Edge Detection

CS 543 / ECE 549 – Saurabh Gupta

Many slides from S. Lazebnik.

Edge Detection

While we wait, what do you see in this image?

CS 543 / ECE 549 – Saurabh Gupta

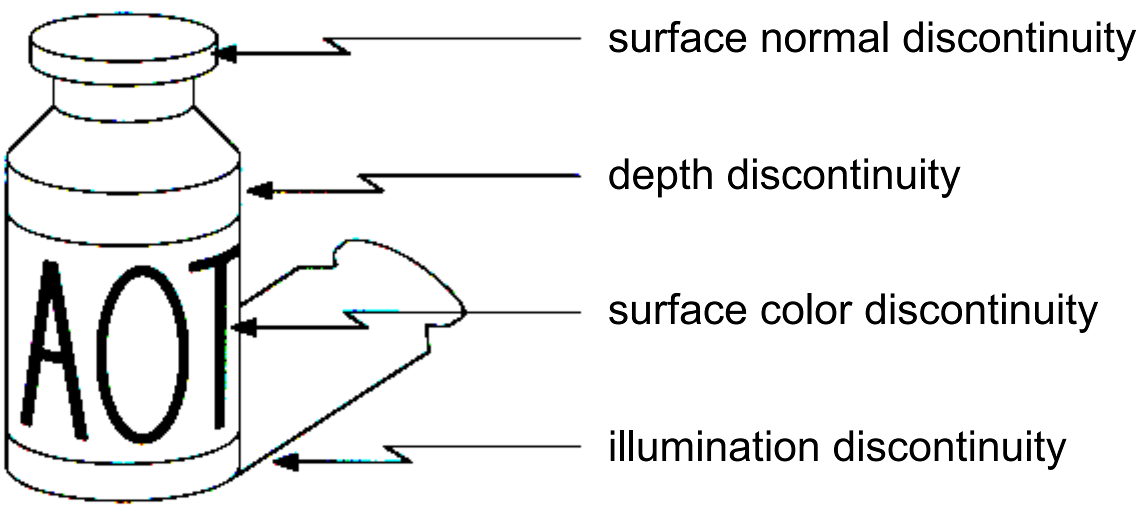
Many slides from S. Lazebnik.

Are Edges an Input or an Output?

Source: Gregory, R. L. (1970). The Intelligent Eye. New York, NY: McGraw-Hill Paperbacks.

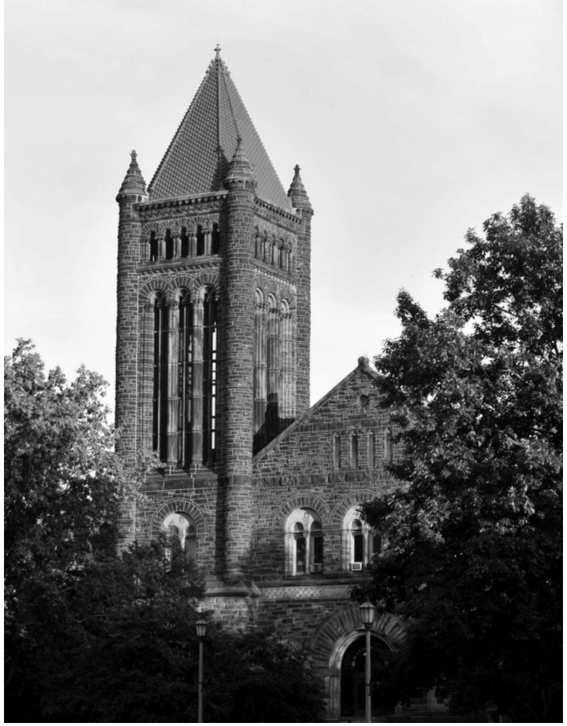
Edge detection

- **Goal:** Identify sudden changes (discontinuities) in an image
- Intuitively, edges carry most of the semantic and shape information from the image

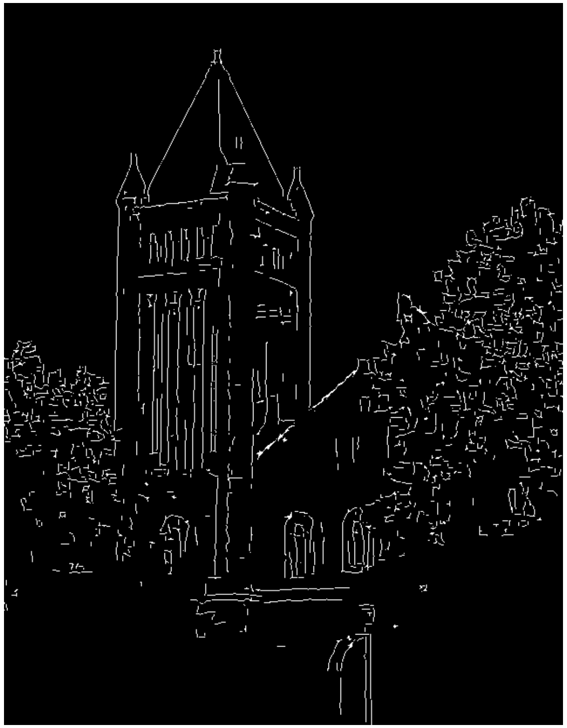


Sources: D. Lowe and S. Seitz

Edge detection



Ideal: artist's line drawing

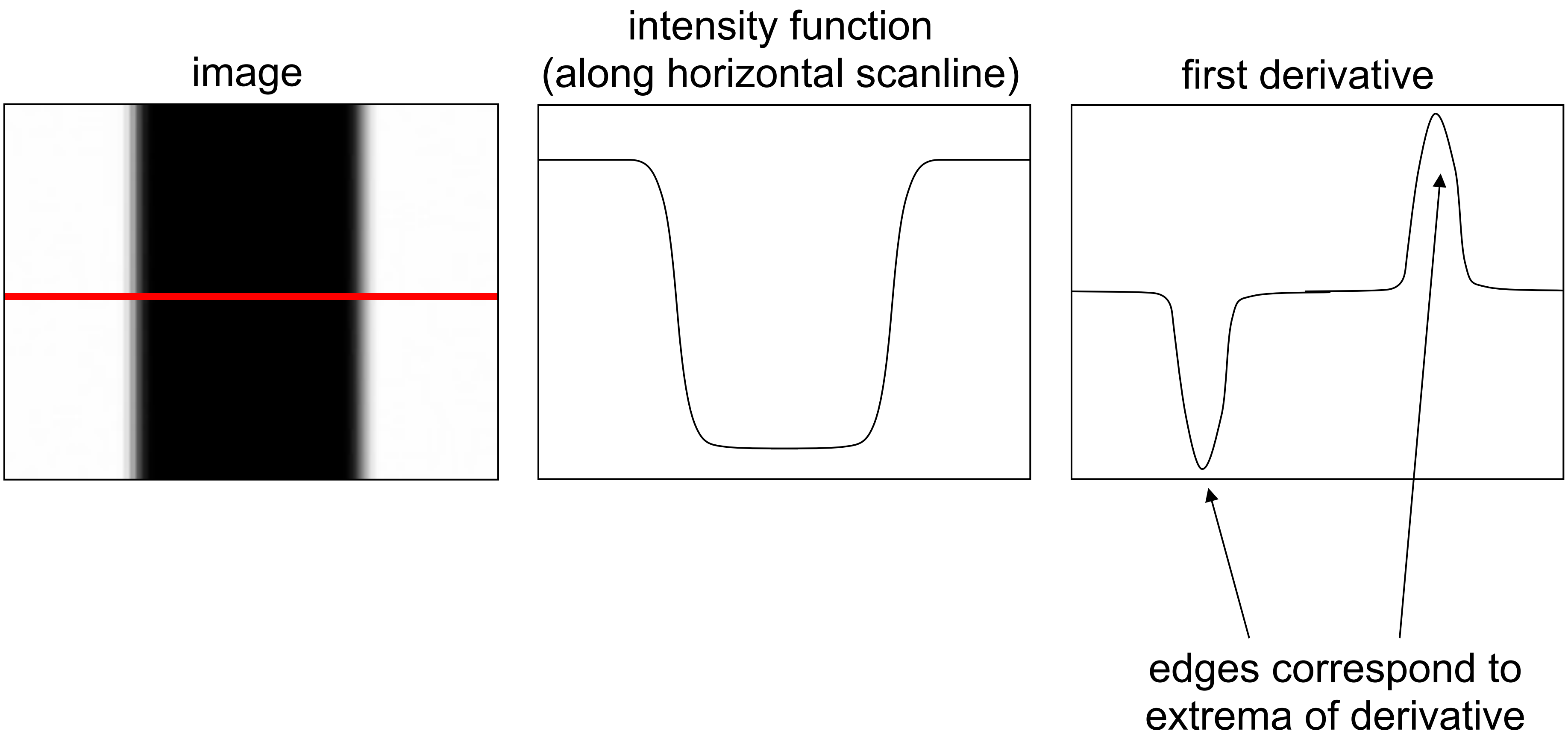


Reality

Source: https://www.clipartkey.com/view/wRJixi_drawing-of-altgeld-hall-chapel/

Edge detection

- An edge is a place of rapid change in the image intensity function



Source: L. Lazebnik

Derivatives with convolution

For 2D function f(x,y), the partial derivative is:

$$\frac{\partial f(x,y)}{\partial x} = \lim_{\varepsilon \to 0} \frac{f(x+\varepsilon, y) - f(x,y)}{\varepsilon}$$

For discrete data, we can approximate using finite differences:

$$\frac{\partial f(x,y)}{\partial x} \approx \frac{f(x+1, y) - f(x,y)}{1}$$

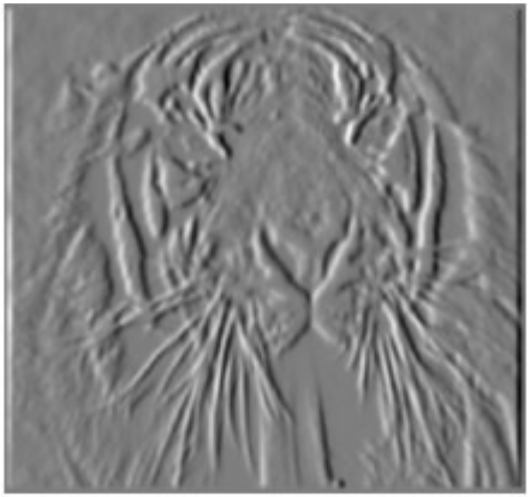
To implement the above as convolution, what would be the associated filter?

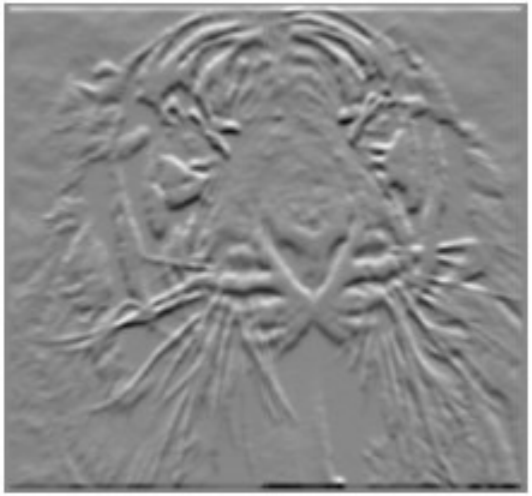
Source: K. Grauman

Partial derivatives of an image

$$\frac{\partial f(x,y)}{\partial x}$$

$$\frac{\partial f(x,y)}{\partial y}$$





-1	1

-1
1

or

1
-1

Which shows changes with respect to x?

Source: L. Lazebnik

Finite difference filters

Other approximations of derivative filters exist:

- Prewitt

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

- Sobel

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

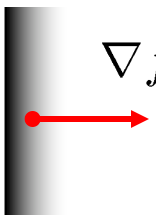
- Roberts

$$M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \qquad M_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

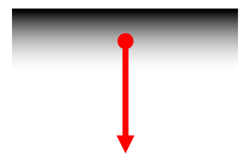
Source: K. Grauman

Image gradient

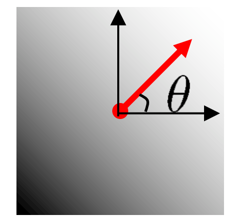
The gradient of an image: $\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$



$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$



$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$



$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$

The gradient points in the direction of most rapid increase in intensity

- How does this direction relate to the direction of the edge?

The gradient direction is given by $\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$

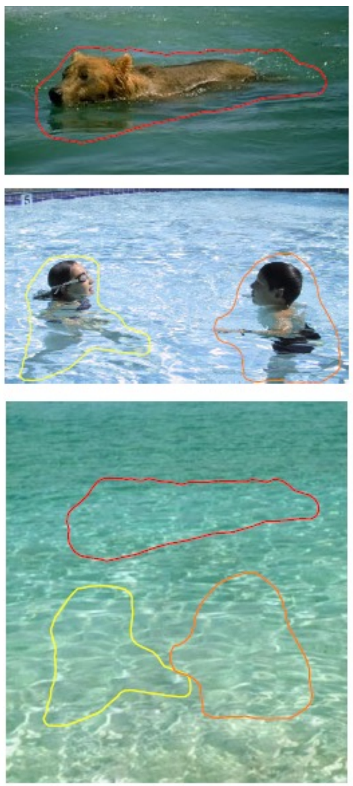
The edge strength is given by the gradient magnitude

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

Source: Steve Seitz

Application: Gradient-domain image editing

- Goal: solve for pixel values in the target region to match gradients of the source region while keeping background pixels the same



sources/destinations

cloning

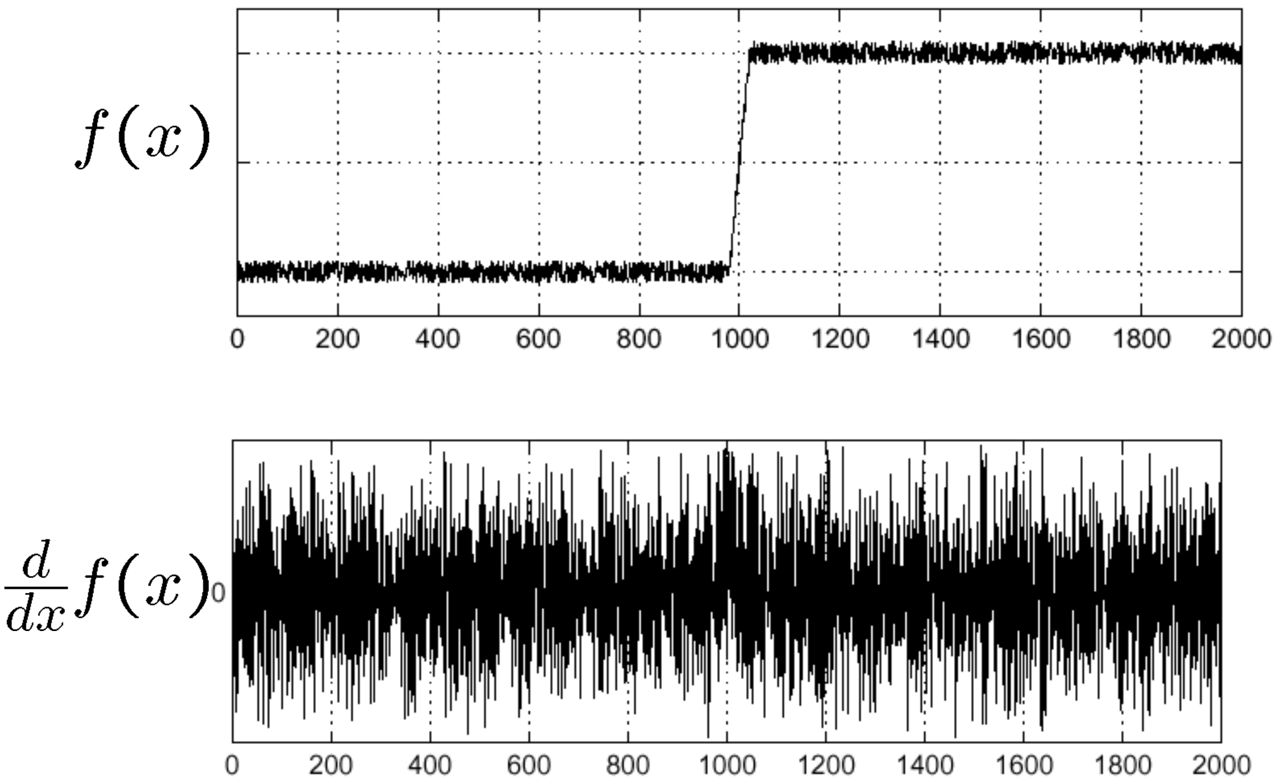
seamless cloning

P. Perez, M. Gangnet, A. Blake, [Poisson Image Editing](), SIGGRAPH 2003

Source: L. Lazebnik

Effects of noise

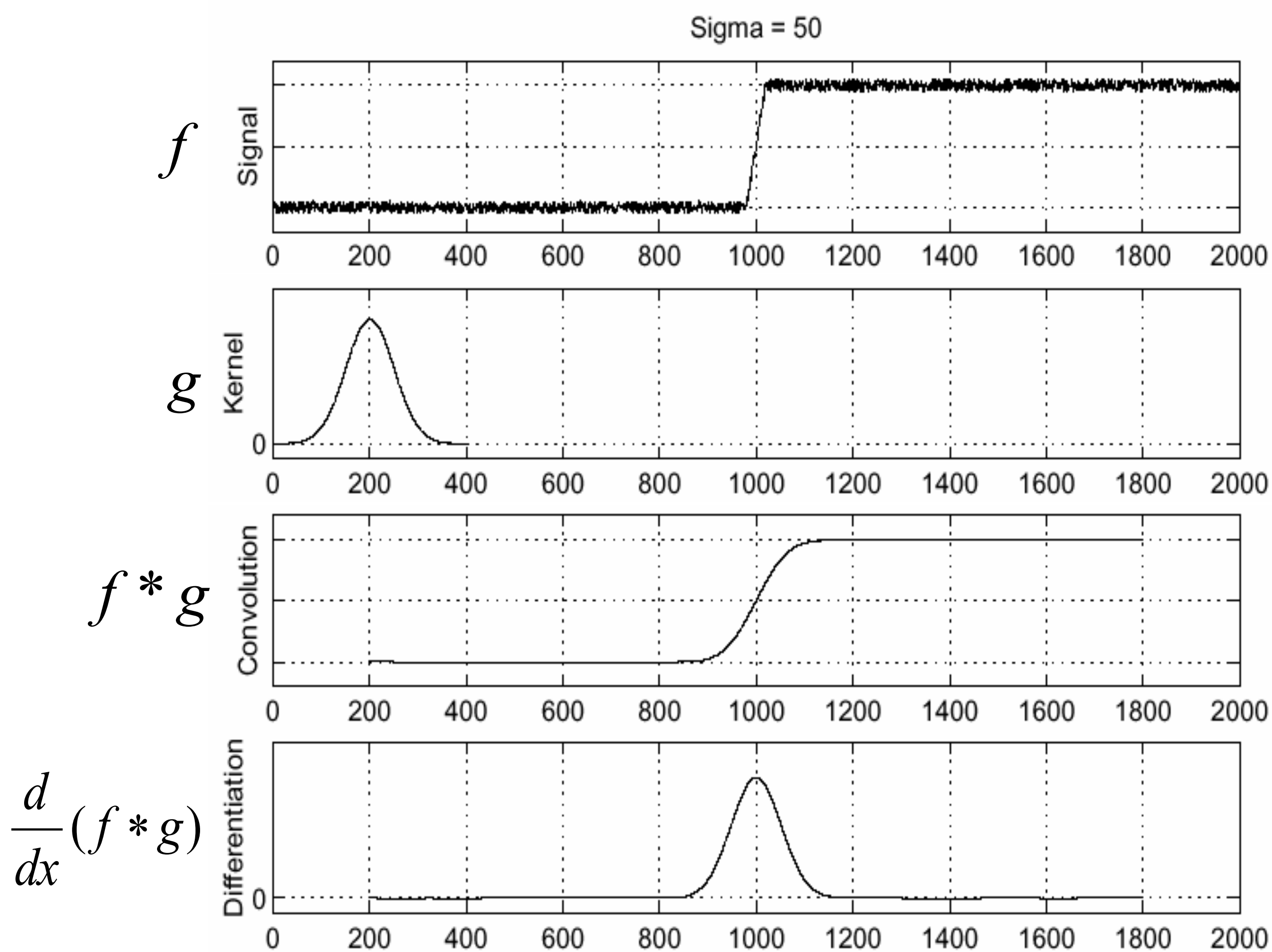
Consider a single row or column of the image



Where is the edge?

Source: S. Seitz

Solution: smooth first



- To find edges, look for peaks in $\frac{d}{dx}(f * g)$

Source: S. Seitz

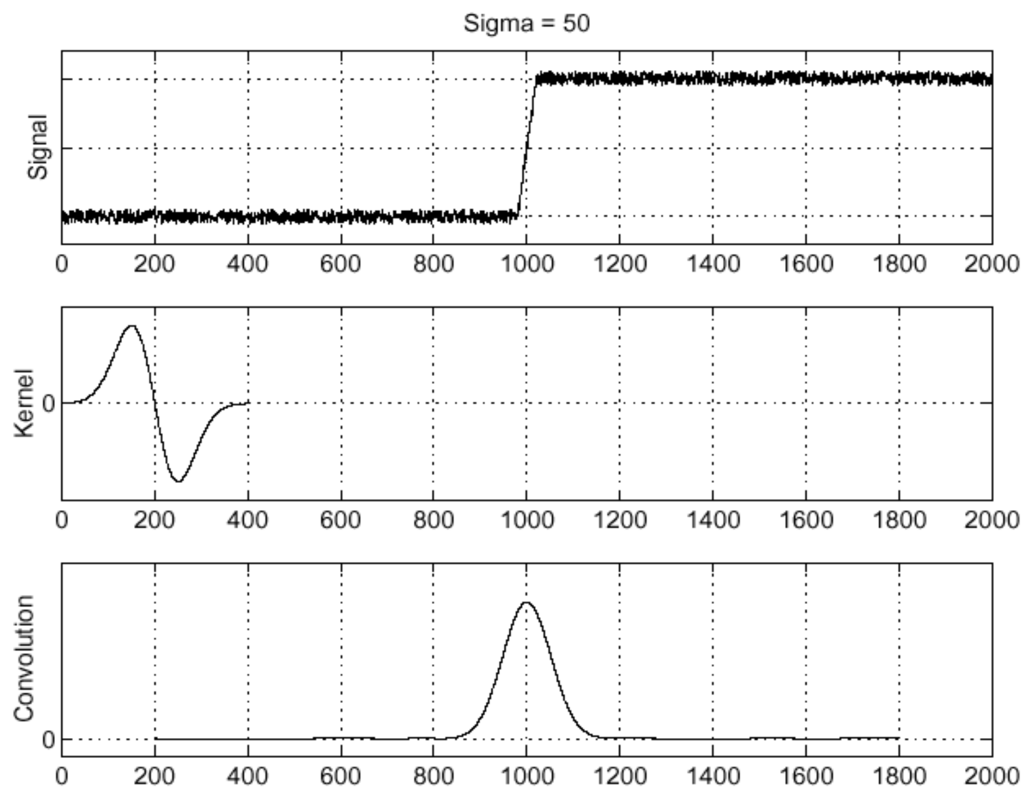
Derivative theorem of convolution

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

$$f$$

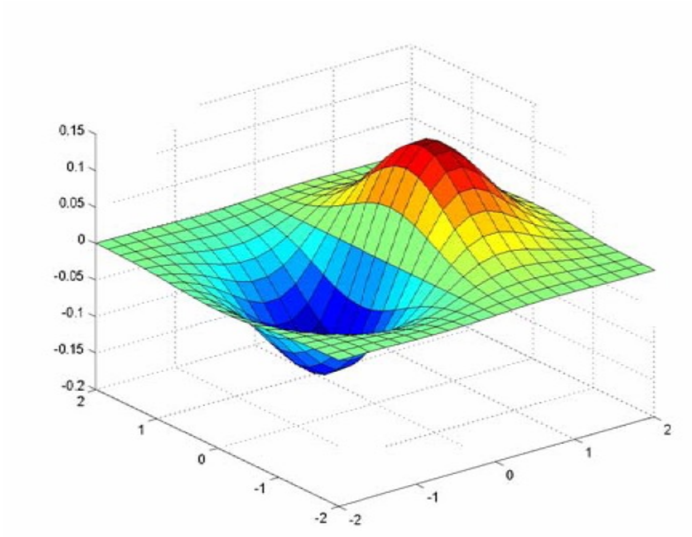
$$\frac{d}{dx} g$$

$$f * \frac{d}{dx} g$$

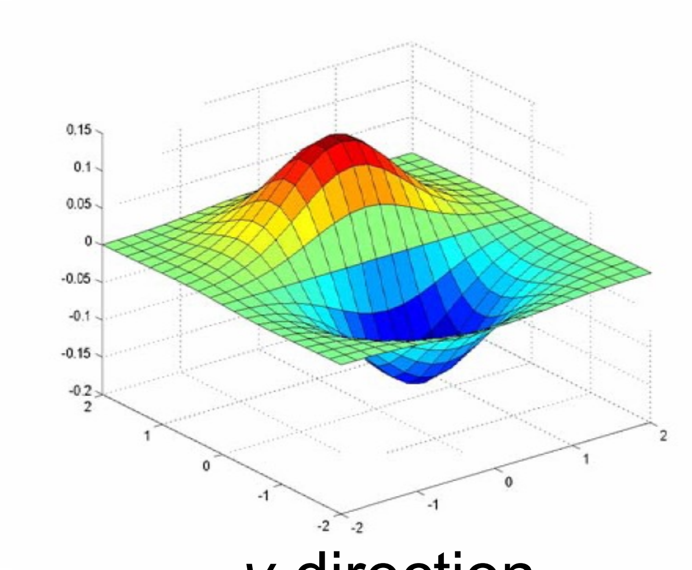


Source: S. Seitz

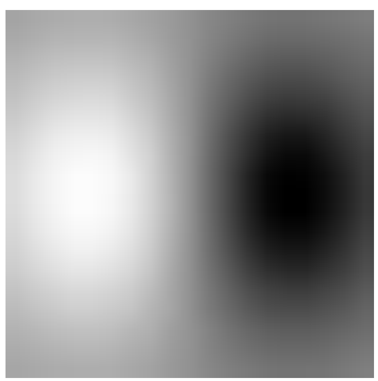
Derivative of Gaussian filters

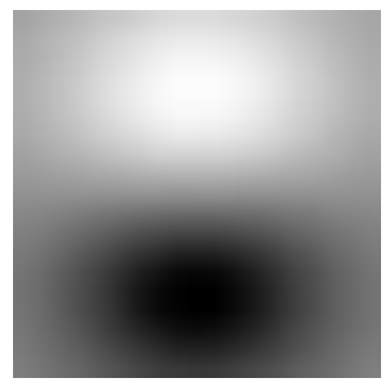


x-direction



y-direction

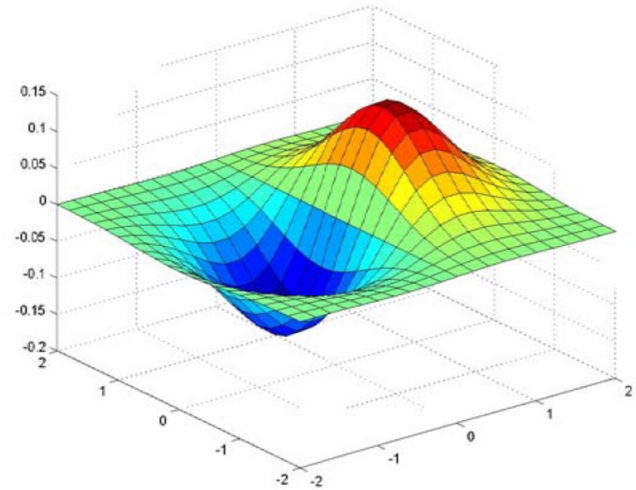




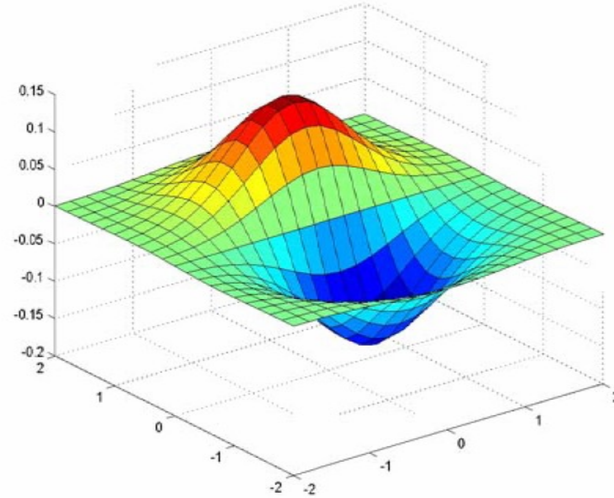
Which one finds horizontal/vertical edges?

Source: L. Lazebnik

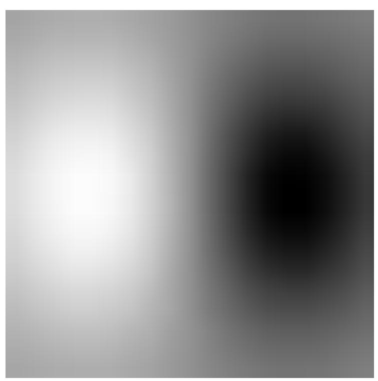
Derivative of Gaussian filters

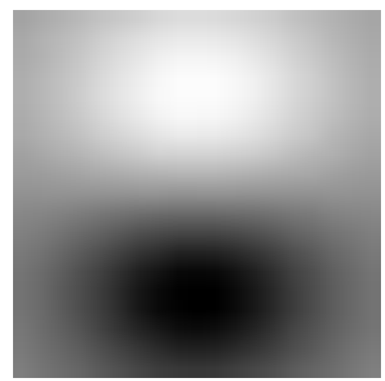


x-direction



y-direction





Are these filters separable?

Recall: Separability of the Gaussian filter

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

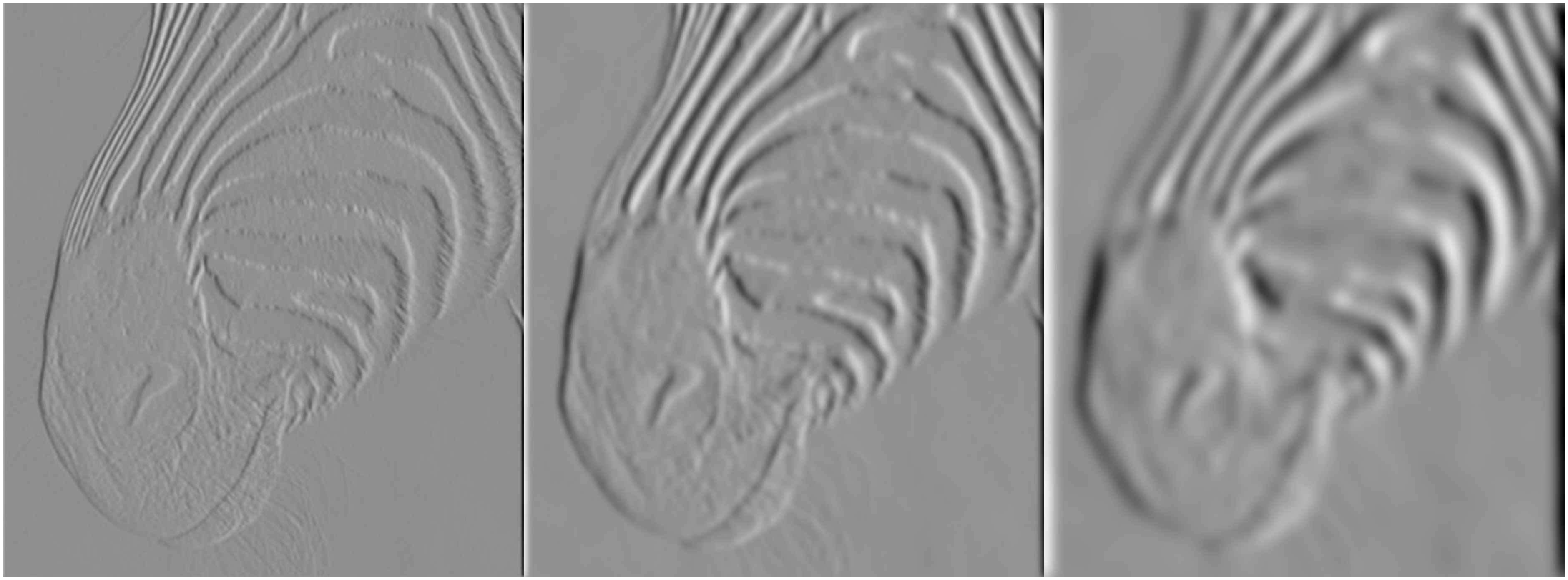
$$= \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}\right)\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{y^2}{2\sigma^2}}\right)$$

The 2D Gaussian can be expressed as the product of two functions, one a function of x and the other a function of y.

In this case the two functions are the (identical) 1D Gaussian.

Source: D. Lowe

Scale of Gaussian derivative filter



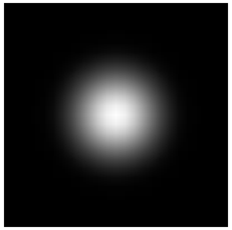
1 pixel 3 pixels 7 pixels

Smoothed derivative removes noise, but blurs edge. Also finds edges at different "scales"

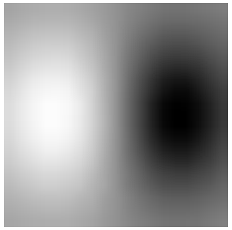
Source: D. Forsyth

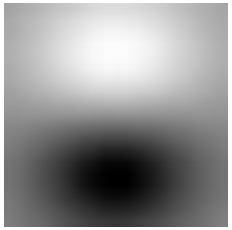
Review: Smoothing vs. derivative filters

Smoothing filters



- Gaussian: remove “high-frequency” components; “low-pass” filter
- Can the values of a smoothing filter be negative?
- What should the values sum to?
  - **One:** constant regions are not affected by the filter



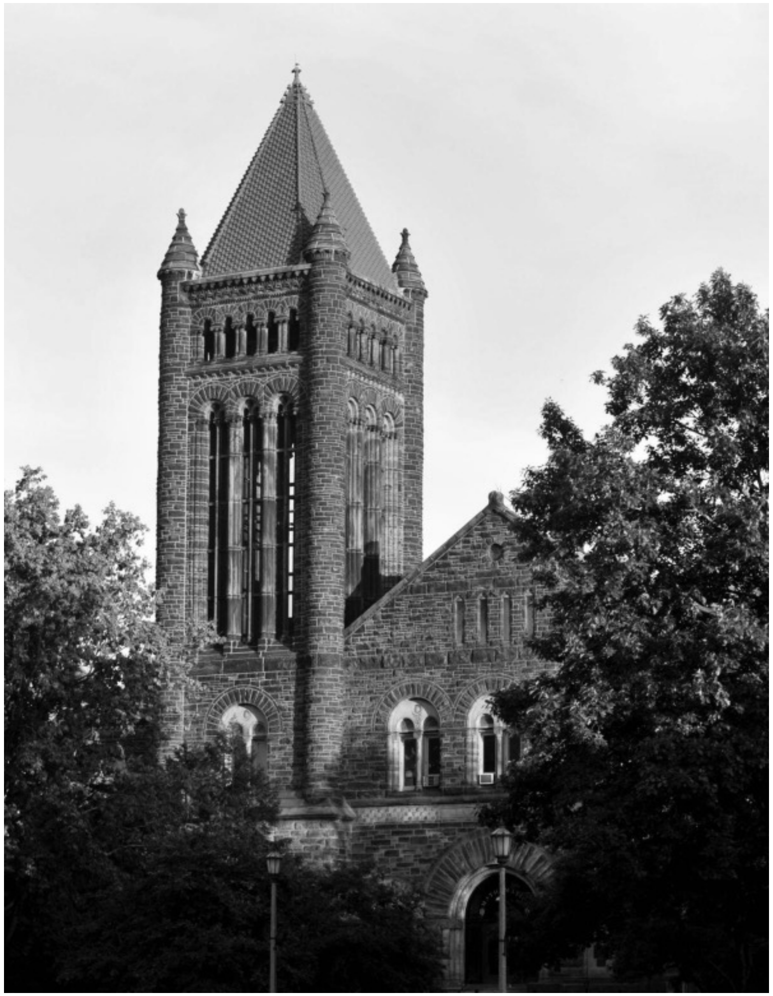


Derivative filters

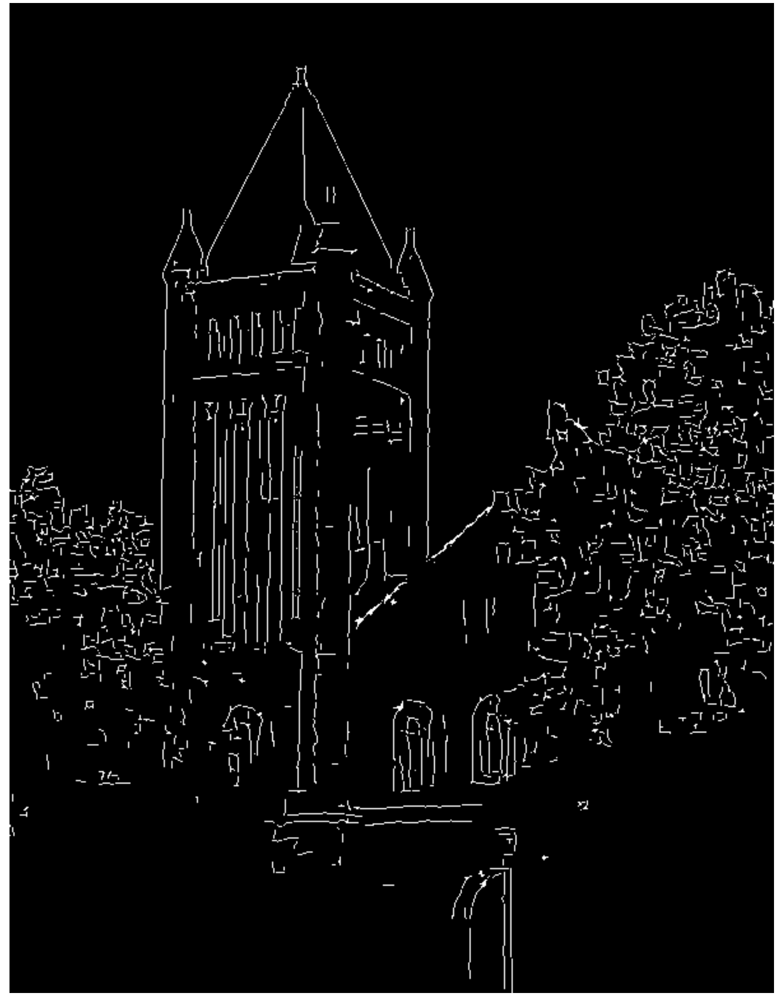
- Derivatives of Gaussian
- Can the values of a derivative filter be negative?
- What should the values sum to?
  - **Zero:** no response in constant regions

Source: L. Lazebnik

Building an edge detector

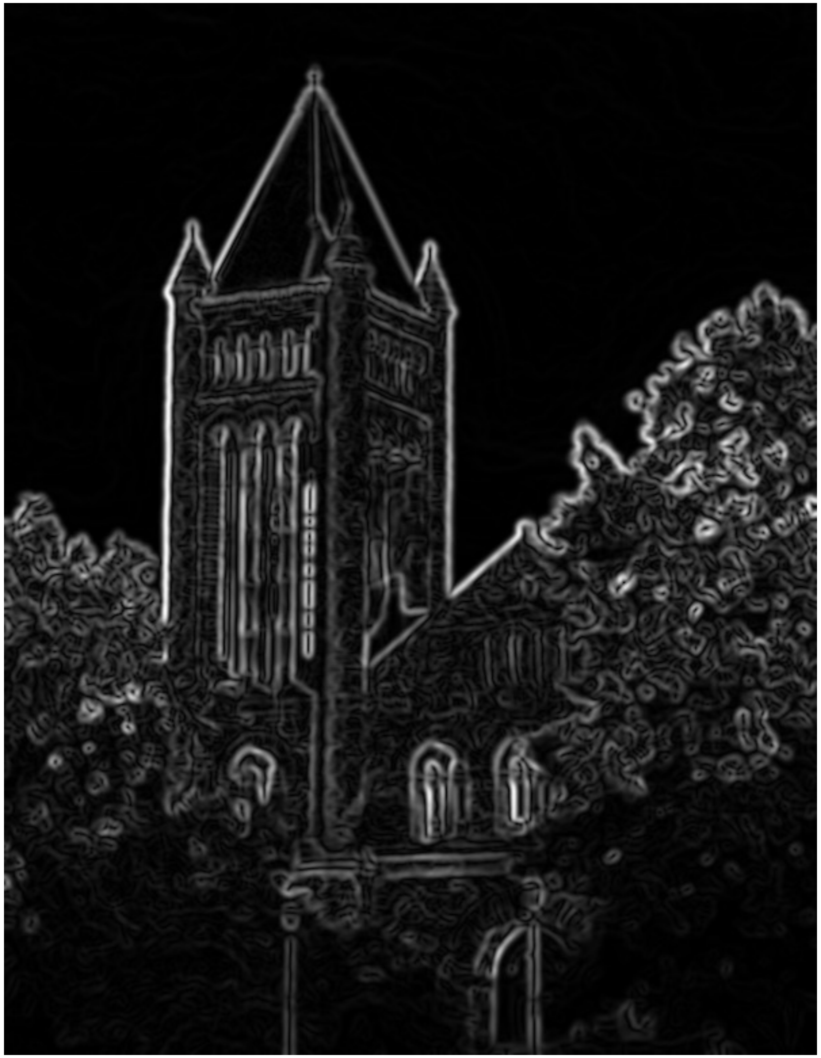


original image



final output

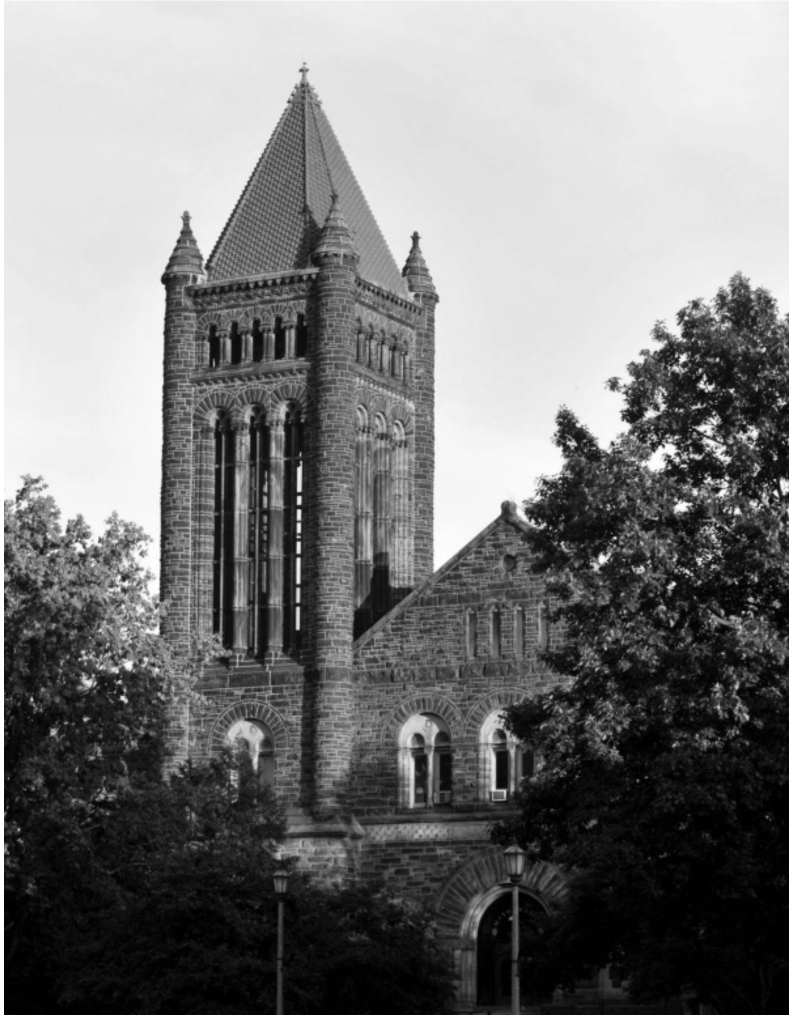
Building an edge detector

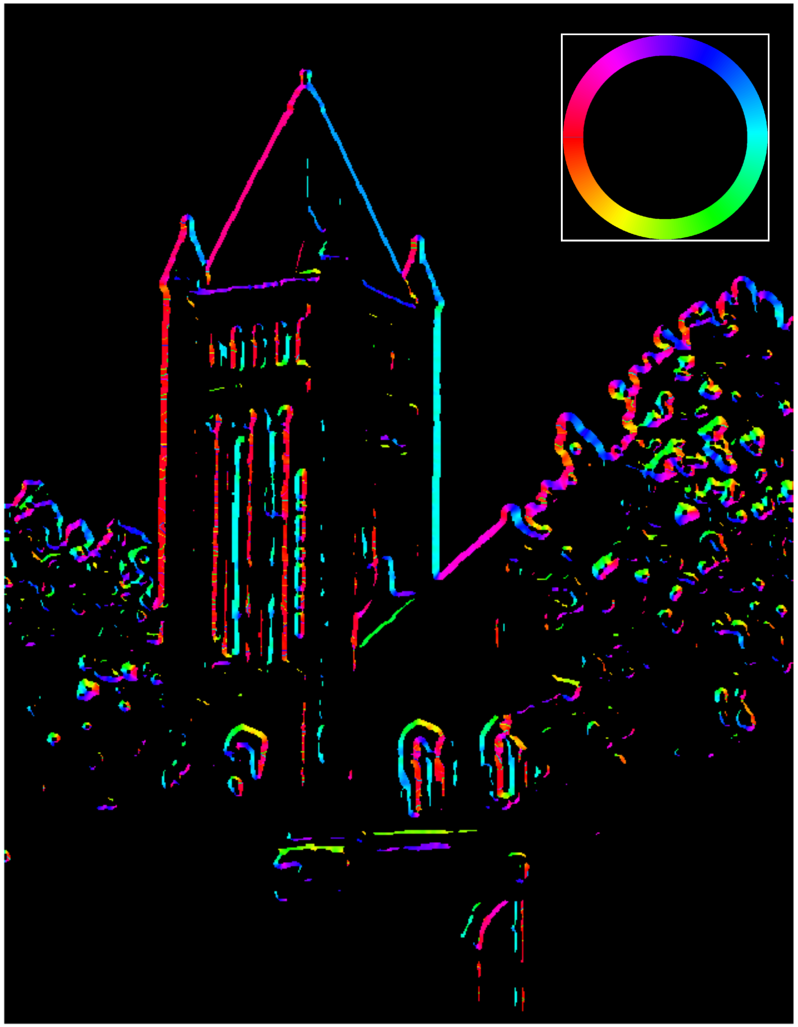


norm of the gradient

Building an edge detector

$\theta = $ `np.arctan2(-gy, gx)`





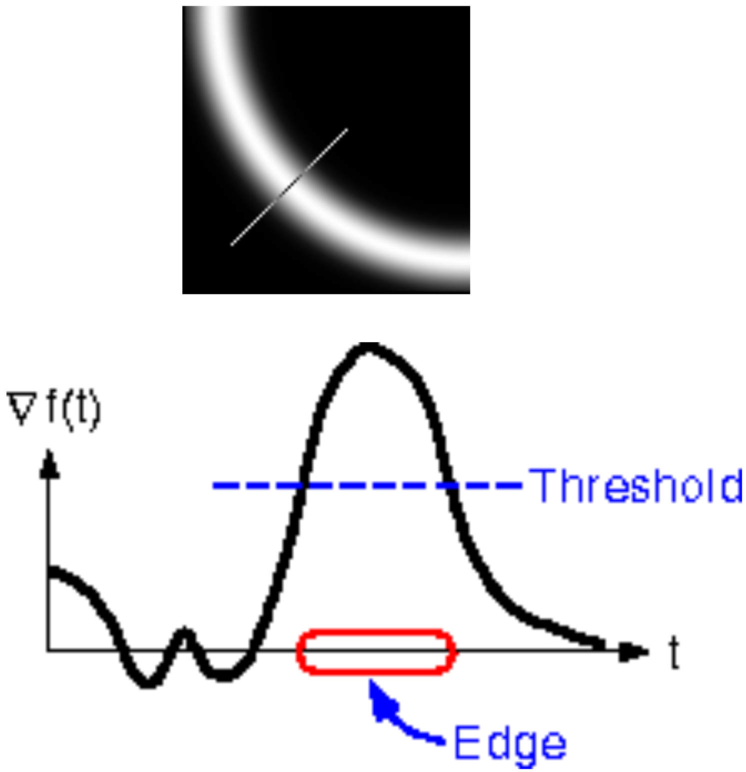
orientation of the gradient

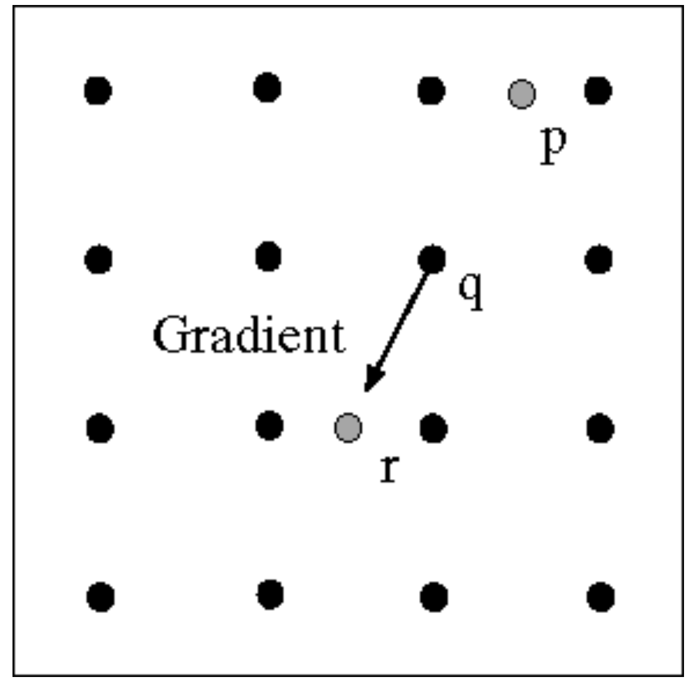
Building an edge detector

How to turn these thick regions of the gradient into curves?

Norm of the gradient > threshold

Non-maximum suppression



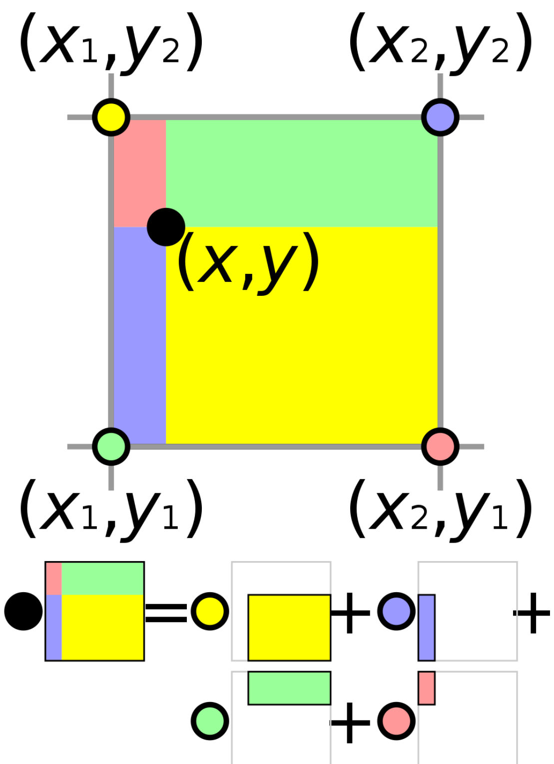


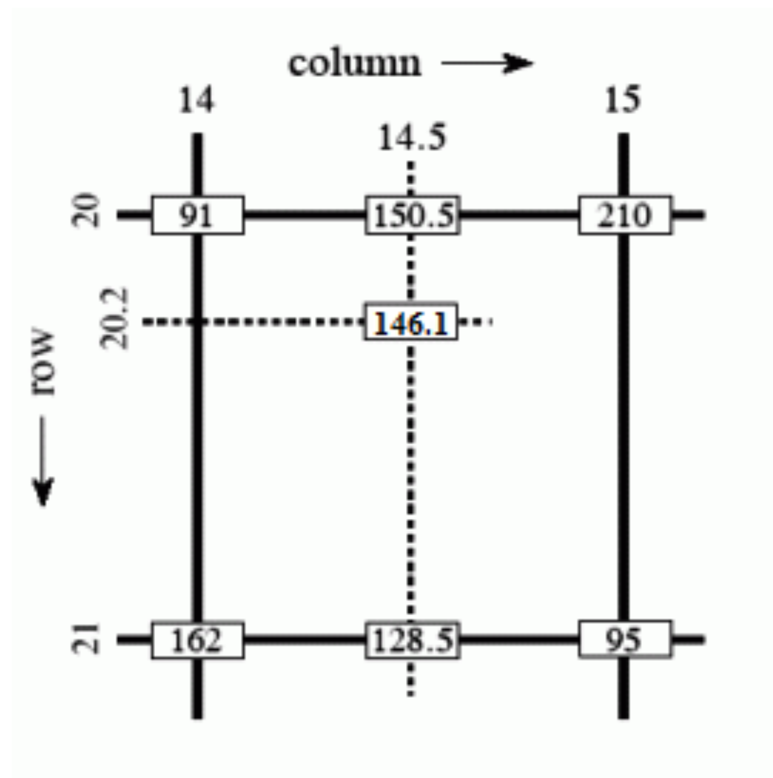
- For each location q above threshold, check that the gradient magnitude is higher than at neighbors p and r along the direction of the gradient
  - May need to interpolate to get the magnitudes at p and r

Source: L. Lazebnik

Bilinear Interpolation

$$f(x,y) \approx \begin{bmatrix} 1-x & x \end{bmatrix} \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1-y \\ y \end{bmatrix}$$





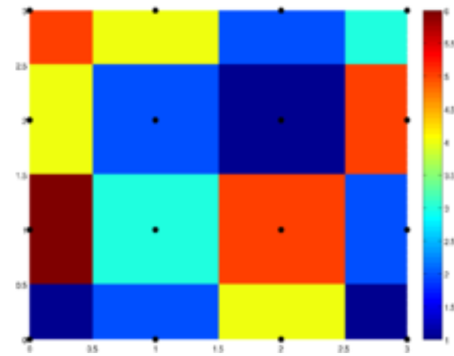
http://en.wikipedia.org/wiki/Bilinear_interpolation

Sidebar: Interpolation options

imx2 = imresize(im, 2,
interpolation_type)

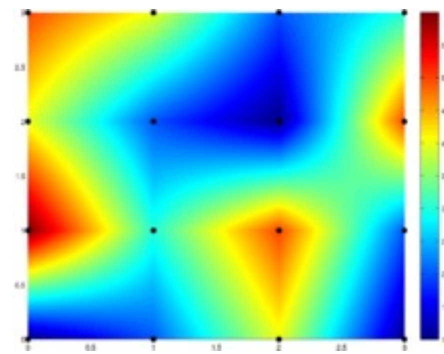
'nearest'

- Copy value from nearest known
- Very fast but creates blocky edges



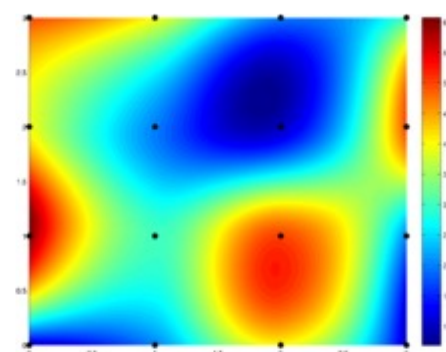
'bilinear'

- Weighted average from four nearest known pixels
- Fast and reasonable results



'bicubic' (default)

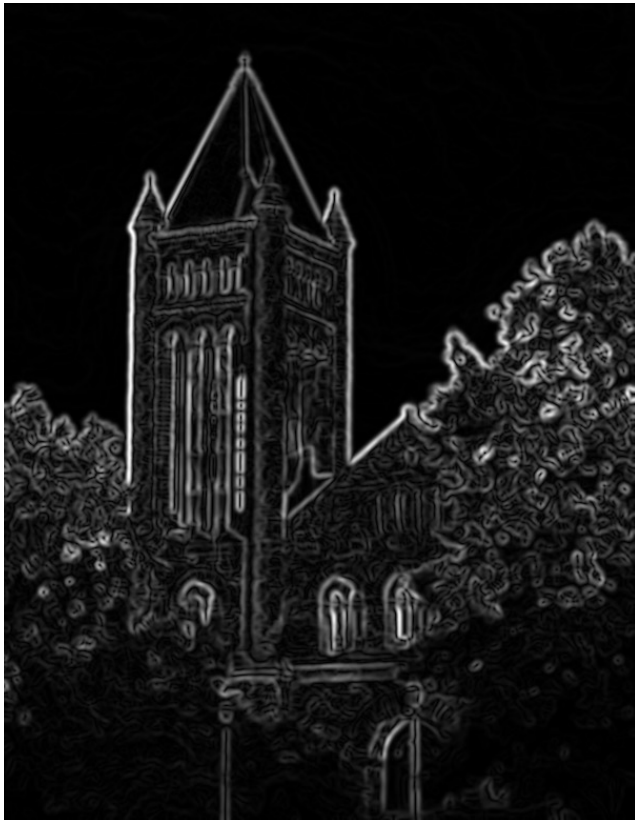
- Non-linear smoothing over larger area
- Slower, visually appealing, may create negative pixel values

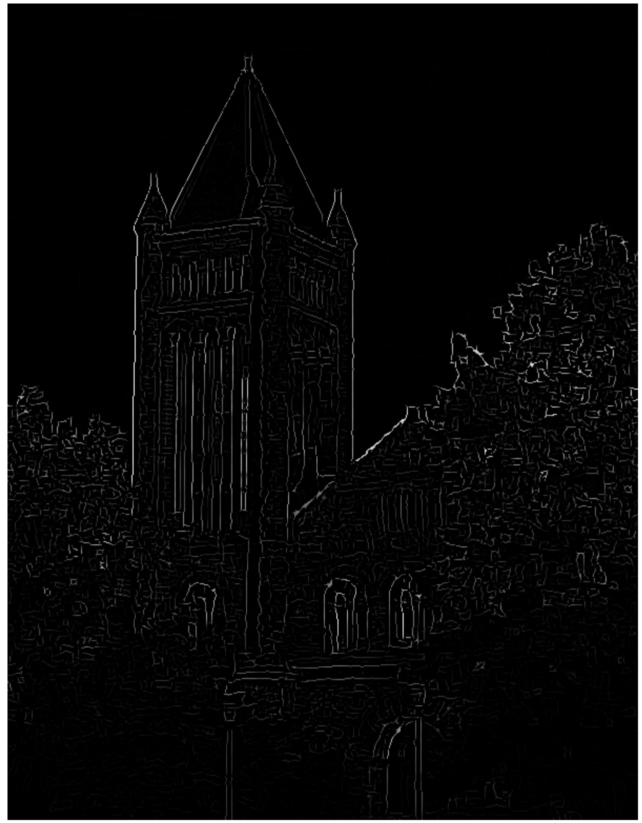


Source: D. Hoeim

Examples from http://en.wikipedia.org/wiki/Bicubic_interpolation

Non-maximum suppression





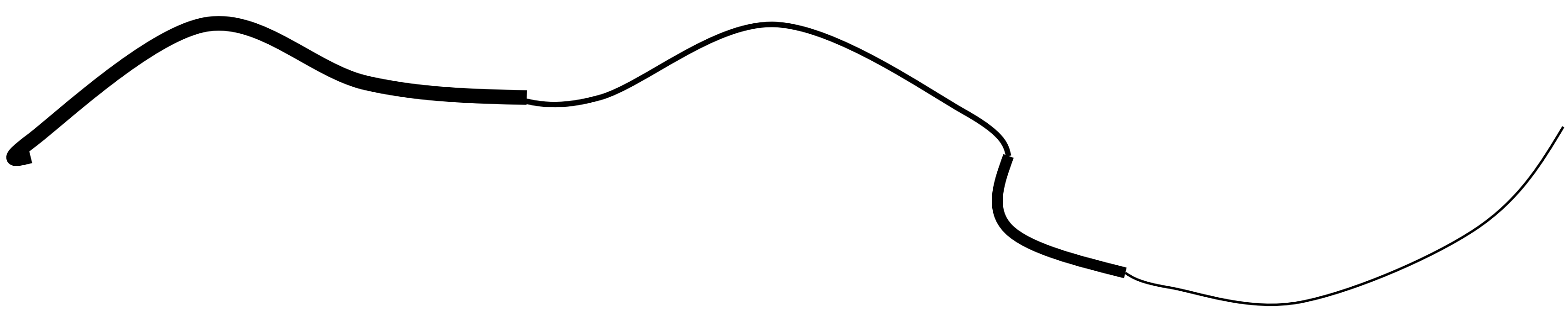
NMS

NMS > threshold

Another problem: pixels along this edge didn’t survive the thresholding

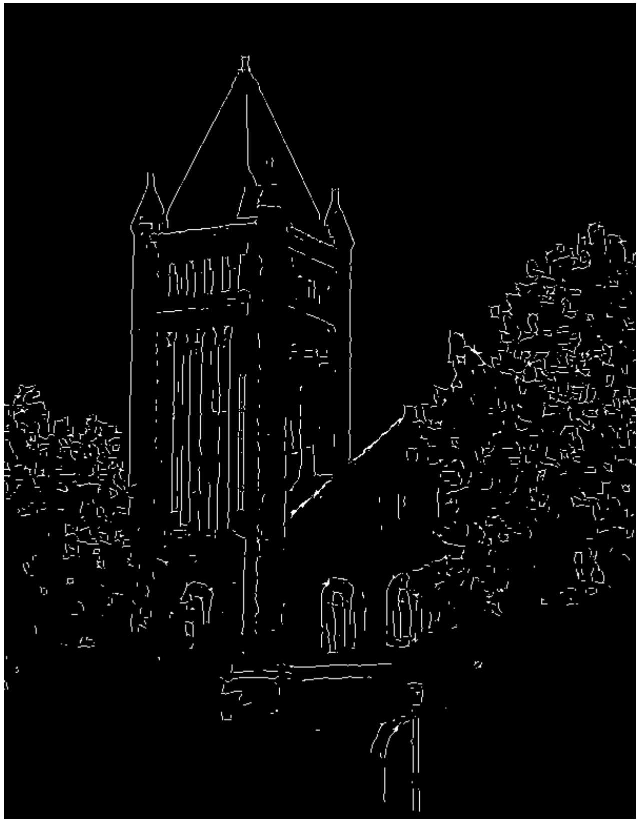
Hysteresis thresholding

Use a high threshold to start edge curves, and a low threshold to continue them.



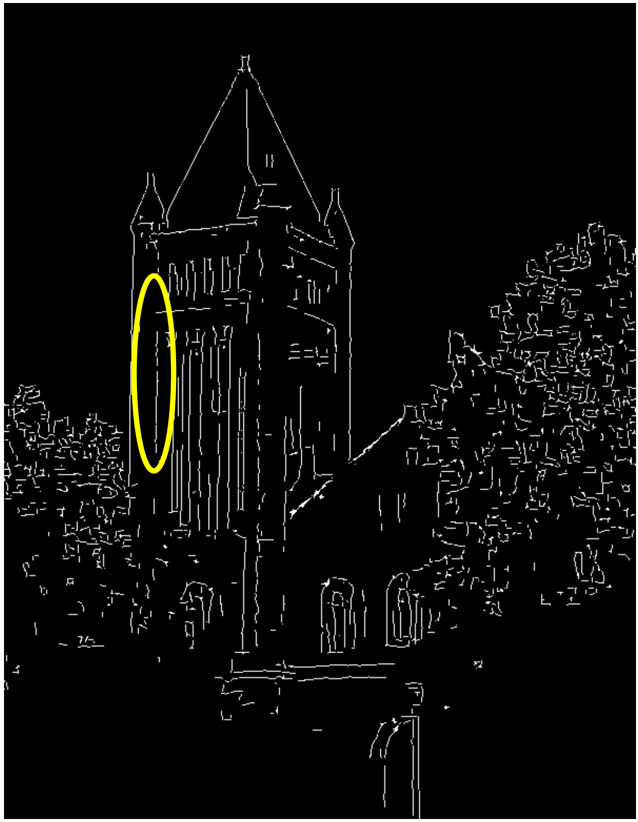
Source: Steve Seitz

Hysteresis



high threshold (strong edges)

low threshold (weak edges)



hysteresis threshold

Hysteresis thresholding

original image

high threshold (strong edges)

low threshold (weak edges)

hysteresis threshold

Source: L. Fei-Fei

Effect of σ (Gaussian kernel spread/size)

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 features

Source: S. Seitz

Recap: Canny edge detector

1. Compute x and y gradient images
2. Find magnitude and orientation of gradient
3. **Non-maximum suppression**:
    - Thin wide "ridges" down to single pixel width
4. **Linking and thresholding** (**hysteresis**):
    - Define two thresholds: low and high
    - Use the high threshold to start edge curves and the low threshold to continue them

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

Source: L. Lazebnik

Image gradients vs. meaningful contours

image | human segmentation | gradient magnitude

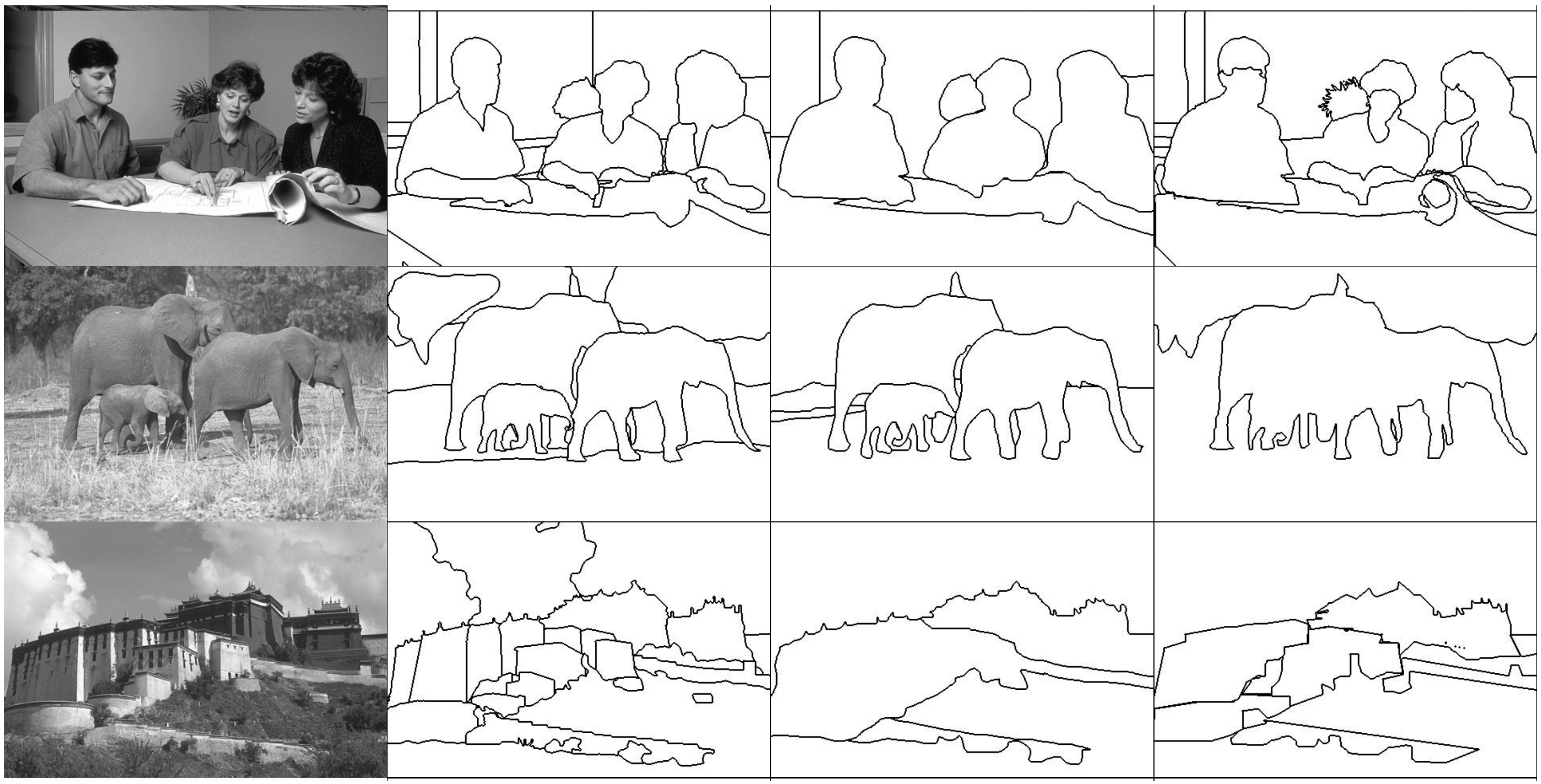
Berkeley segmentation database:

http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/

Source: L. Lazebnik

Do humans consistently segment images?

Divide each image into pieces, where each piece represents a distinguished thing in the image. It is important that all of the pieces have approximately equal importance. The number of things in each image is up to you. Something between 2 and 20 should be reasonable for any of our images.



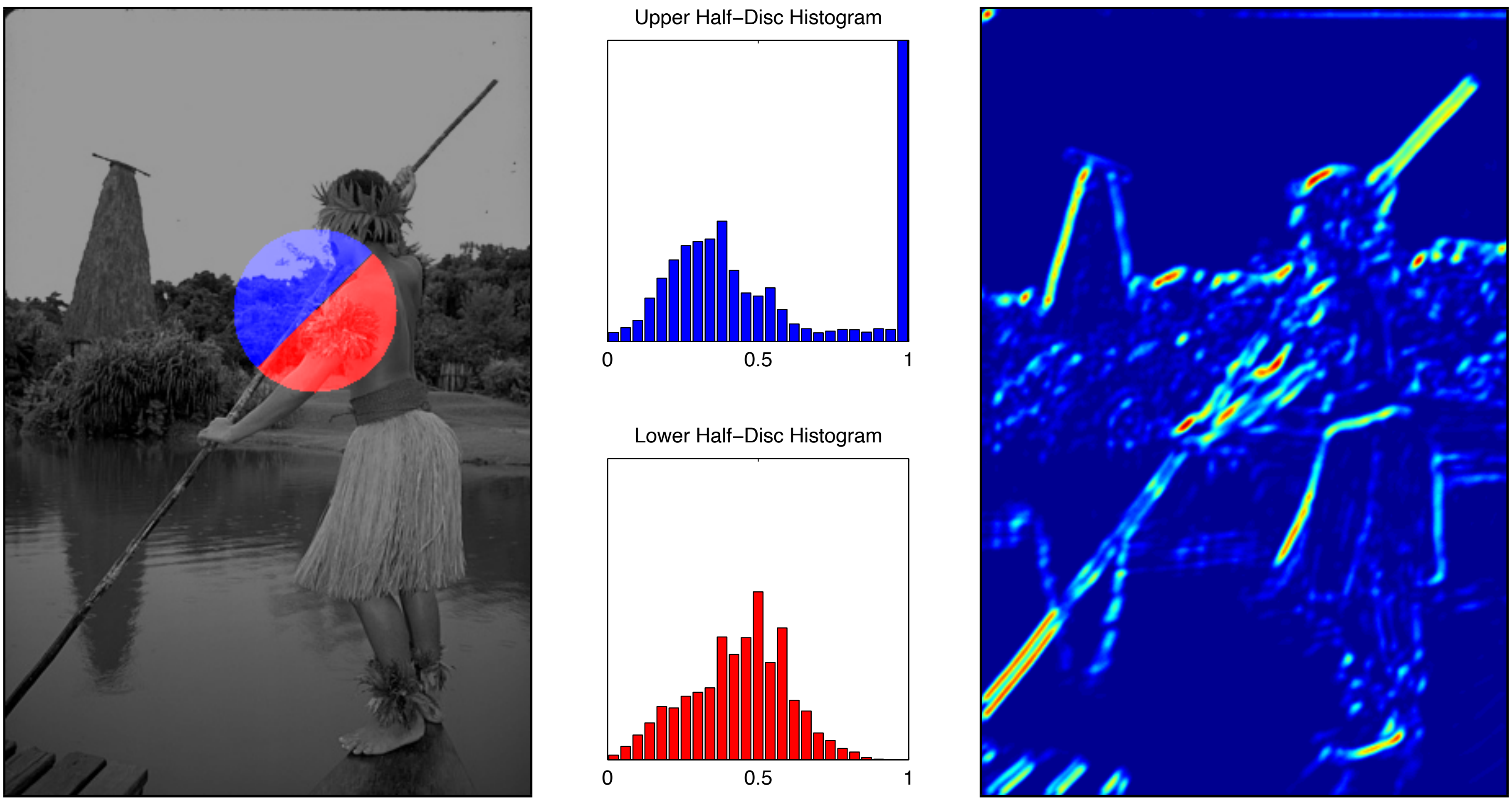
[A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics]

Aside: Datasets, Metrics and Benchmarks

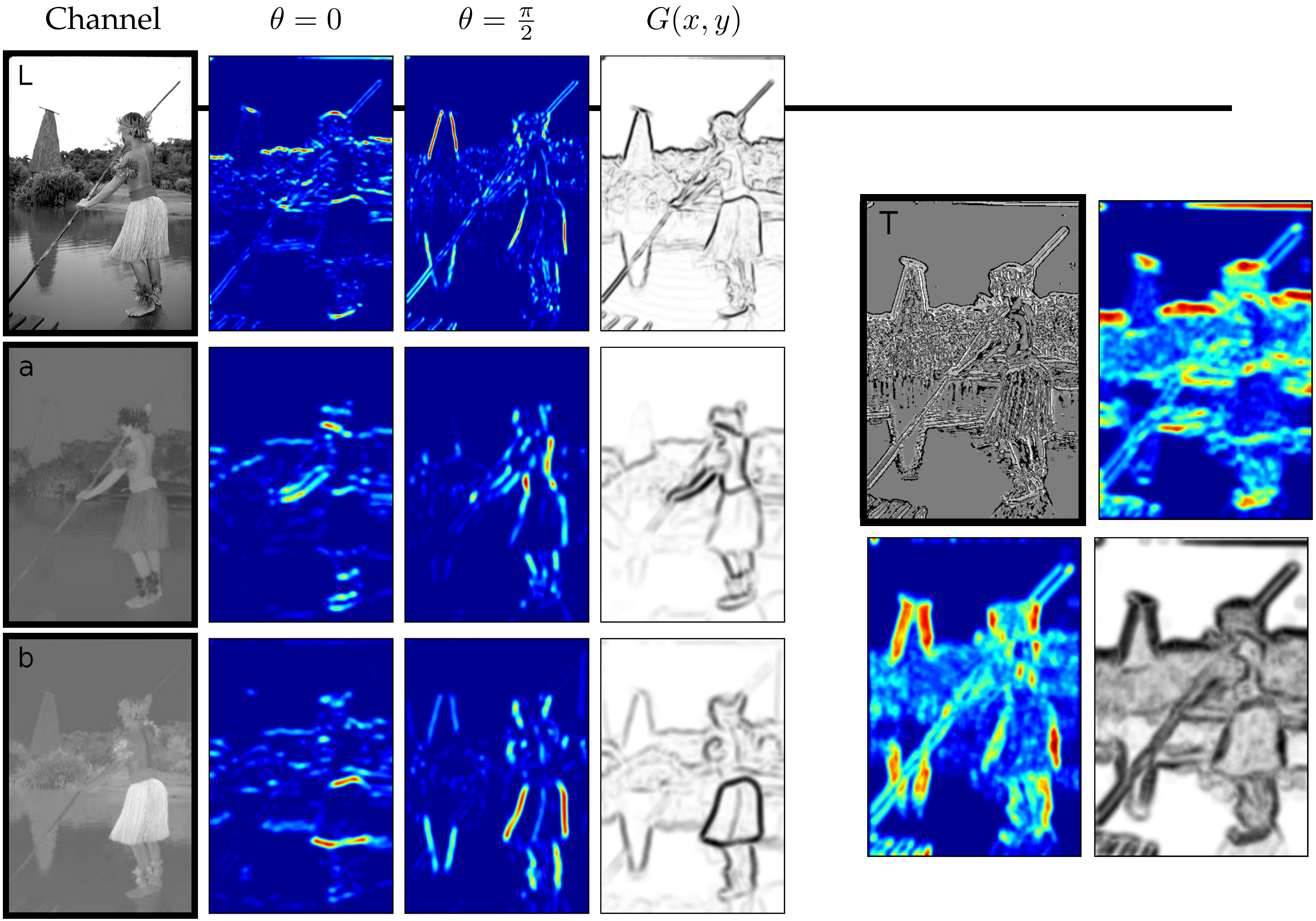
- Standard image sets
- Standard metrics
- Possible to *quantitatively* compare different methods

[A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics]()

pB Boundary Detector

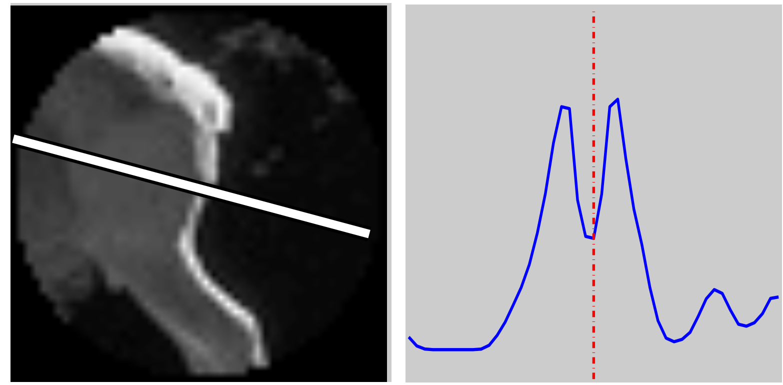


Contour Detection and Hierarchical Image Segmentation P. Arbeláez. PAMI 2010.



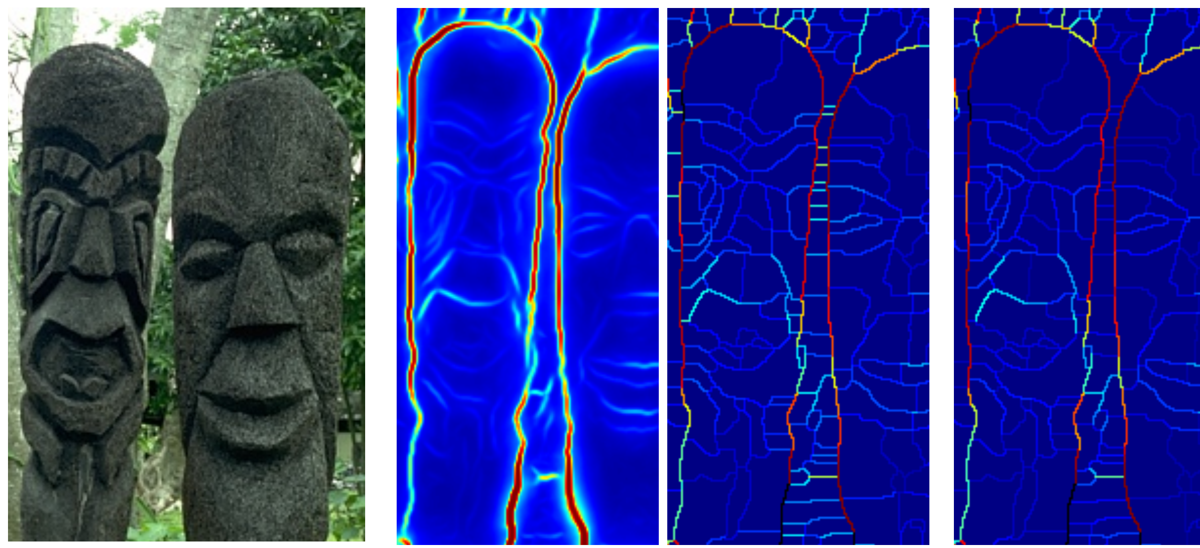
Contour Detection and Hierarchical Image Segmentation P. Arbeláez. PAMI 2010.

Lots of Tricks



“Bird Edge” Problem with Texture Gradients

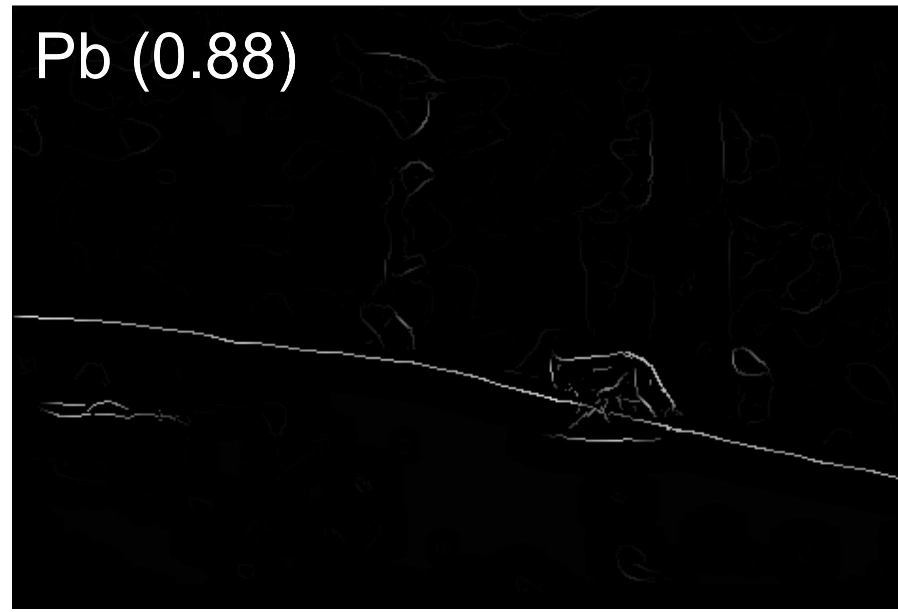
Sky Breaking

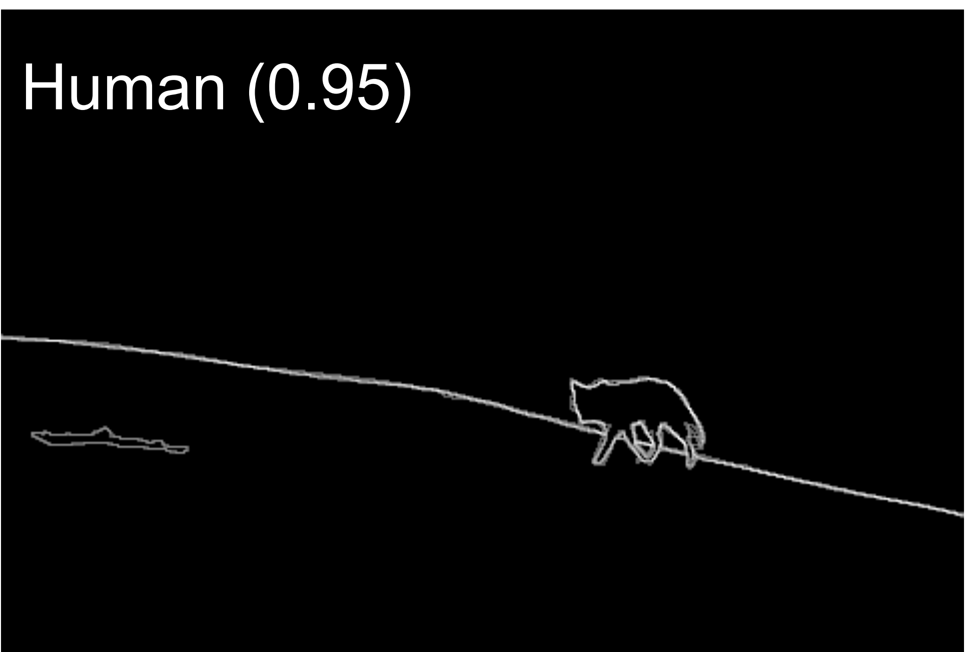


Information Leakage across Orientations

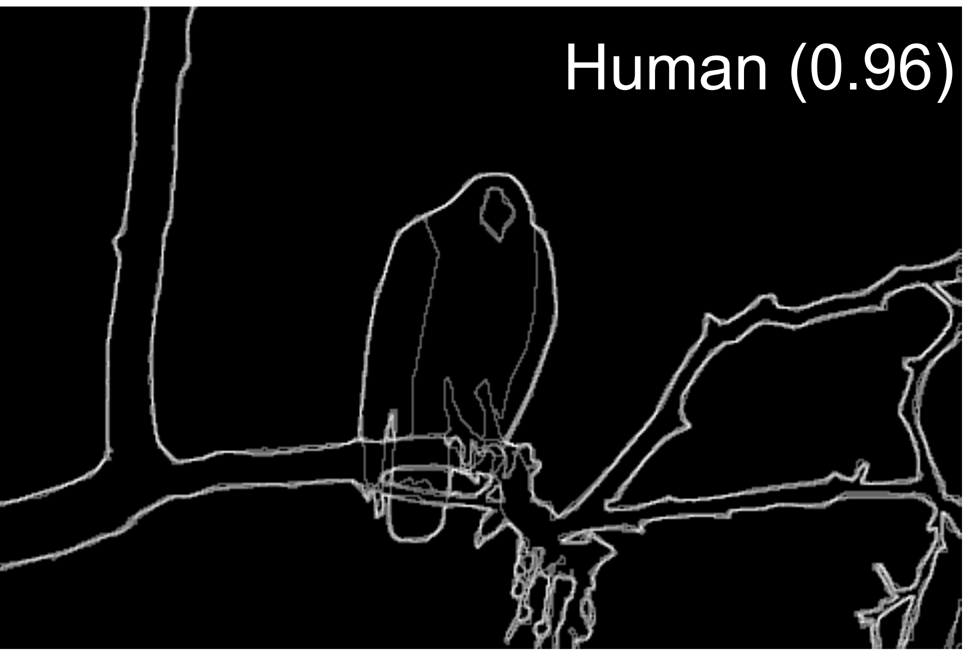
[Learning to Detect Natural Image Boundaries Using Brightness and Texture](#)
[Contour Detection and Hierarchical Image Segmentation](#)

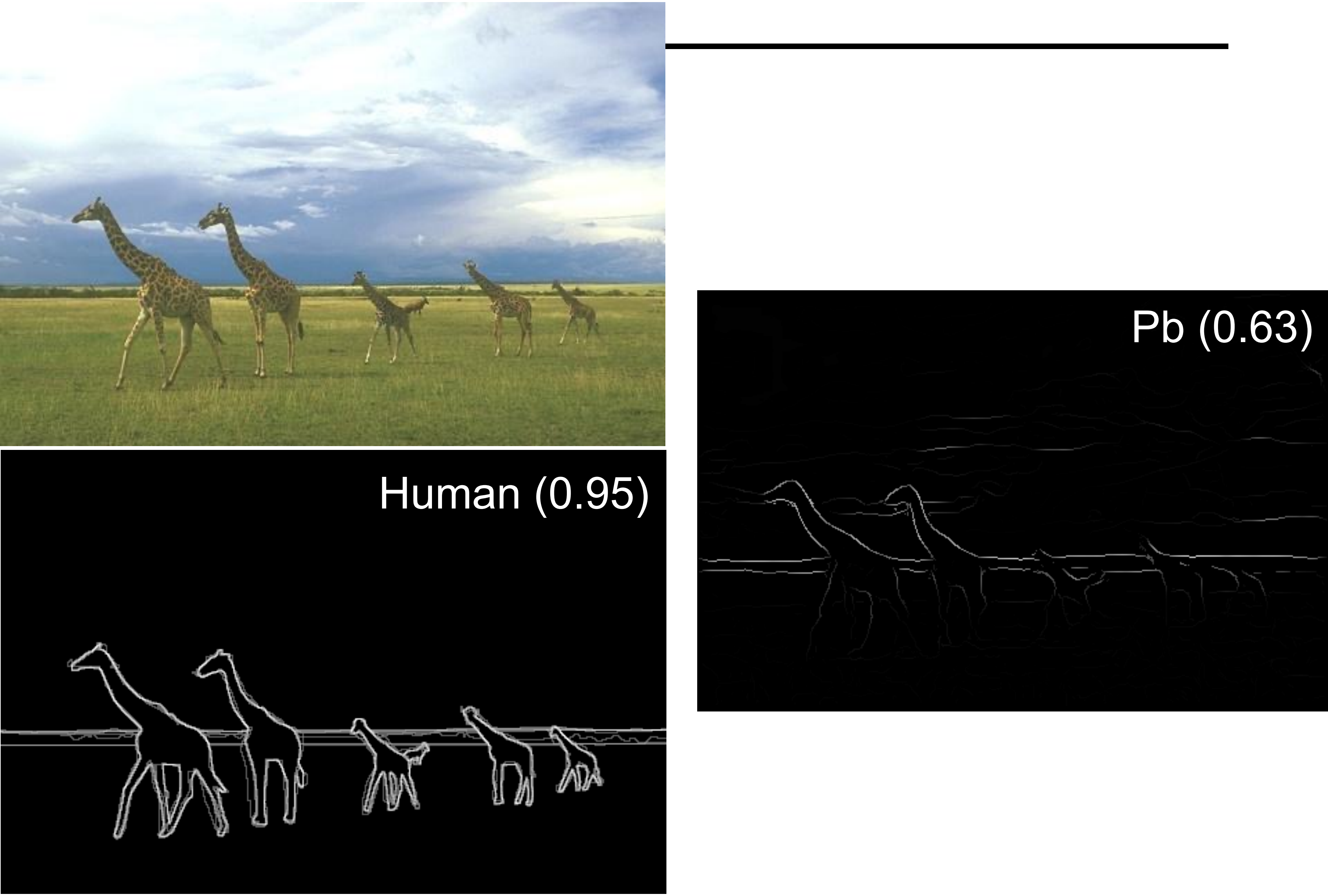
Results

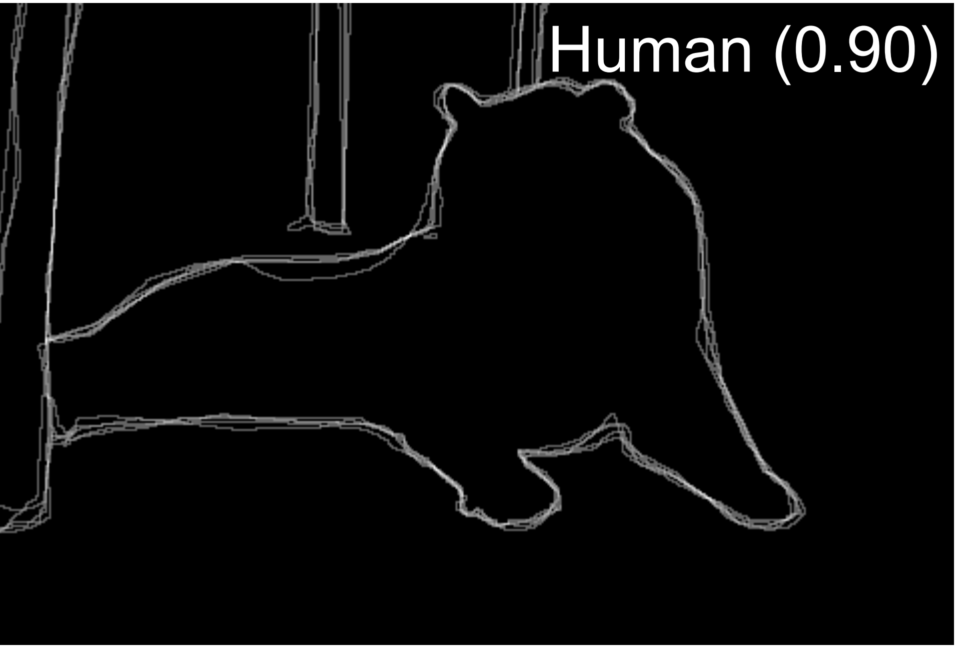




Results



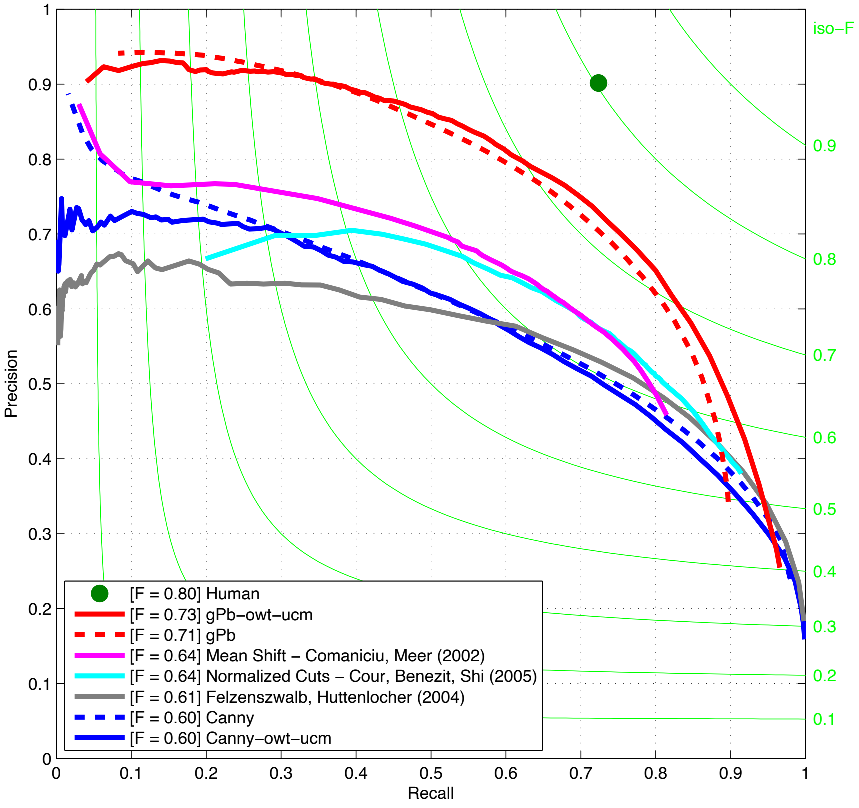

Human (0.95)
Pb (0.63)



For more:
http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/bench/html/108082-color.html

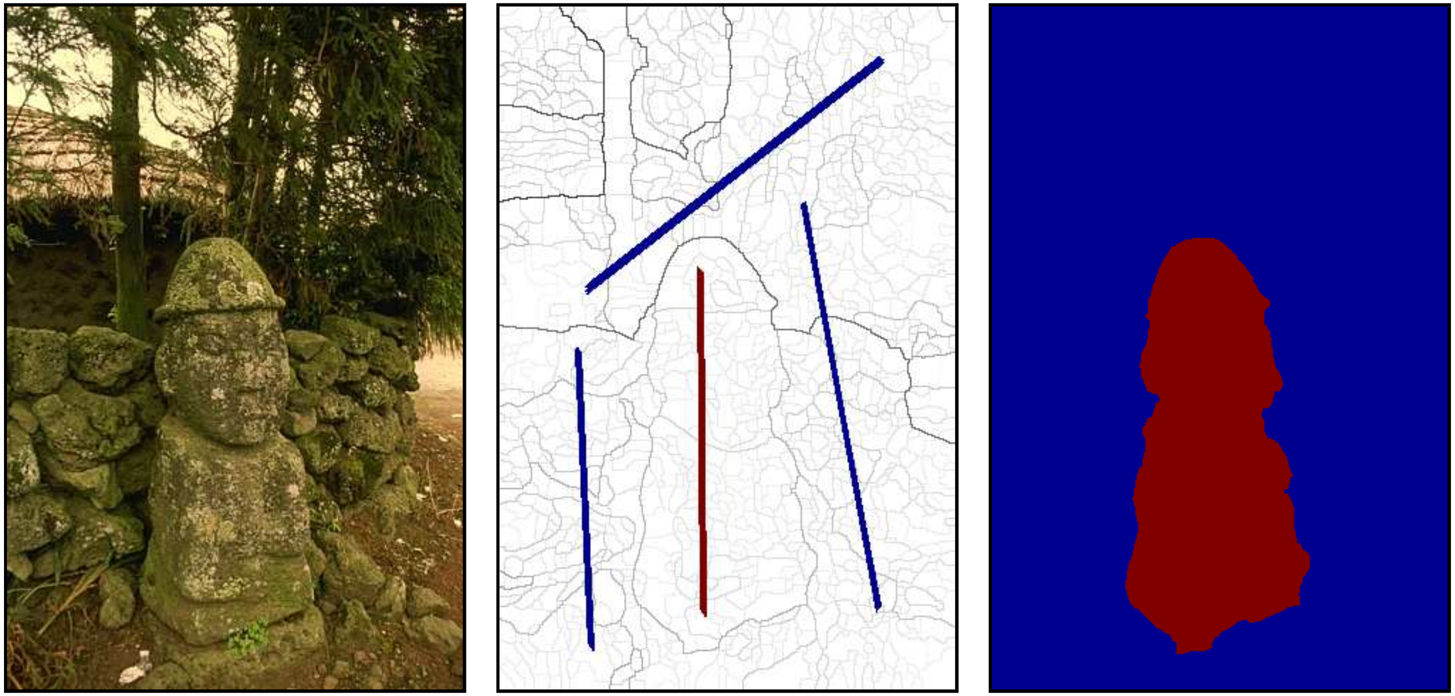
Empirical Research

	BSDS300		
	ODS	OIS	AP
Human	0.79	0.79	—
gPb-owt-ucm	**0.71**	**0.74**	**0.73**
[34] Mean Shift	0.63	0.66	0.54
[33] NCuts	0.62	0.66	0.43
Canny-owt-ucm	0.58	0.63	0.58
[32] Felz-Hutt	0.58	0.62	0.53
[31] SWA	0.56	0.59	0.54
Quad-Tree	0.37	0.39	0.26
gPb	0.70	0.72	0.66
Canny	0.58	0.62	0.58



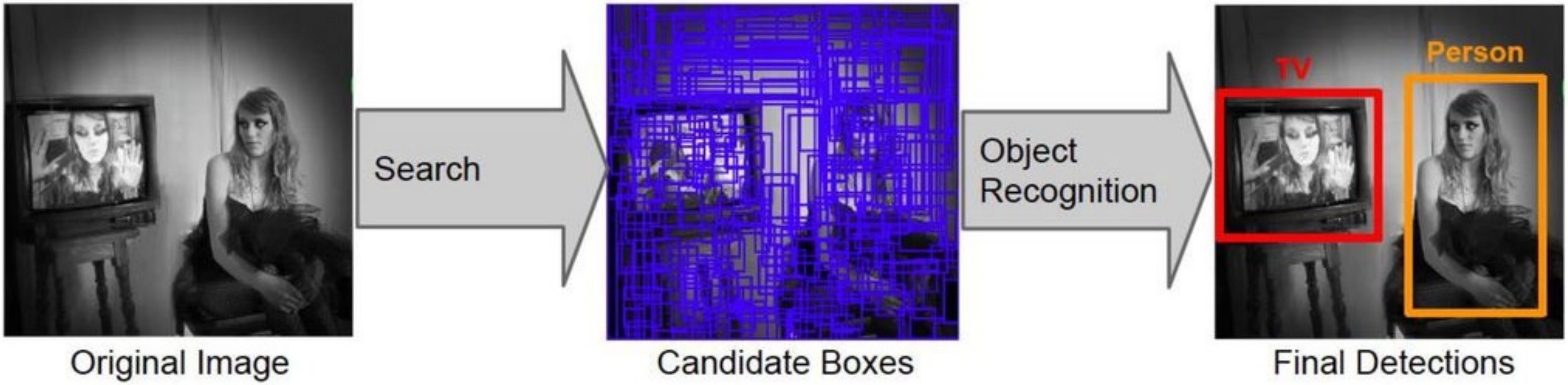
[Contour Detection and Hierarchical Image Segmentation]()

Applications: Interactive Segmentation



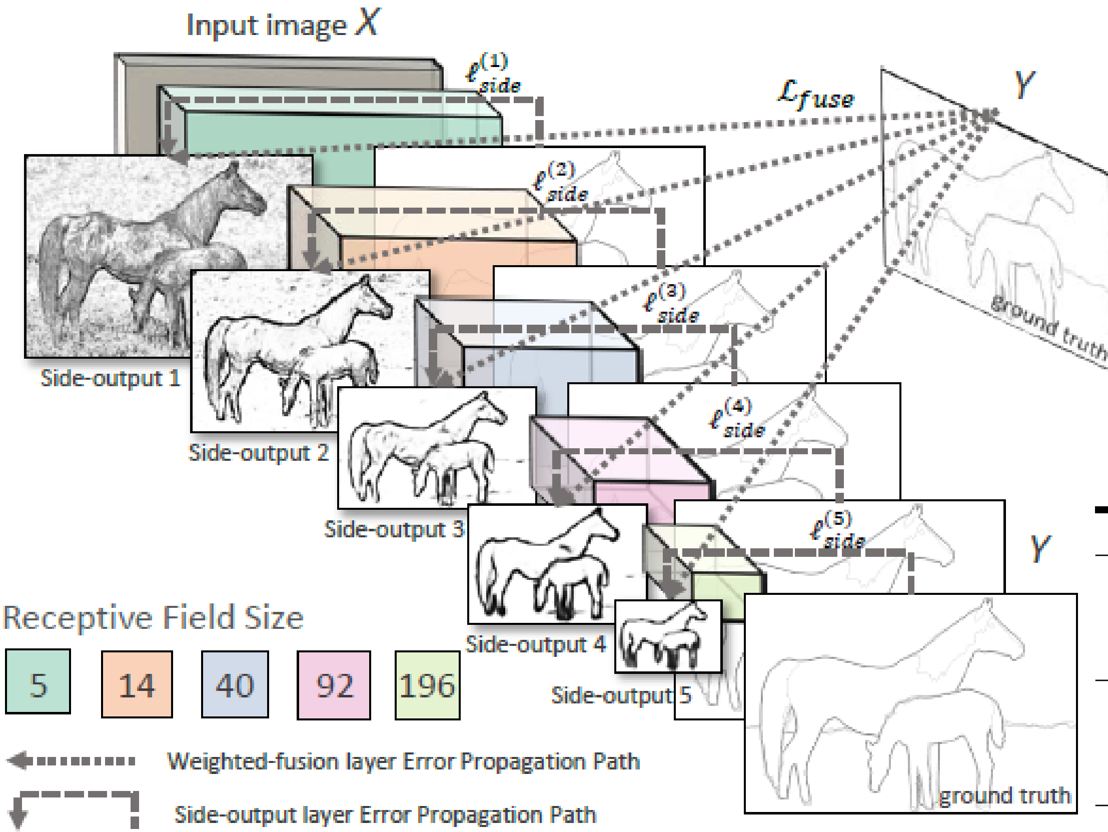
[Contour Detection and Hierarchical Image Segmentation. P. Arbeláez et al. PAMI 2010](#)

Applications: Pre-processing for Object Detection



J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders, Selective Search for Object Recognition, IJCV 2013

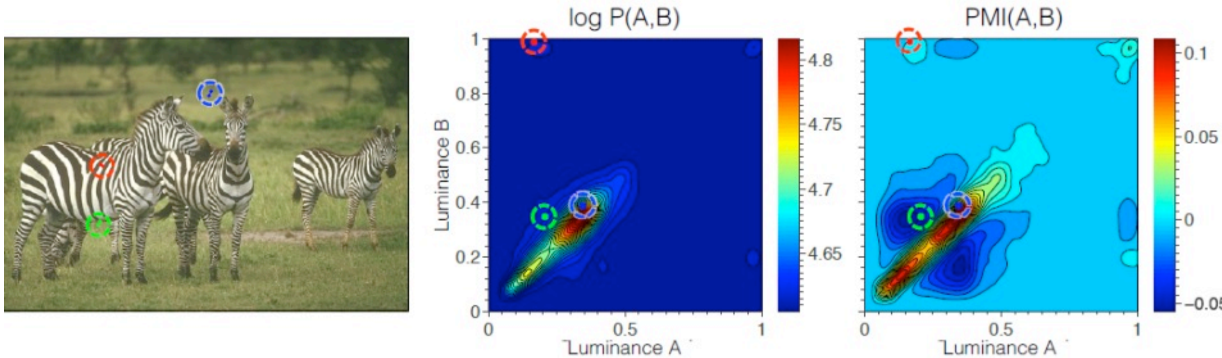
Holistically nested edge detection



	ODS	OIS	AP	FPS
Human	.80	.80	-	-
Canny	.600	.640	.580	15
Felz-Hutt [9]	.610	.640	.560	10
BEL [5]	.660∗	-	-	1/10
gPb-owt-ucm [1]	.726	.757	.696	1/240
Sketch Tokens [24]	.727	.746	.780	1
SCG [31]	.739	.758	.773	1/280
SE-Var [6]	.746	.767	.803	2.5
OEF [13]	.749	.772	.817	-
DeepNets [21]	.738	.759	.758	1/5†
N4-Fields [10]	.753	.769	.784	1/6†
DeepEdge [2]	.753	.772	.807	$1/10^3$†
CSCNN [19]	.756	.775	.798	-
DeepContour [34]	.756	.773	.797	1/30†
HED (ours)	**.782**	**.804**	**.833**	2.5†, 1/12

https://arxiv.org/pdf/1504.06375.pdf

Crisp Boundary Detection using Pointwise Mutual Information (Isola et al. ECCV 2014)



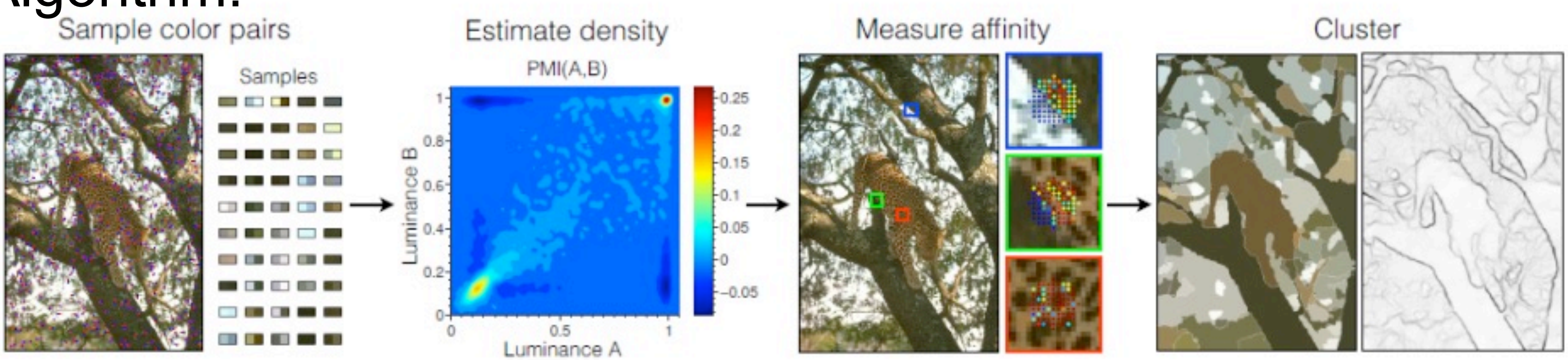
$$\mathrm{PMI}_\rho(A,B) = \log \frac{P(A,B)^\rho}{P(A)P(B)}$$

Pixel combinations that are unlikely to be together are edges

Algorithm:

Kernel density estimation

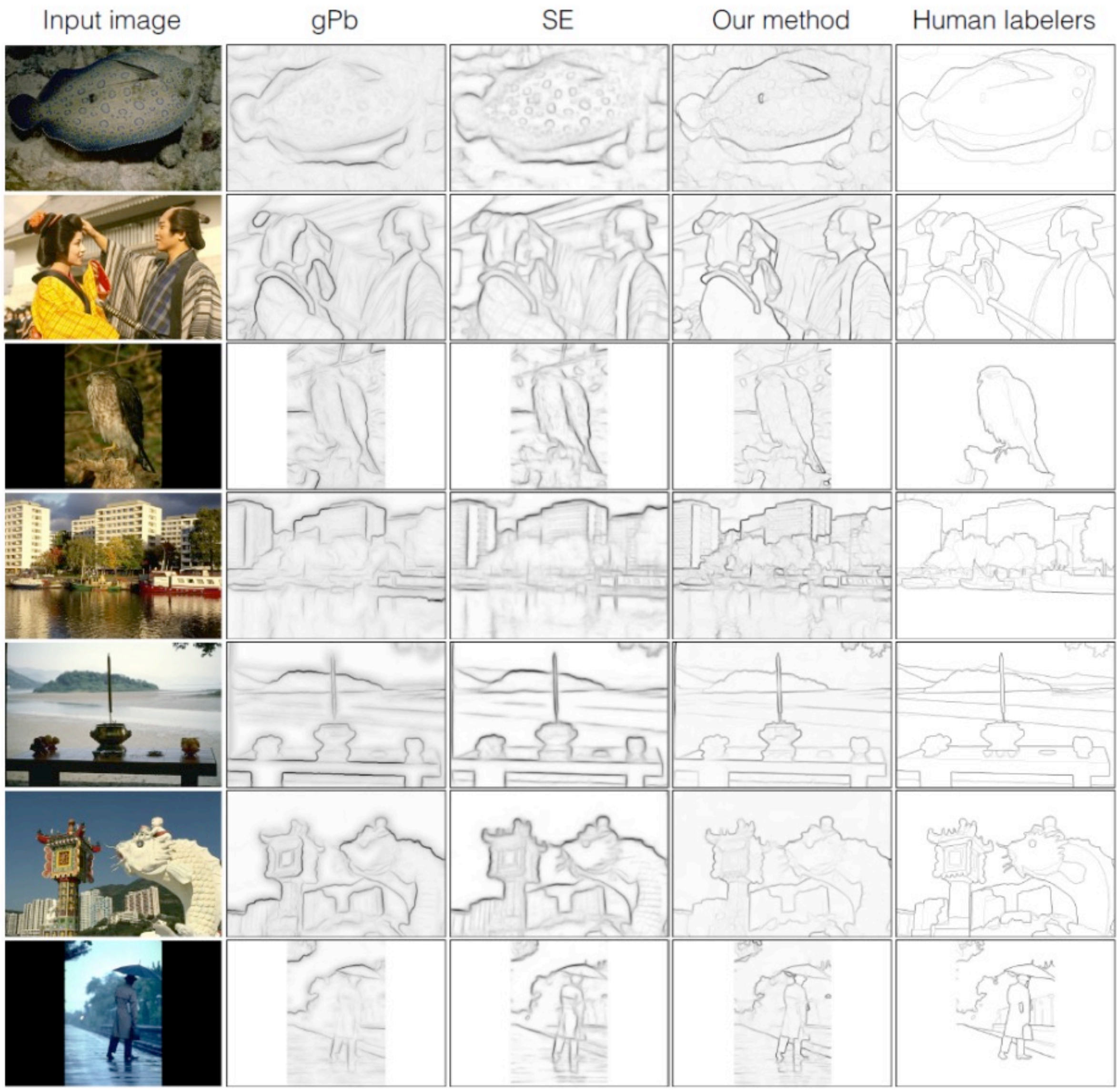
Spectral clustering



http://web.mit.edu/phillipi/www/publications/crisp_boundaries.pdf

Crisp Boundary Detection using Pointwise Mutual Information

Algorithm	ODS	OIS	AP
Canny [14]	0.60	0.63	0.58
Mean Shift [36]	0.64	0.68	0.56
NCuts [37]	0.64	0.68	0.45
Felz-Hutt [38]	0.61	0.64	0.56
gPb [1]	0.71	0.74	0.65
gPb-owt-ucm [1]	0.73	0.76	0.73
SCG [9]	**0.74**	0.76	0.77
Sketch Tokens [7]	0.73	0.75	0.78
SE [8]	**0.74**	0.76	0.78
Our method – SS, color only	0.72	0.75	0.77
Our method – SS	0.73	0.76	**0.79**
Our method – MS	**0.74**	**0.77**	0.78



State of edge detection

Local edge detection is mostly solved

- Intensity gradient, color, texture
- HED on BSDS 500 is near human performance

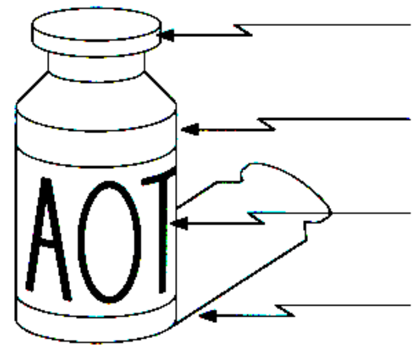
Some room for improvement by taking advantage of higher-level knowledge (e.g., objects)

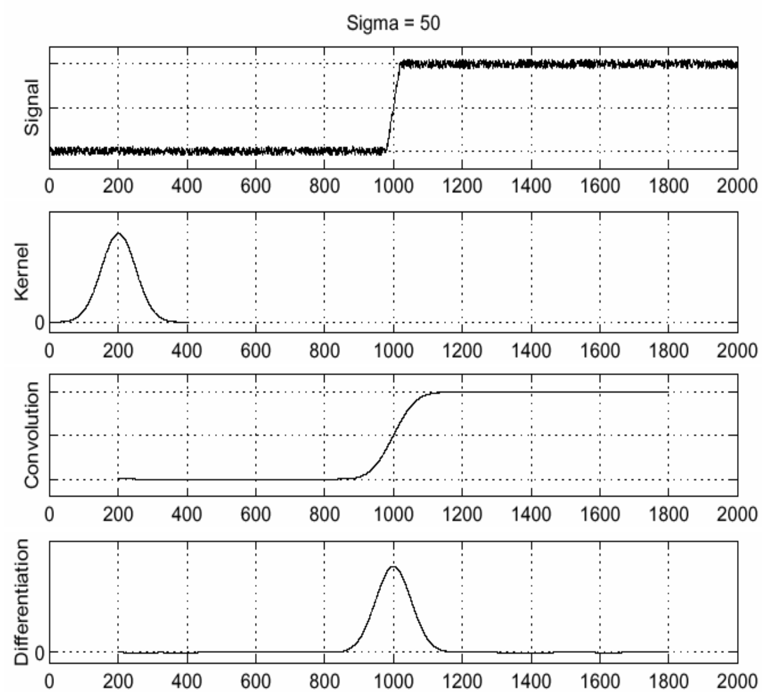
Still hard to produce all objects within a small number of regions

Are Edges an Input or an Output?

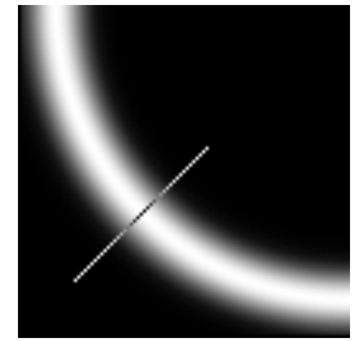
Source: Gregory, R. L. (1970). The Intelligent Eye. New York, NY: McGraw-Hill Paperbacks.

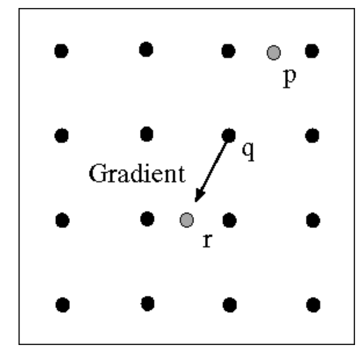
Recap



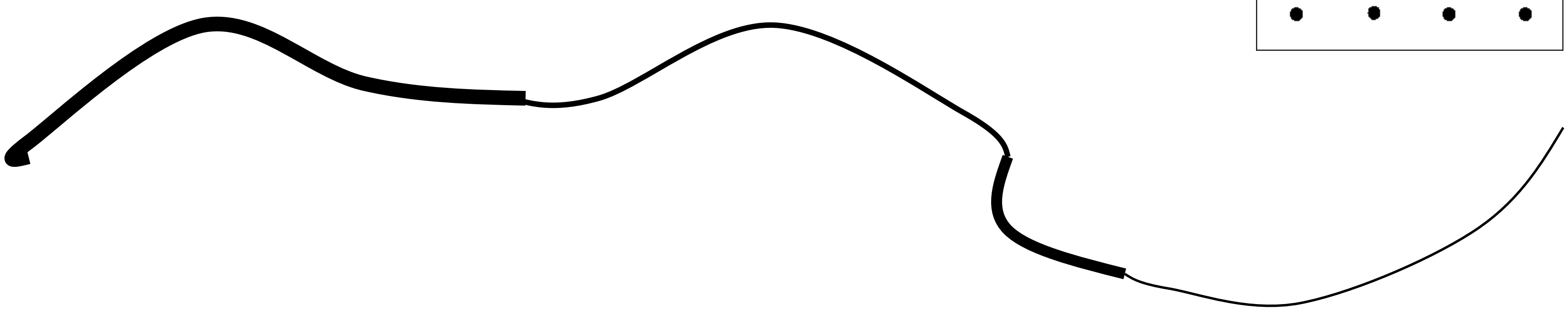


Sigma = 50
Signal
Kernel
Convolution
Differentiation
0
200
400
600
800
1000
1200
1400
1600
1800
2000

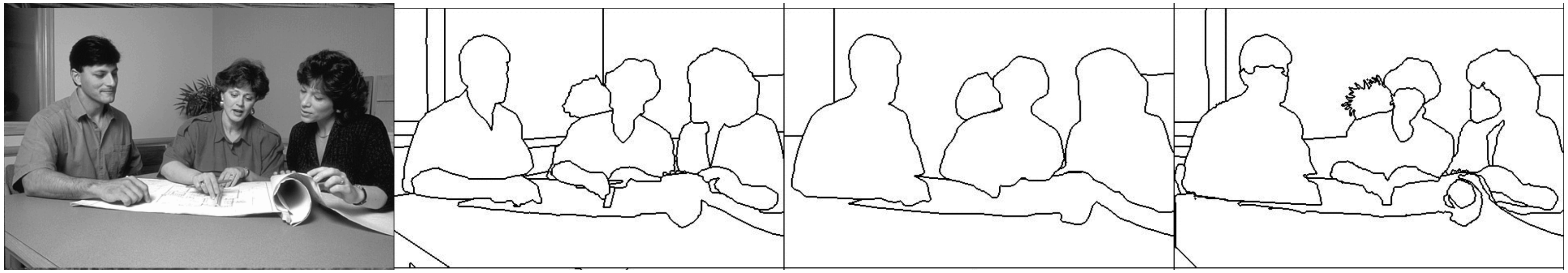


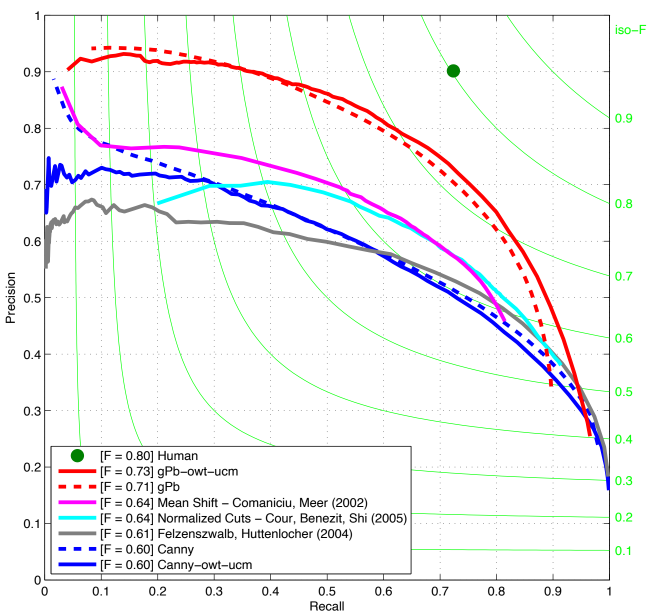


p
q
Gradient
r



Recap





	BSDS300		
	ODS	OIS	AP
Human	0.79	0.79	–
gPb-owt-ucm	**0.71**	**0.74**	**0.73**
[34] Mean Shift	0.63	0.66	0.54
[33] NCuts	0.62	0.66	0.43
Canny-owt-ucm	0.58	0.63	0.58
[32] Felz-Hutt	0.58	0.62	0.53
[31] SWA	0.56	0.59	0.54
Quad-Tree	0.37	0.39	0.26
gPb	0.70	0.72	0.66
Canny	0.58	0.62	0.58