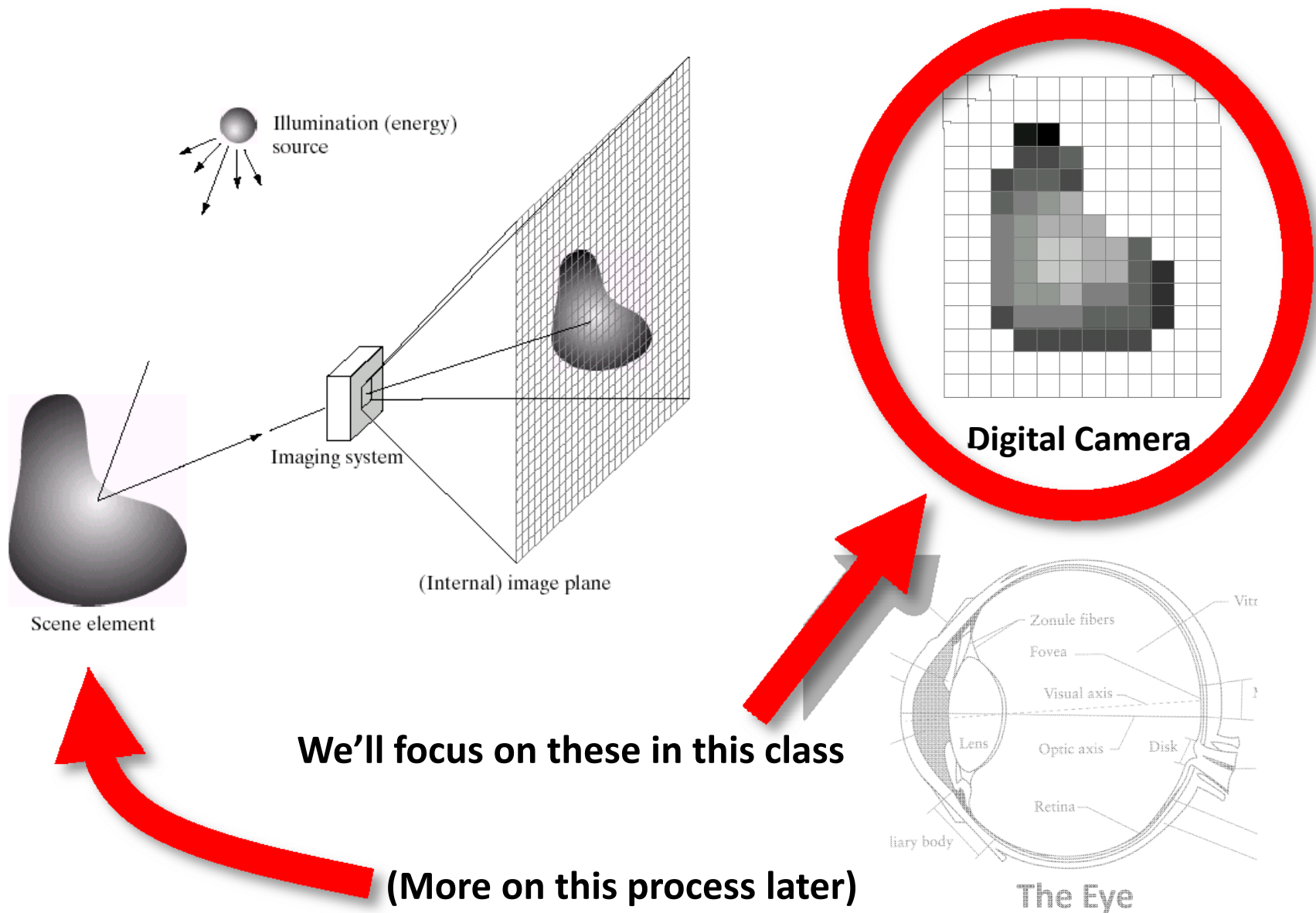
What is an image?

- A grid of intensity values
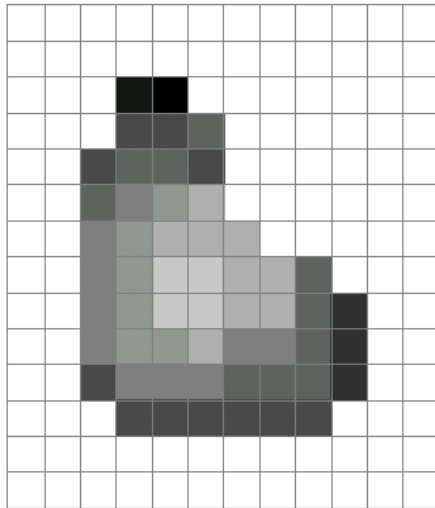
| 255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	20	0	255	255	255	255	255	255	255
255	255	255	75	75	75	255	255	255	255	255	255
255	255	75	95	95	75	255	255	255	255	255	255
255	255	96	127	145	175	255	255	255	255	255	255
255	255	127	145	175	175	175	255	255	255	255	255
255	255	127	145	200	200	175	175	95	255	255	255
255	255	127	145	200	200	175	175	95	47	255	255
255	255	127	145	145	175	127	127	95	47	255	255
255	255	74	127	127	127	95	95	95	47	255	255
255	255	255	74	74	74	74	74	74	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255

(common to use one byte per value: 0 = black, 255 = white)

What is an image?

- We can think of a (grayscale) image as a **function**, f, from R^2 to R:
  - $f(x,y)$ gives the **intensity** at position (x,y)

snoop
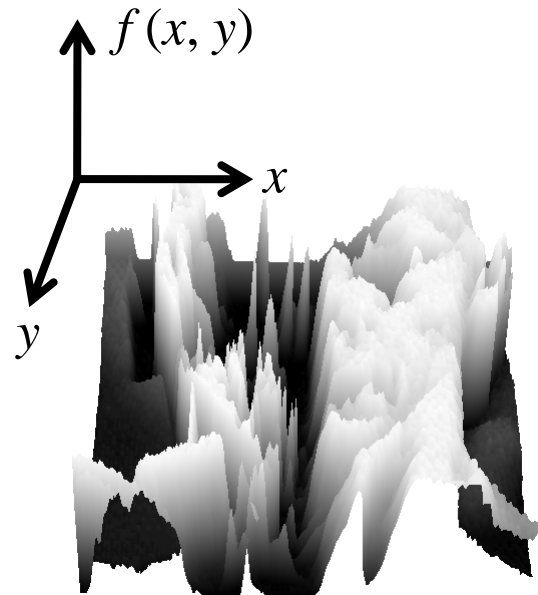
3D view

  - A **digital** image is a discrete (**sampled**, **quantized**) version of this function

Image transformations

- As with any function, we can apply operators to an image

$g\,(x,y) = f\,(x,y) + 20$ $g\,(x,y) = f\,(-x,y)$

- We'll talk about a special kind of operator, *convolution* (linear filtering)

Question: Noise reduction

- Given a camera and a still scene, how can you reduce noise?

Take lots of images and average them!

What's the next best thing?

Image filtering

- Modify the pixels in an image based on some function of a local neighborhood of each pixel

10	5	3
4	5	1
1	1	7

Some function

→

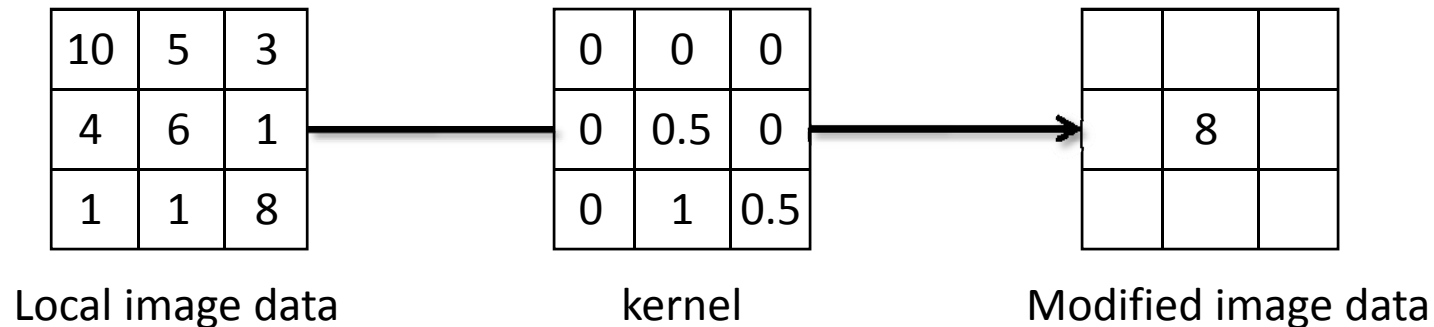|		
	7	

Local image data

Modified image data

Linear filtering

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

10	5	3
4	6	1
1	1	8

0	0	0
0	0.5	0
0	1	0.5

	8	

Local image data   kernel   Modified image data

Cross-correlation

Let F be the image, H be the kernel (of size 2k+1 x 2k+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:

$$G = H \otimes F$$

Convolution

- Same as cross-correlation, except that the kernel is "flipped" (horizontally and vertically)
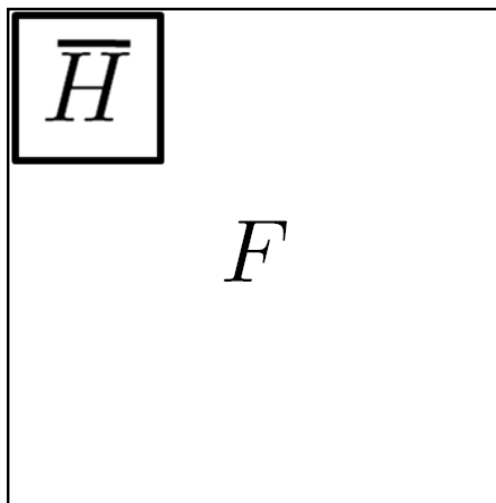
$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v]F[i-u,j-v]$$
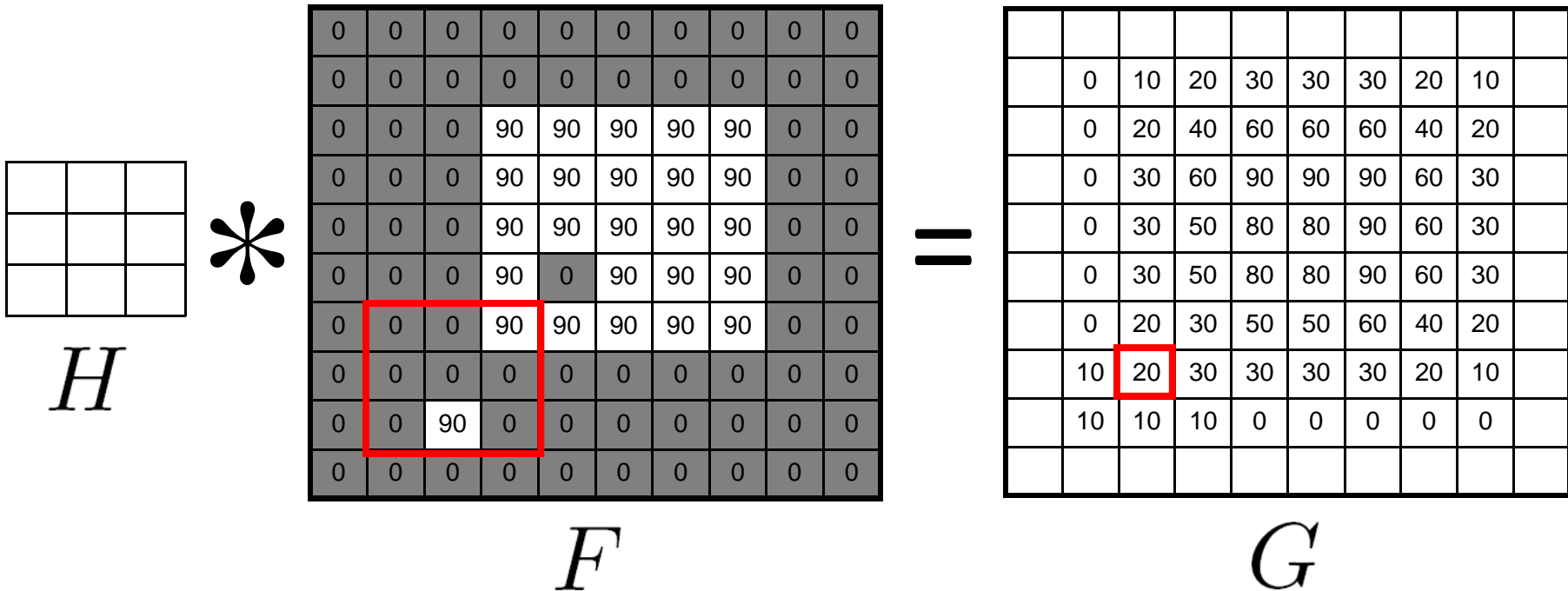
This is called a **convolution** operation:

$$G = H * F$$

- Convolution / cross-correlation are **commutative** and **associative**

Convolution

$\overline{H}$

$\overline{H}$ F

Mean filtering

H * F = G

H (3×3 kernel, empty)

F:

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

G:

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	60	90	90	90	60	30	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	10	20	30	30	30	30	20	10	
	10	10	10	0	0	0	0	0	

Linear filters: examples

Original

$*$

0	0	0
0	1	0
0	0	0

$=$

Identical image

Linear filters: examples

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Original

*   =

Shifted left
By 1 pixel

Linear filters: examples

Original

$$\ast \quad \frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \quad =$$
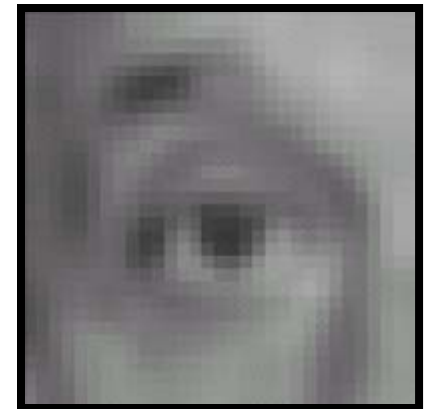
Blur (with a mean filter)
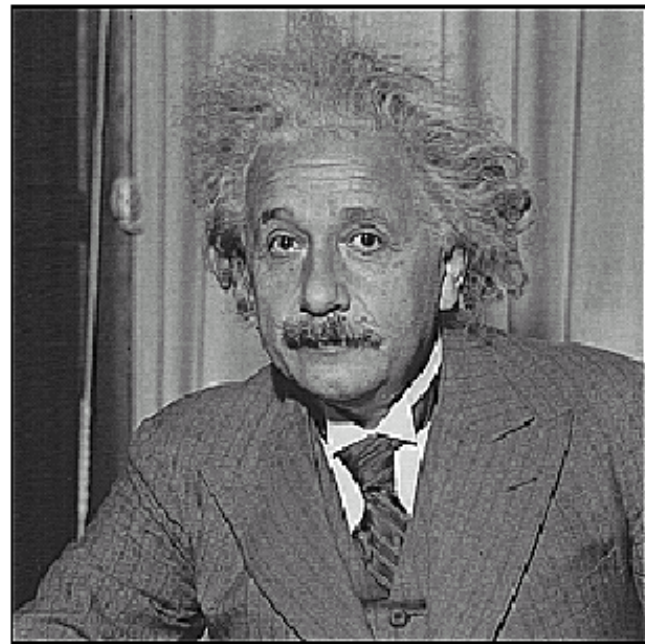
Linear filters: examples

Original

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

Sharpening filter
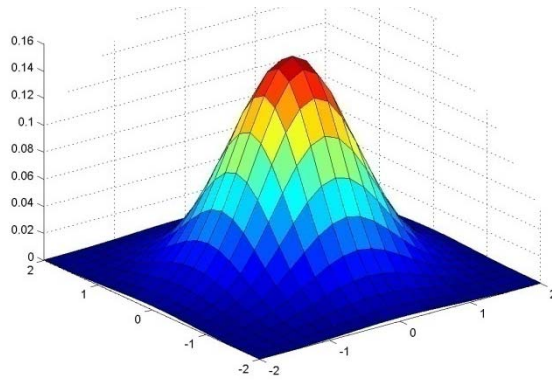(accentuates edges)

Sharpening
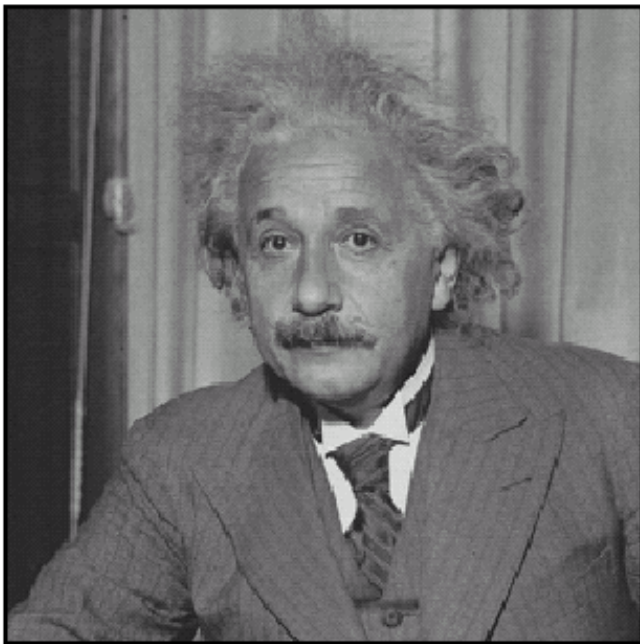
before   after

Smoothing with box filter revisited

Gaussian Kernel

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

Source: C. Rasmussen
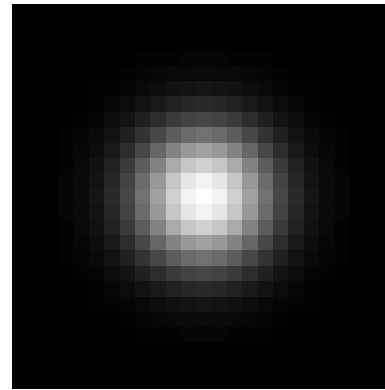
Mean vs. Gaussian filtering
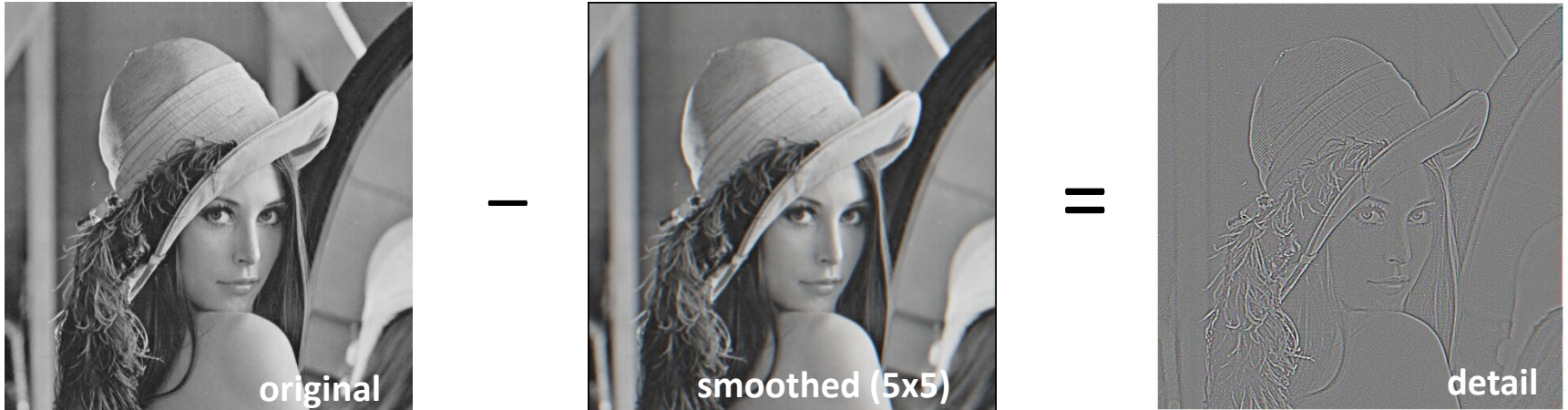
Gaussian filter

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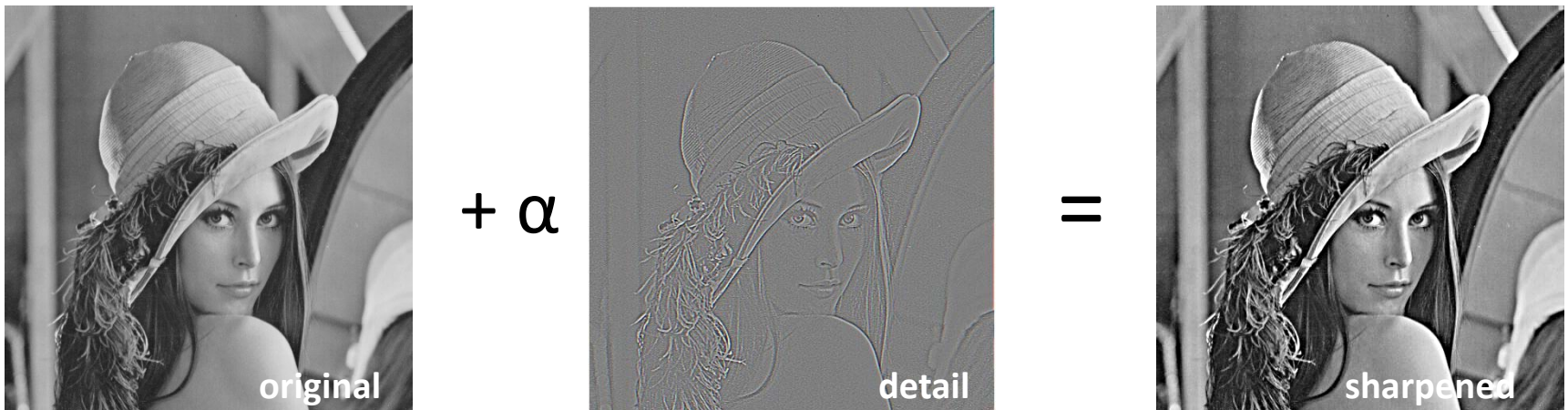
- Convolution with self is another Gaussian

 $*$ =

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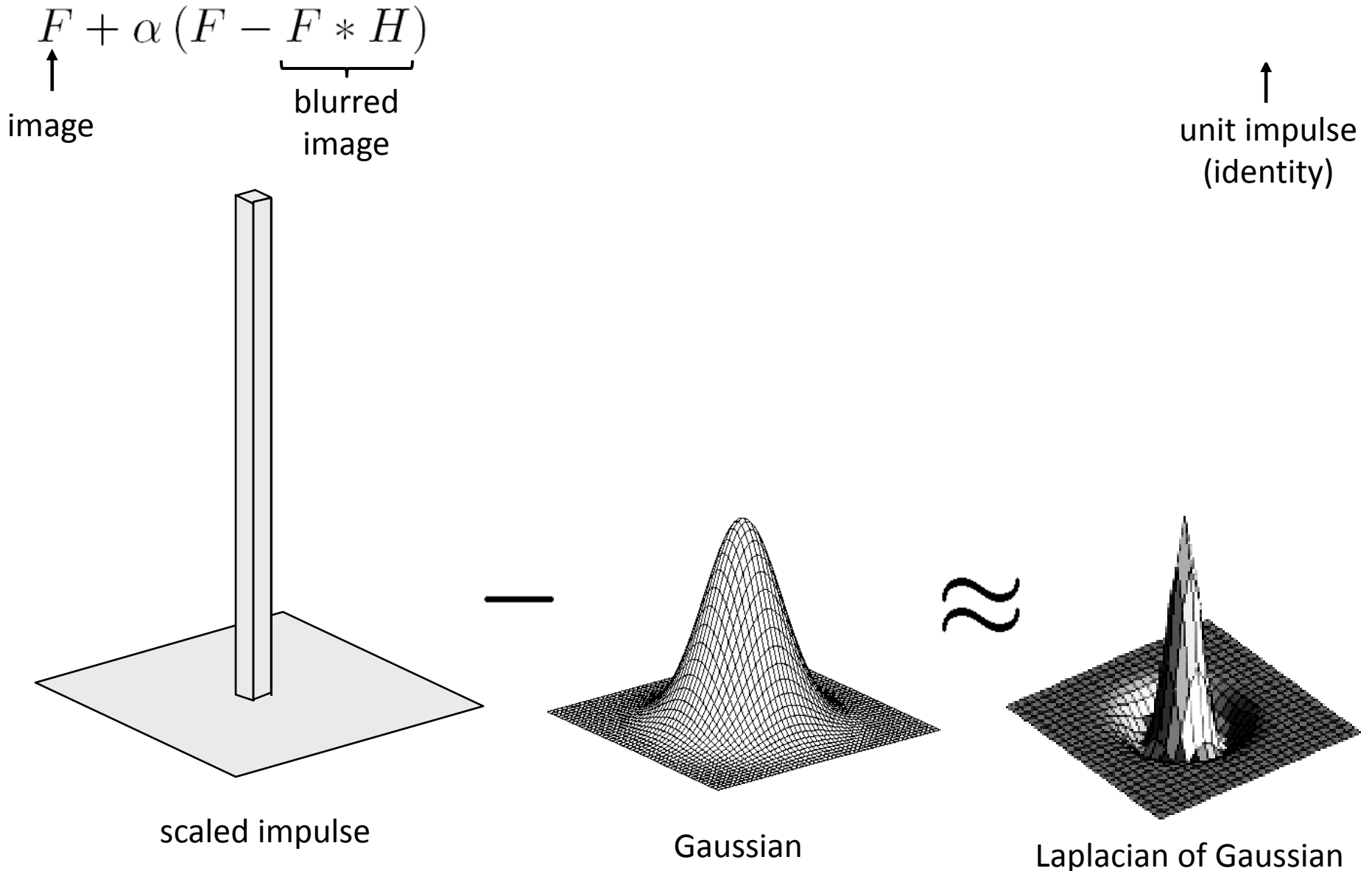
Sharpening revisited

- What does blurring take away?

original − smoothed (5x5) = detail

Let's add it back:

original + α detail = sharpened

Source: S. Lazebnik

Sharpen filter

$$F + \alpha \left(F - F * H\right)$$

↑
image

blurred
image

↑
unit impulse
(identity)

$-$ $\approx$

scaled impulse

Gaussian

Laplacian of Gaussian

Sharpen filter

unfiltered

filtered

Convolution in the real world

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

Questions?

- For next time:
  - Read Szeliski, Chapters 1, 3.1-3.2

- Next time:
  - See you on Tuesday, Sept. 8!
  - Feature and edge detection