Neighborhood Operations (cont'd)

CS 355: Introduction to Graphics and Image Processing

Spatial Filtering

45	60	98	127	132	133	137	133
46	65	98	123	126	128	131	133
47	65	96	115	119	123	135	137
47	63	91	107	113	122	138	134
50	59	80	97	110	123	133	134
49	53	68	83	97	113	128	133
50	50	58	70	84	102	116	126
50	50	52	58	69	86	101	120

*

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

=

69	95	116	125	129	132
68	92	110	120	126	132
66	86	104	114	124	132
62	78	94	108	120	129
57	69	83	98	112	124
53	60	71	85	100	114

notation for convolution operator

$$I' = I * w$$

Negative Weights

- Detecting edges or sharpening images involve finding or accentuating differences
- Requires mix of *positive and negative weights*

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

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

1	-2	1
1	-2	1
1	-2	1

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

Unsharp Masking

- Originated in analog darkrooms
- Key idea: mask (subtract) out the blur
- Procedure:
  - Blur more (yes, really!)
  - Subtract from original
  - Multiply by some fraction
  - Add back to the original

Unsharp Masking

- Mathematically: $I' = I + \alpha(I - \overline{I})$
  - Input image I
  - Blurred input image $\overline{I}$
  - Weighting (controls sharpening) α
  - Output image I'

Unsharp Masking

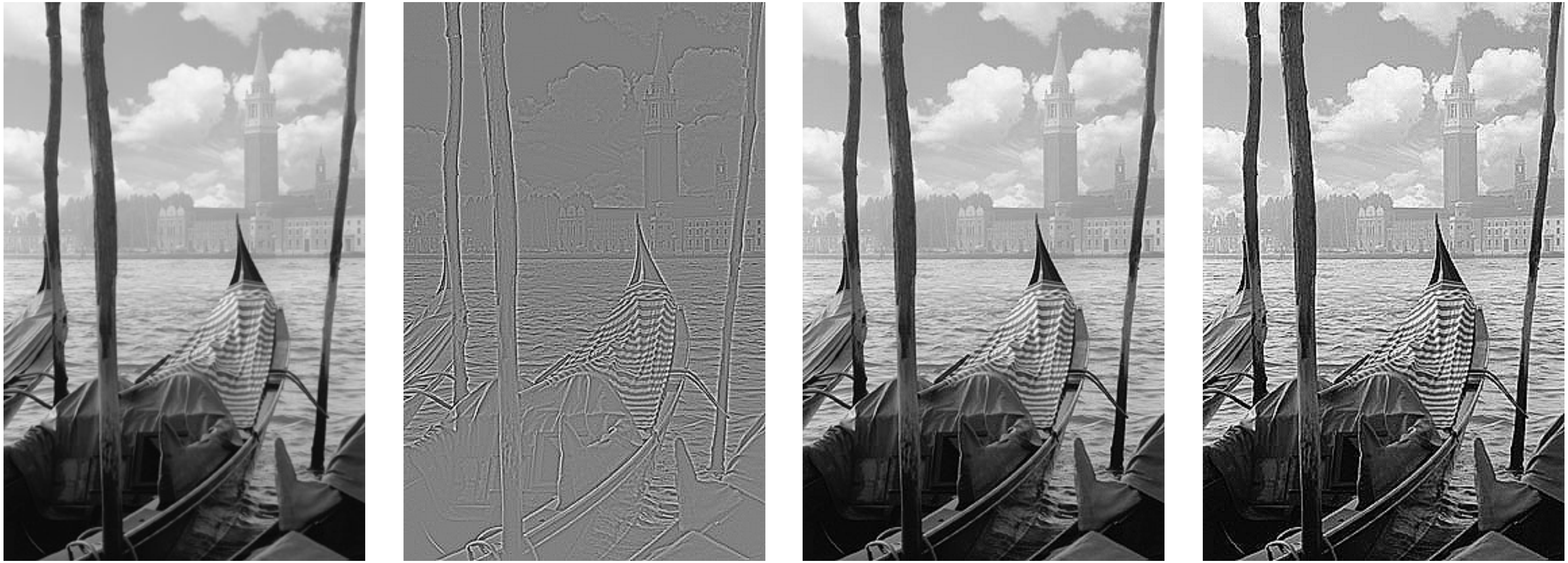
Let $\alpha = \frac{5}{A}$, then

$$I' = I + \alpha(I - \overline{I}) = \frac{1}{A}\left(AI + 5(I - \overline{I})\right)$$

$$\frac{1}{A}\left[\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & A & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array} + \left(\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 5 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array} - \begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 1 & 1 & 1 \\ \hline 0 & 1 & 0 \\ \hline\end{array}\right)\right]$$

$$= \frac{1}{A}\begin{array}{|c|c|c|}\hline 0 & -1 & 0 \\ \hline -1 & A+4 & -1 \\ \hline 0 & -1 & 0 \\ \hline\end{array}$$

Unsharp Masking

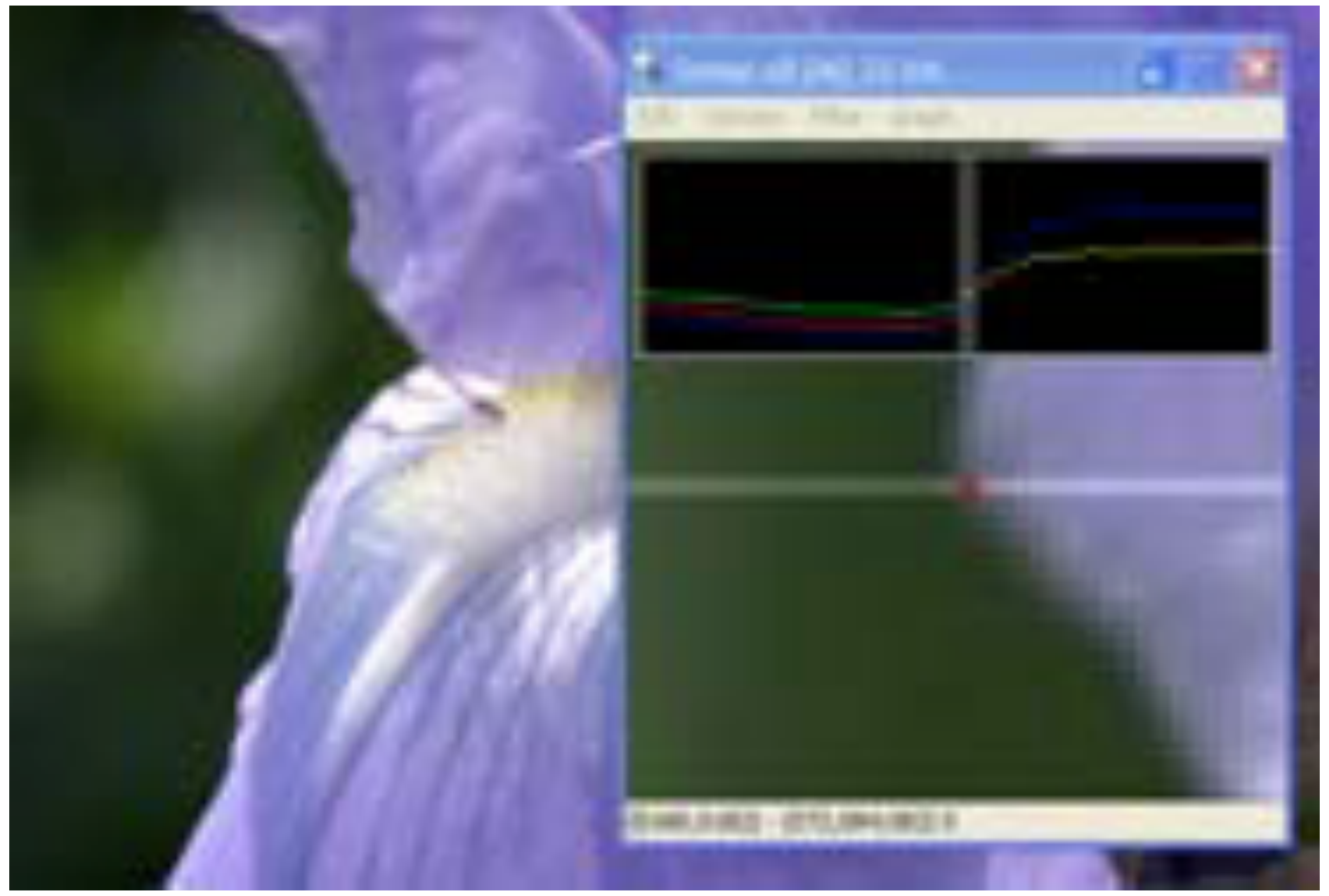


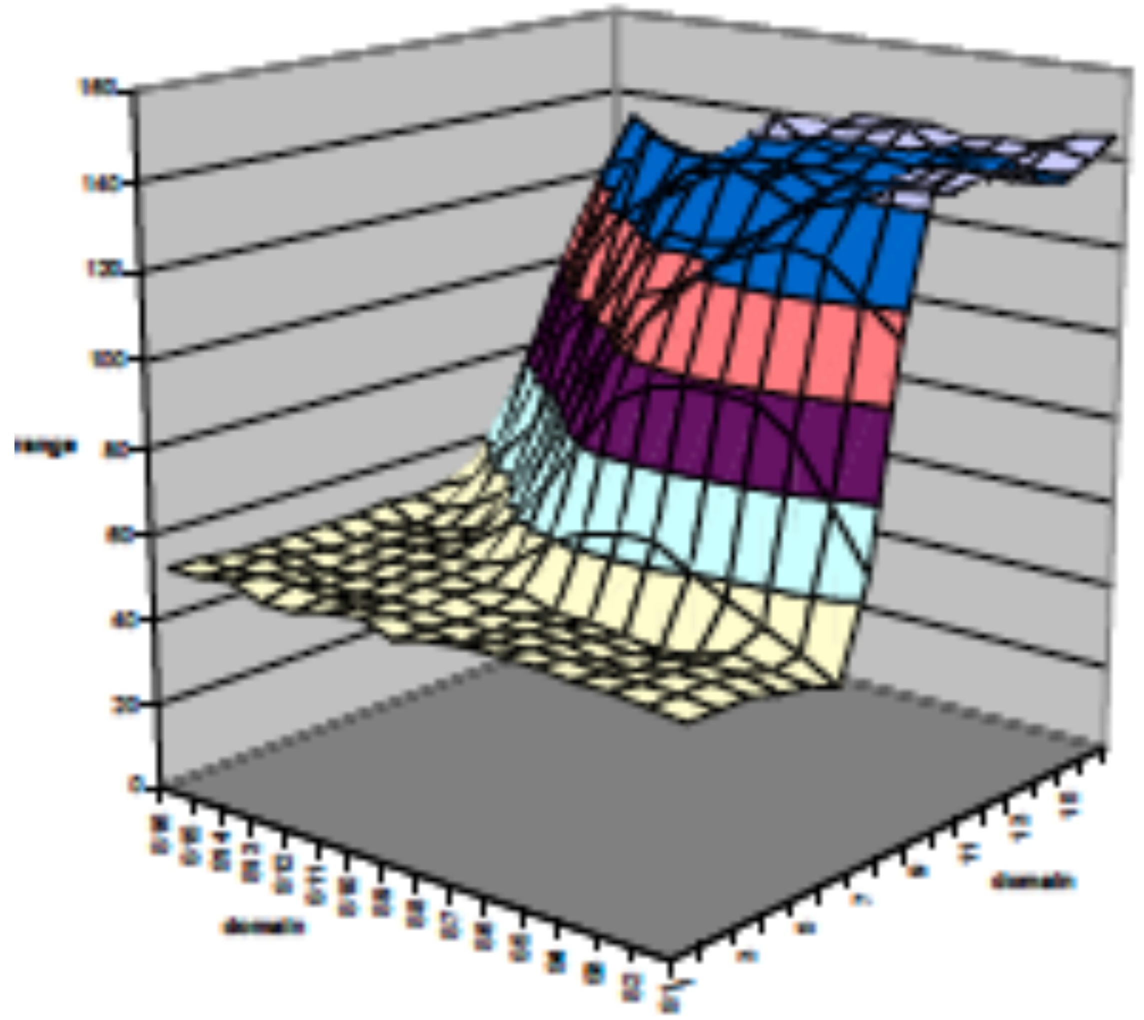
Tradeoff

- Blurring:
  - Reduces noise
  - Causes blur
- Sharpening:
  - Reduces blur
  - Strengthens noise

Edge Detection

- Edges between objects in images are often places where there are strong changes
- Find these using (approximations to) image derivatives





Approximating Derivatives

- Can approximate derivatives with finite differences

$$\frac{d}{dt}f(t) = \lim_{h \to 0} \frac{f(t+h) - f(t)}{h}$$

- Can choose
  - Forward (right)
  - Backward (left)
  - Central

$$\frac{d}{dt}f(t) \approx \frac{f(t+1) - f(t)}{1}$$

$$\frac{d}{dt}f(t) \approx \frac{f(t) - f(t-1)}{1}$$

$$\frac{d}{dt}f(t) \approx \frac{f(t+1) - f(t-1)}{2}$$

Approximating Derivatives

- Simplest:
  Just take central differences horizontally and vertically

- Approximates partial derivatives

0	0	0
-1	0	1
0	0	0

$$\frac{\partial}{\partial x}$$

0	-1	0
0	0	0
0	1	0

$$\frac{\partial}{\partial y}$$

(Divide by 2, the separation between the pixels you're taking the difference between)

Prewitt Kernels

- Better still:
  Average in other direction

- More robust since you're reducing noise in one direction while taking derivative in the other

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

$$\frac{\partial}{\partial x}$$

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

$$\frac{\partial}{\partial y}$$

(Divide by 3, the sum of the averaging weights)

(Divide by 2, the separation between the pixels you're taking the difference between)

Sobel Kernels

- More common:
  Use a center-weighted average

- More robust to noise

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

$$\frac{\partial}{\partial x}$$

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

$$\frac{\partial}{\partial y}$$

(Divide by 4, the sum of the averaging weights)

(Divide by 2, the separation between the pixels you're taking the difference between)

2nd Derivatives

- Can also do second derivatives
- Differences of differences
- Can still combined with smoothing in other direction

0	0	0
1	-2	1
0	0	0

0	1	0
0	-2	0
0	1	0

1	-2	1
1	-2	1
1	-2	1

1	1	1
-2	-2	-2
1	1	1

Gradients

- The *gradient* is a vector of partial derivatives with respect to each of a function's inputs

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \vdots \end{bmatrix}$$

- The direction of the gradient is the *direction of greatest increase*

- The magnitude of the gradient is the *amount of increase in that direction*

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 + \dots}$$

Gradient Magnitude

- The magnitude of the local gradient is the most common form of edge detector

$$\nabla I = \begin{bmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \end{bmatrix}$$

$$\|\nabla I\| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}$$

Lab #3

- Level operations with color:
  - Convert to grey (1)
  - Adjust brightness (2)
  - Adjust contrast (3)
- Image blending (4)
- Alpha blending (5)
- Uniform averaging (6)
- Median filtering (7)
- General convolution (8)
- Sharpening (9)
- Gradient magnitude (10)

Coming up...

- Points, vectors, matrices and other concepts from linear algebra
- Coordinate systems
- Transformations