CSC 589 Introduction to Computer Vision

Lecture 14
Boundary Detection

Prof. Bei Xiao

Spring, 2014

American University

Edge detection

- **Goal:** Identify sudden changes (discontinuities) in an image
  - Intuitively, most semantic and shape information from the image can be encoded in the edges
  - More compact than pixels
- **Ideal:** artist's line drawing (but artist is also using object-level knowledge)

Source: D. Lowe

Canny Edge Operator

1. **Noise reduction:** Filter image with x, y derivatives of Gaussian
2. **Intensity gradients:** Find magnitude and orientation of gradient
3. **Non-maximum suppression:**
   - Thin multi-pixel wide “ridges” down to single pixel width
4. **Thresholding and linking** (hysteresis):
   - Define two thresholds: low and high
   - Use the high threshold to start edge curves and the low threshold to continue them

- Python: cv2.Canny(img, lo, hi),
- skimage.filter.canny
- canny(img, sigma)

Original image

Gradient magnitude

(a) Smoothed

(b) Gradient magnitudes

Gradient magnitude

Sobel Filter on x and y directions

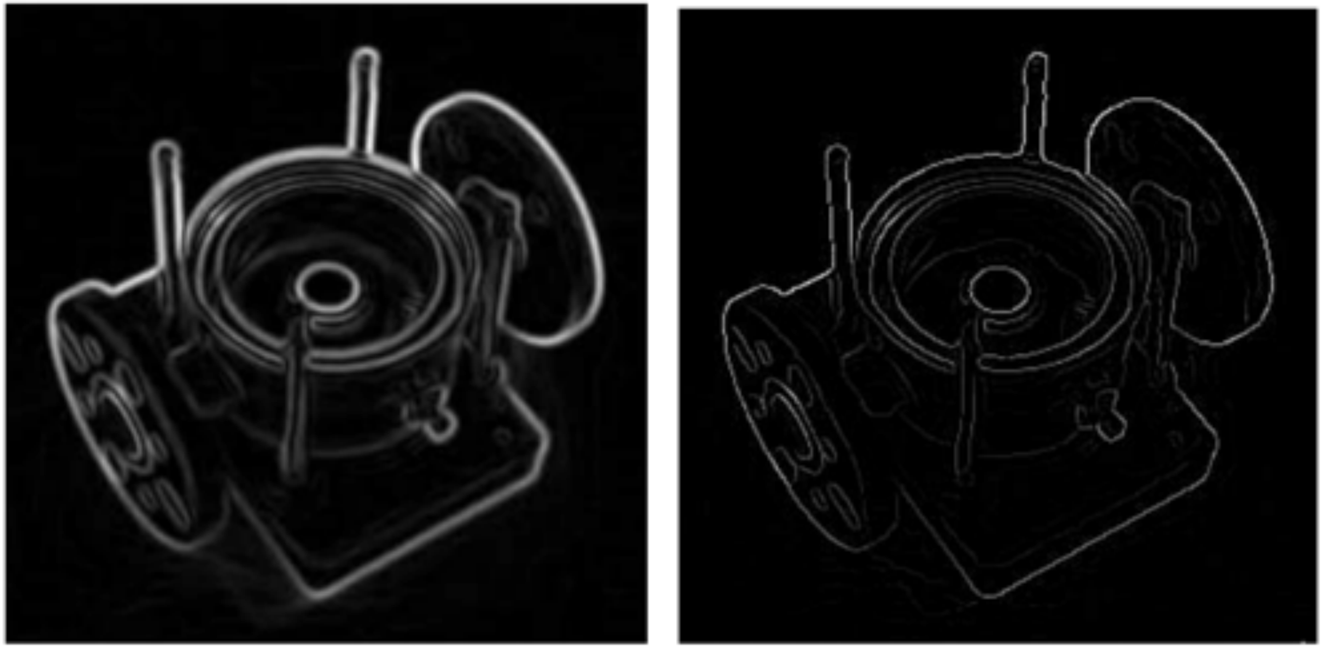
$$K_{\mathrm{GX}} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$K_{\mathrm{GY}} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

$$|G| = \sqrt{G_{\mathrm{x}}^2 + G_{\mathrm{y}}^2}$$

$$|G| = |G_{\mathrm{x}}| + |G_{\mathrm{y}}|$$

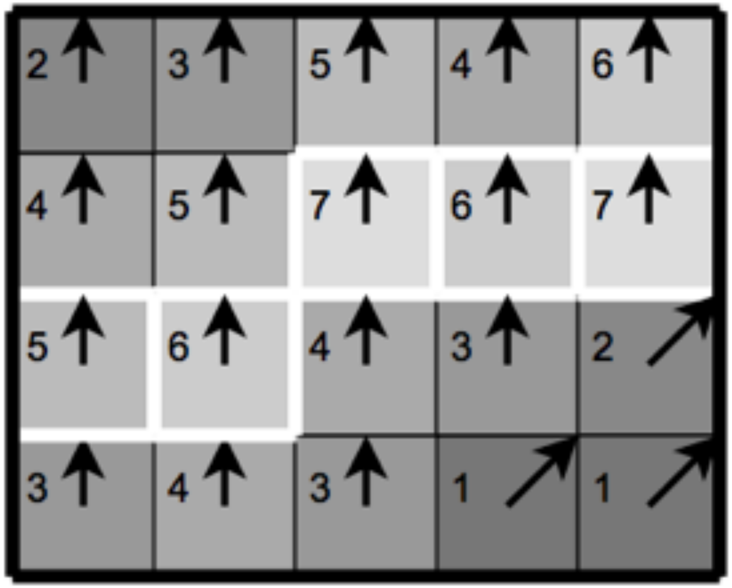
Non-maximum suppression



(a) Gradient values (b) Edges after non-maximum suppression

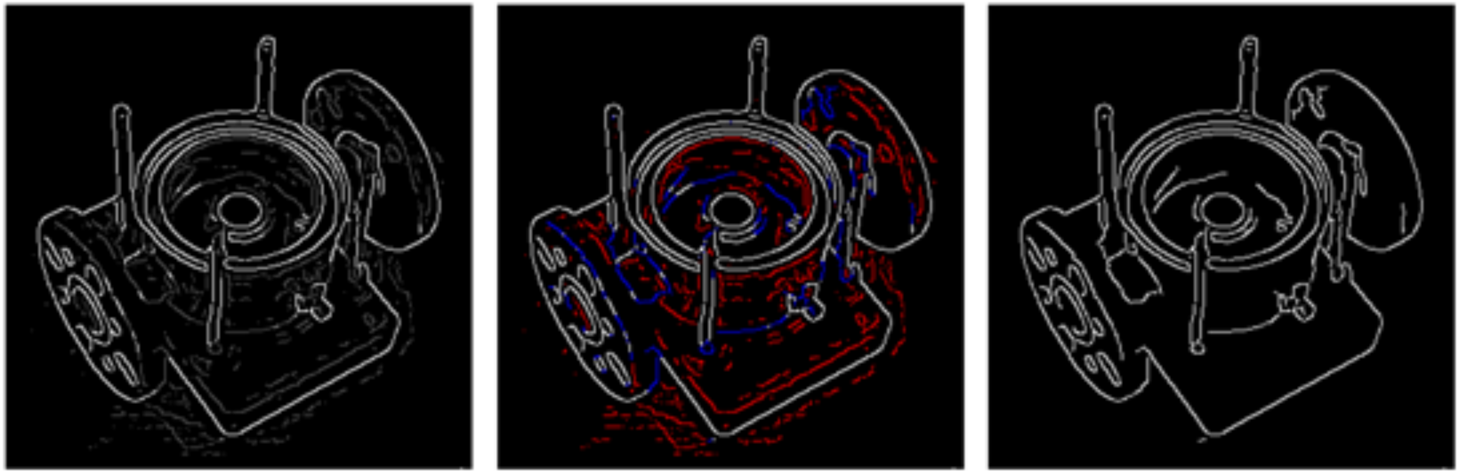
http://www.cse.iitd.ernet.in/~pkalra/csl783/canny.pdf

Non-maximum suppression



http://www.cse.iitd.ernet.in/~pkalra/csl783/canny.pdf

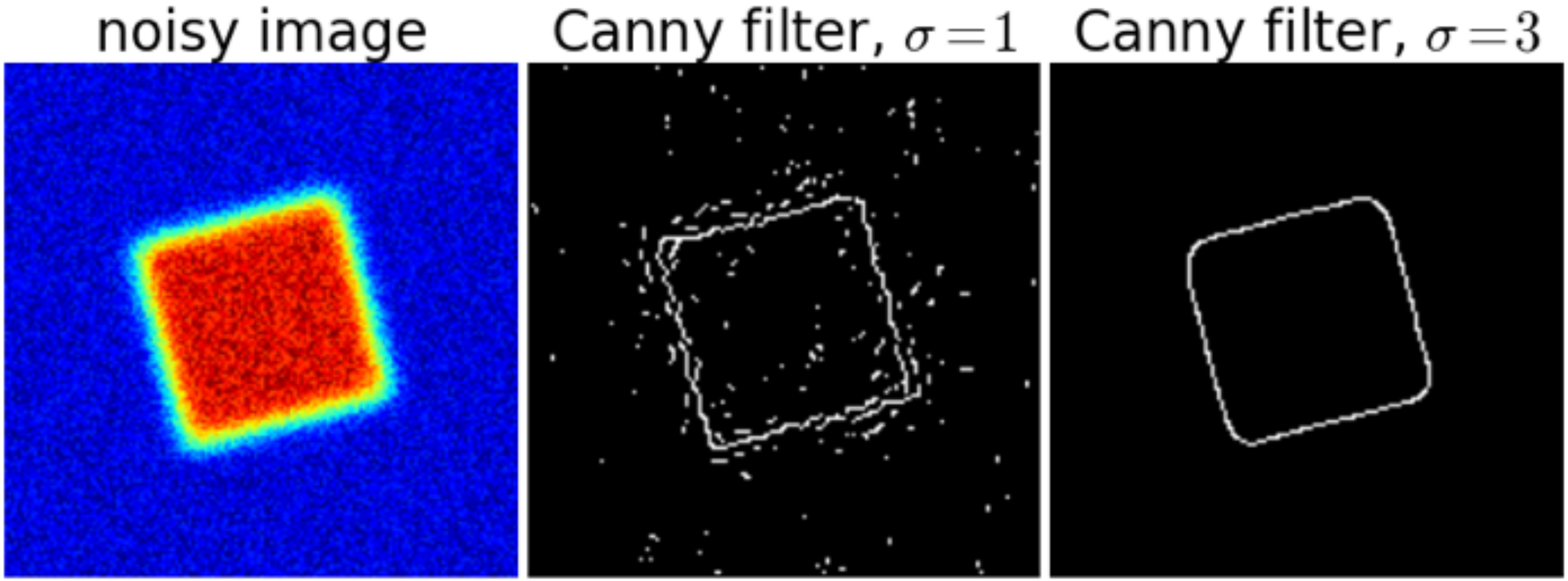
Hysteresis



(a) Double thresholding (b) Edge tracking by hysteresis (c) Final output

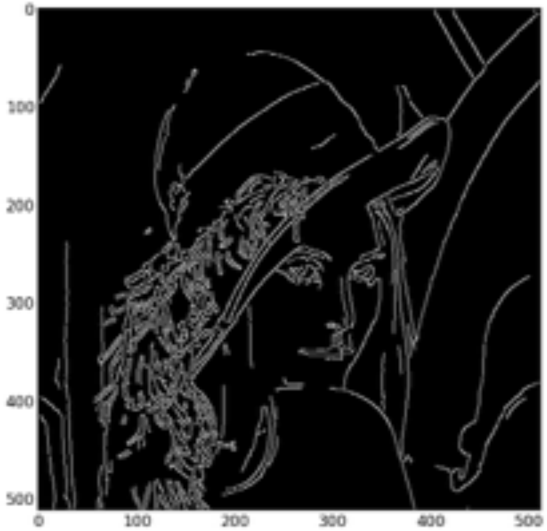
http://www.cse.iitd.ernet.in/~pkalra/csl783/canny.pdf

Canny Edge Detector

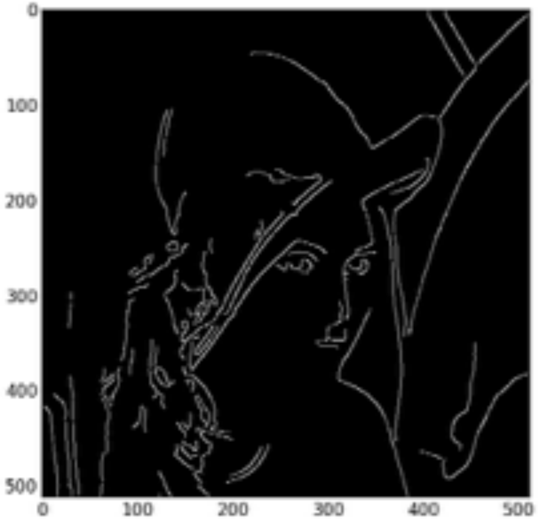


Effect of σ (Gaussian kernel spread/size)

original



Canny with $\sigma = 1$



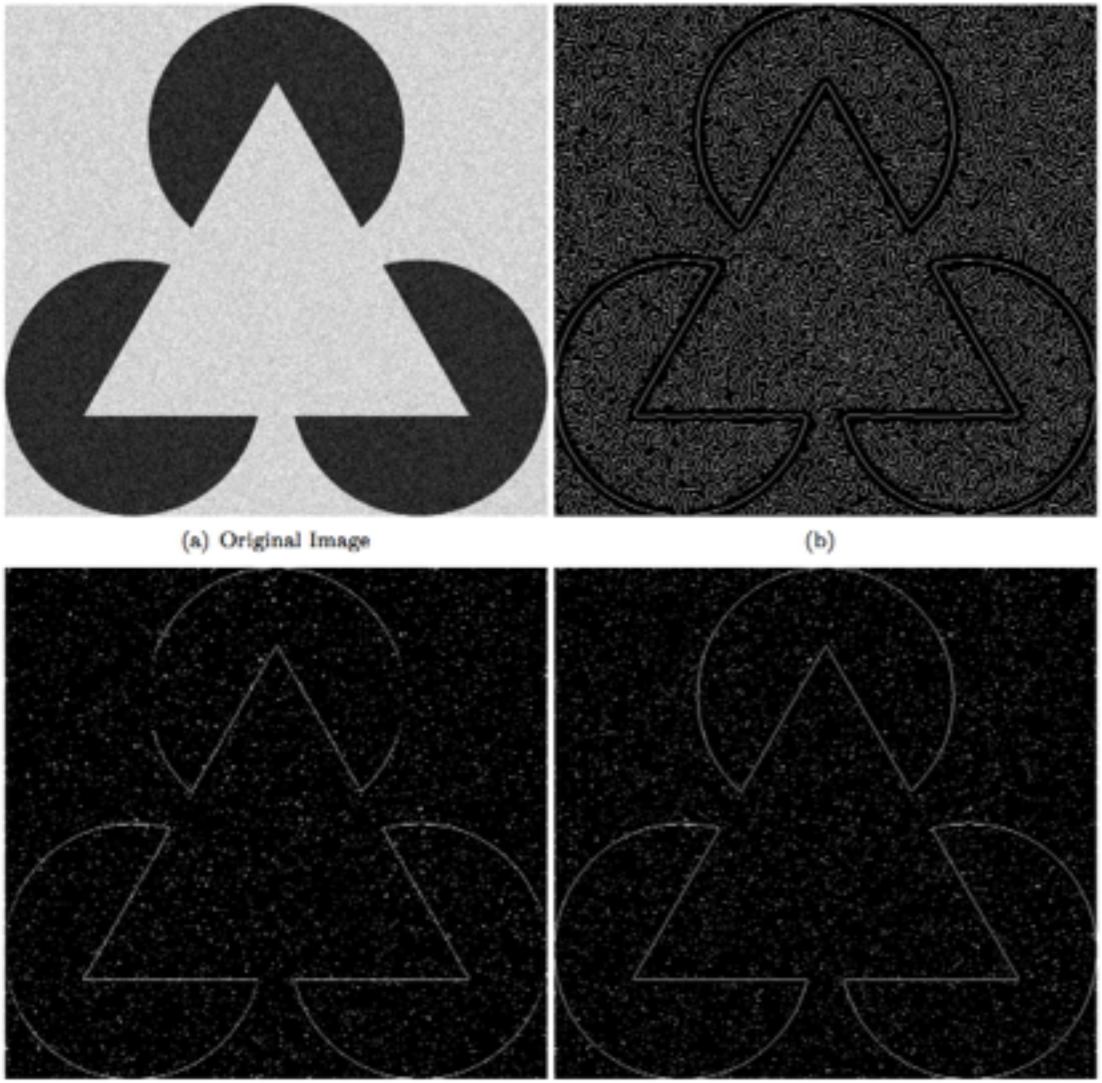
Canny with $\sigma = 2$

The choice of σ depends on desired behavior

- large σ detects large scale edges
- small σ detects fine features

Source: S. Seitz

Quiz: which image (b), (c), and (d) is the result of applying canny edge detector? Explain your answer.



(a) Original Image (b)

(c) (d)

What is difference between Boundary and edges?

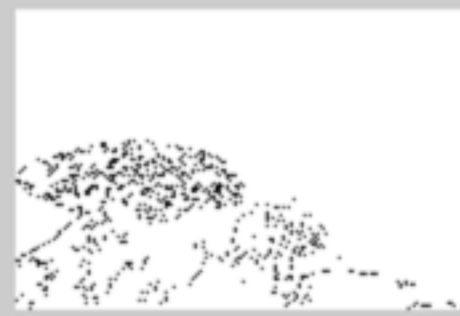
Boundary: High-level object information. Whether a pixel belongs to an object or not.

Edges: Low-level information, sudden change in intensity values.

Input Image

Crispy Boundary

Canny

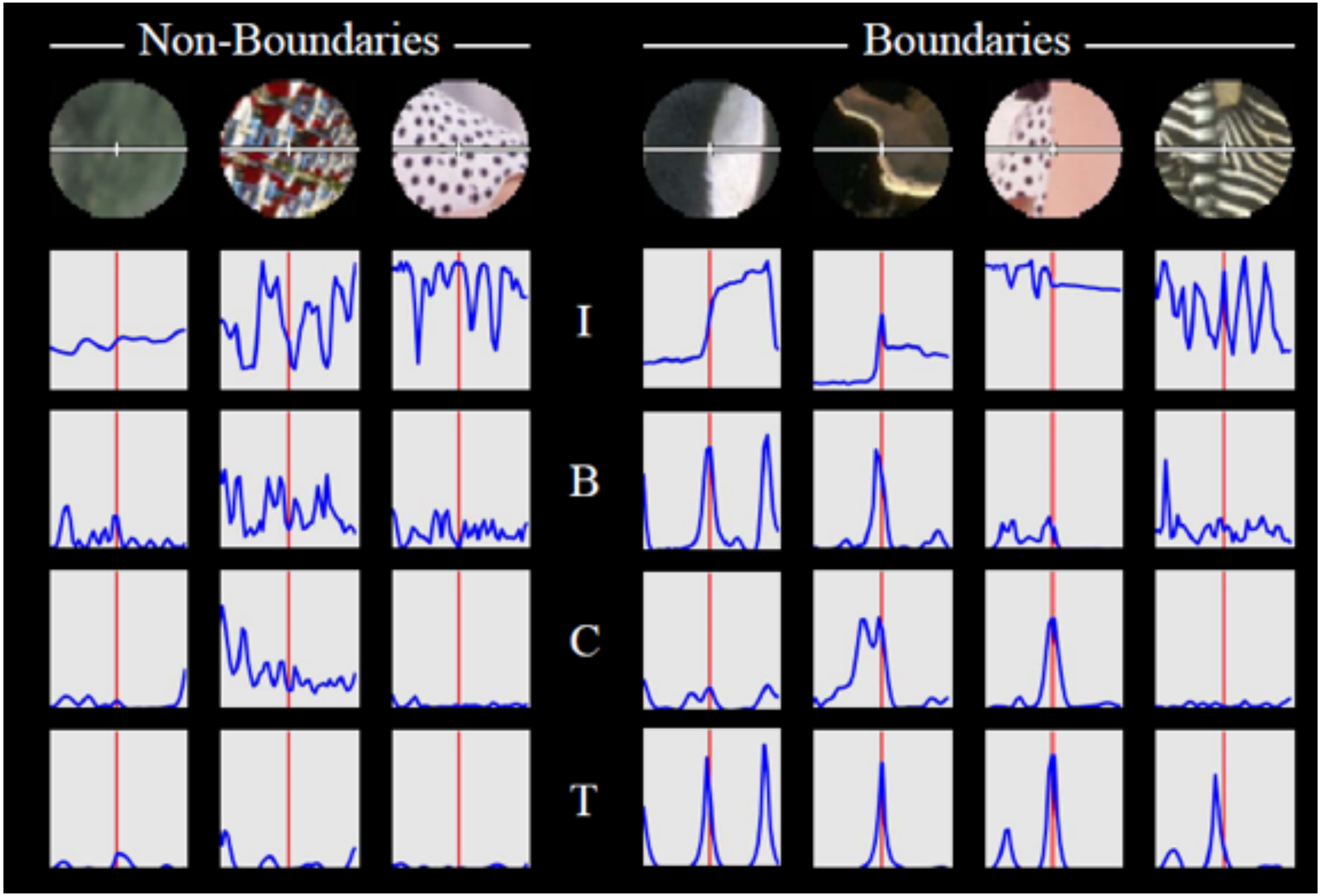


Where do humans see boundaries?

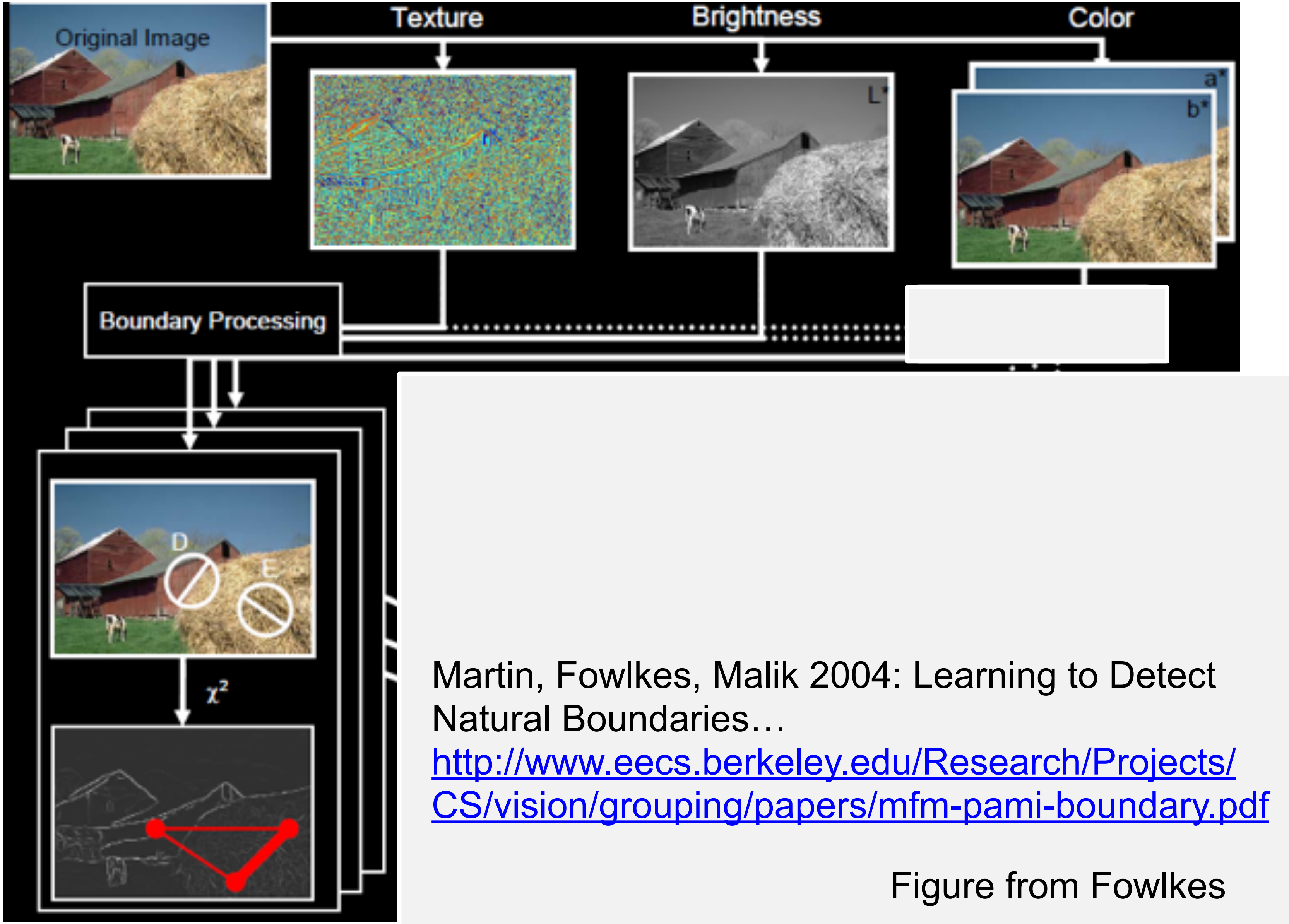
image | human segmentation | gradient magnitude

- Berkeley segmentation database:
  http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/

Look for changes in texture, color, brightness



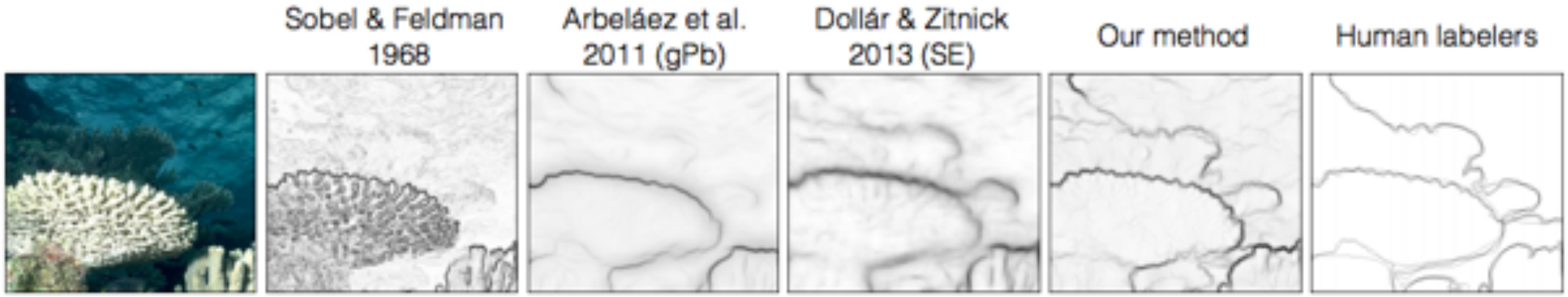
pB boundary detector



Martin, Fowlkes, Malik 2004: Learning to Detect Natural Boundaries…
http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/papers/mfm-pami-boundary.pdf

Figure from Fowlkes

Edge detection vs. boundary detection



1. Classical methods use local derivative filters with fixed scales and only a few orientations. Tend to emphasize small and unimportant edges.
2. Contemporary methods uses multiple scales, multiple feature from image patches (color, textures, intensity). Using statistical methods (give each pixel a probability of being a boundary) to learn boundaries.
3. Isola et al. (2014) uses mutual information between pixels to detect boundary.

Key observation: *Pixels belonging to the same object have higher statistical association than pixels belonging to different objects.*

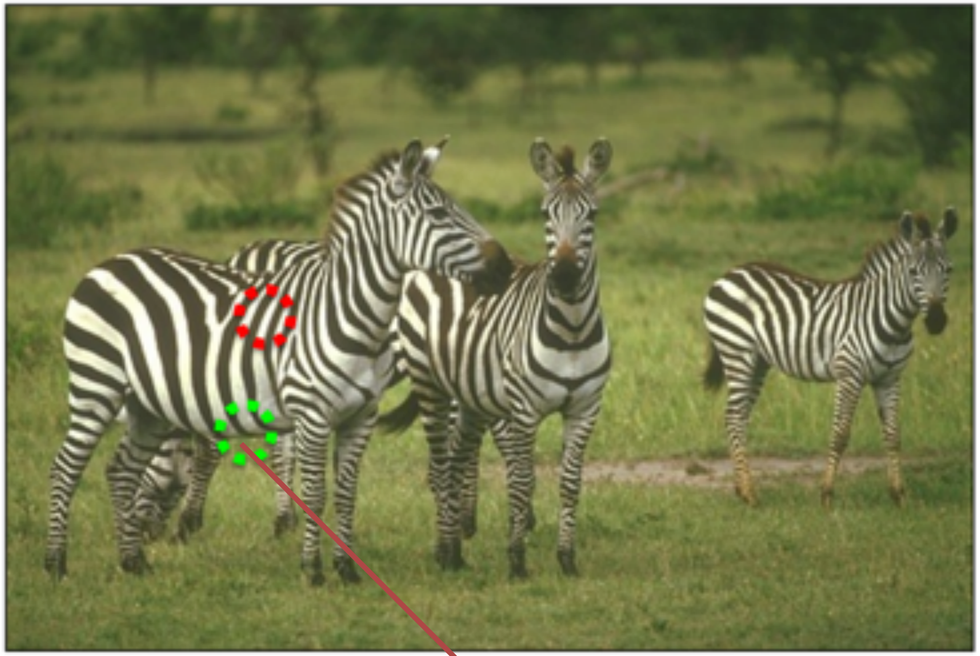
Slide courtesy from Philip Isola

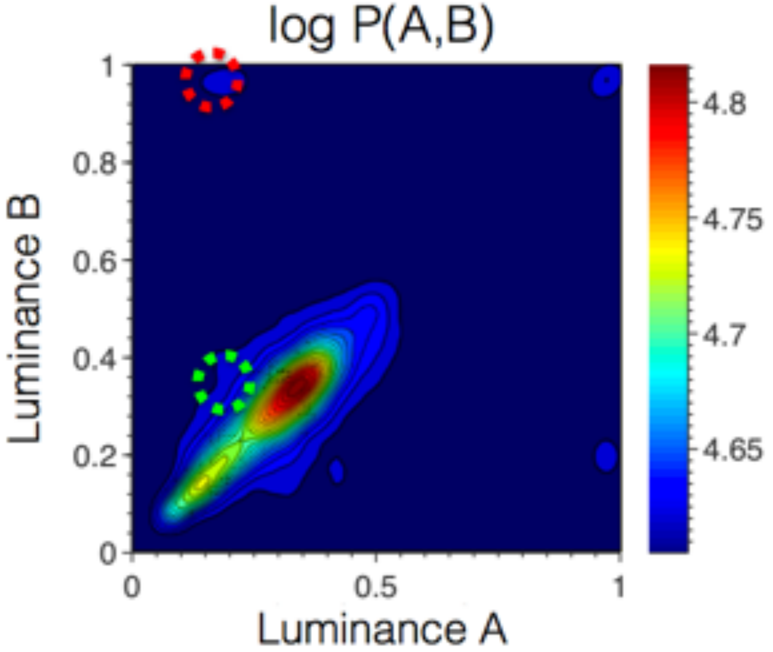
Point-wise mutual information reveals object structure

Above, black-next-to-white occurs over and over again. This pattern shows up in the image's statistics as a *suspicious coincidence* — these colors must be part of the same object!

How do we distinguish the red and the green patches?

Slide courtesy from Philip Isola





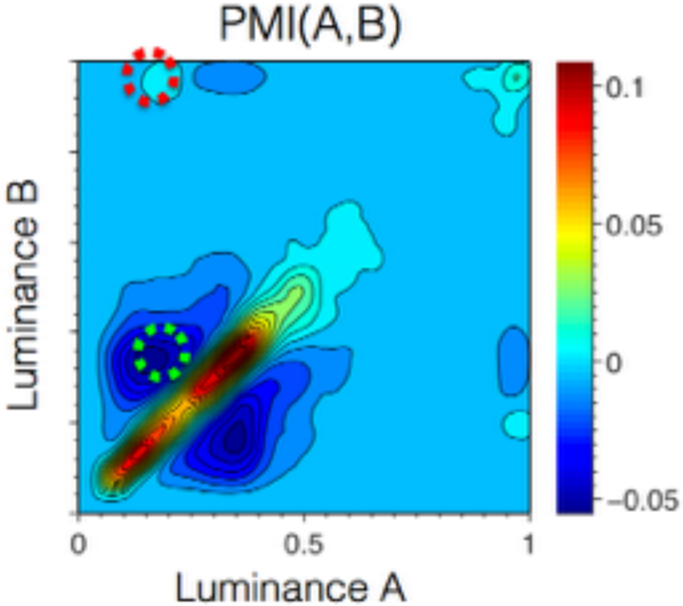
Object Boundary

A: Green pixel

B: Black pixel

$P(A, B) =$ how often each color A occurs next to each color B *within this image.*

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



Pointwise mutual information (PMI)

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

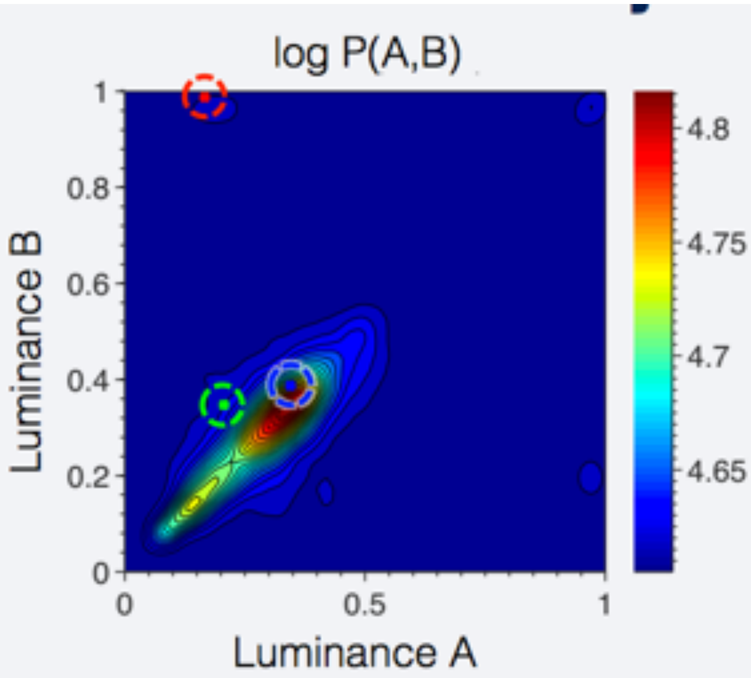
Use PMI as affinity measure for affinity-based pixel grouping.

How much more likely is observing A given that we saw B in the same local region, compared to the base rate of observing A in the image.

Joint distribution of two pixels

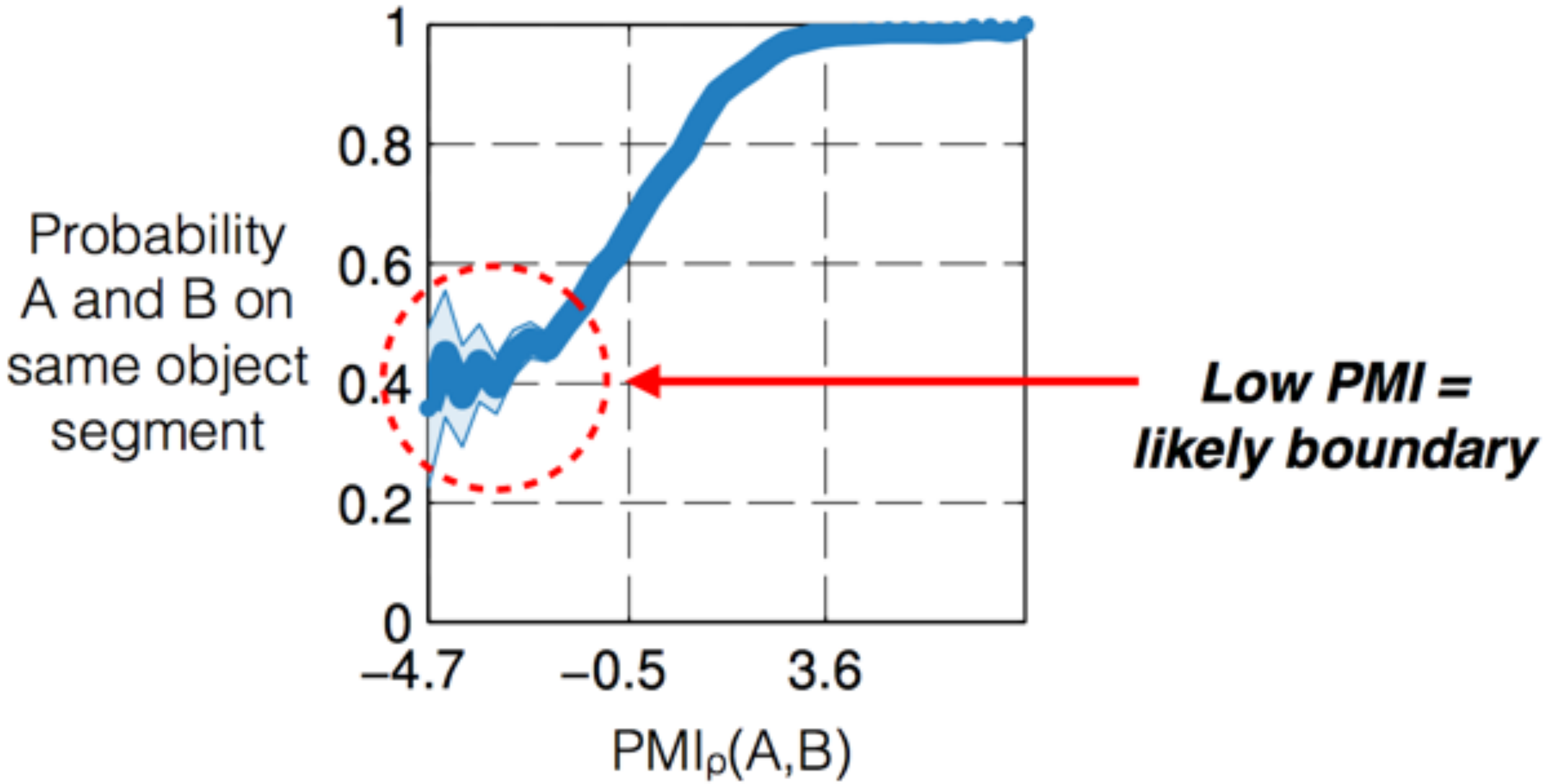
Above, black-next-to-white occurs over and over again. This pattern shows up in the image's statistics as a *suspicious coincidence* — these colors must be part of the same object!

$$P(A,B) = \frac{1}{Z}\sum_{d=d_0}^{\infty} w(d)p(A,B;d),$$



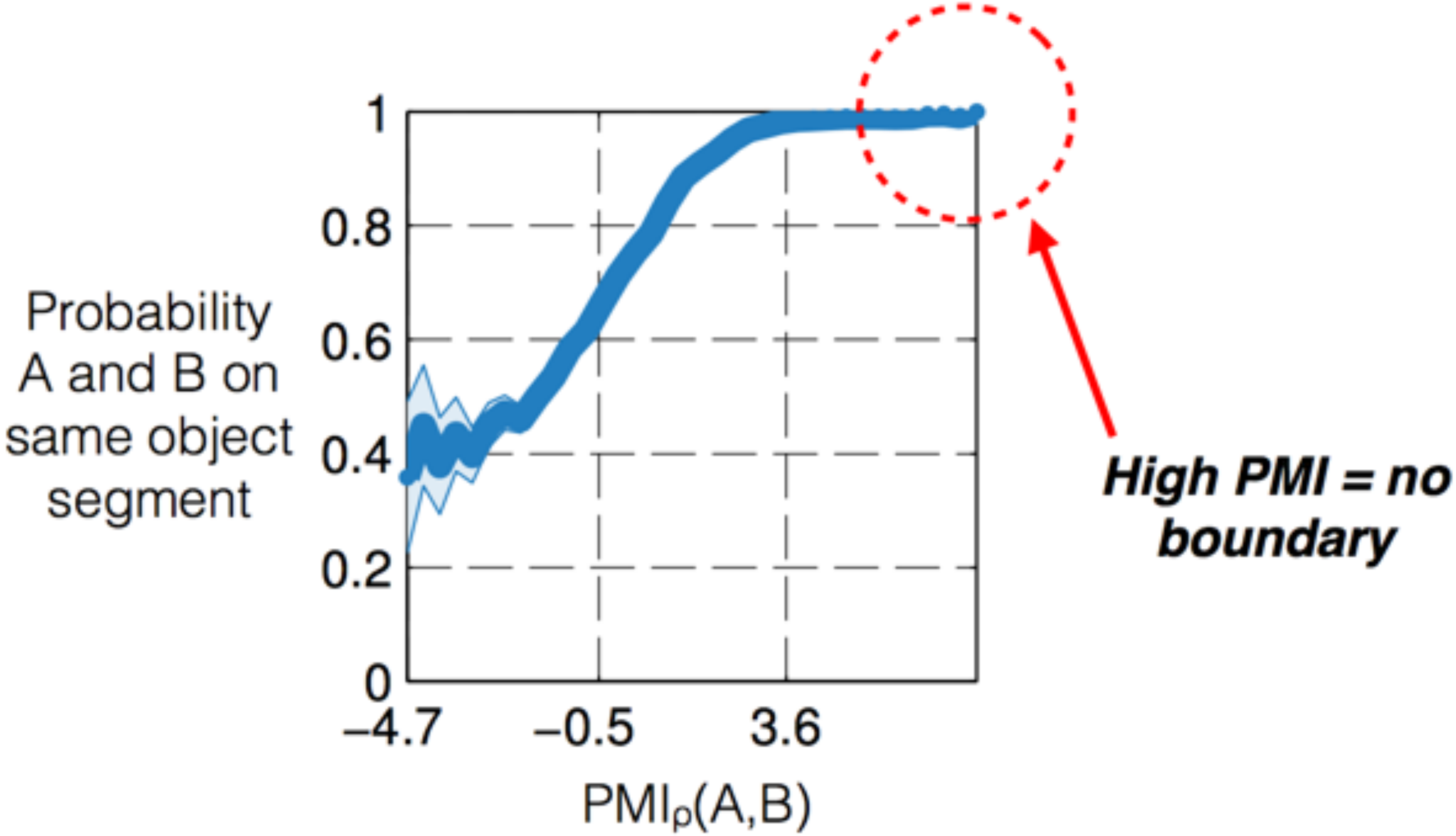
We measure how often each color A occurs next to each color B *within the image.*

Is PMI informative about object boundaries?

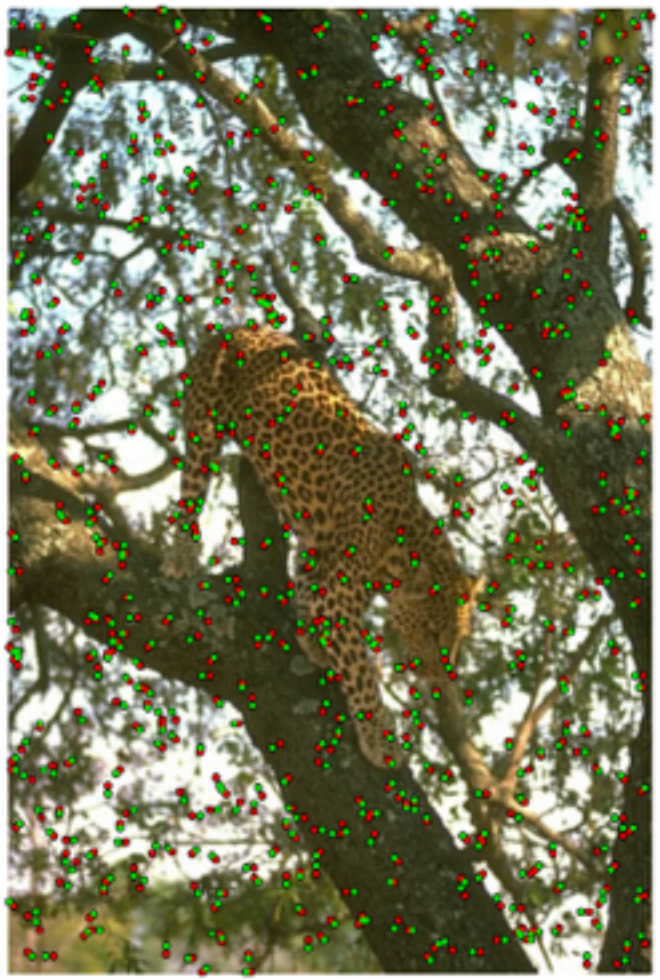


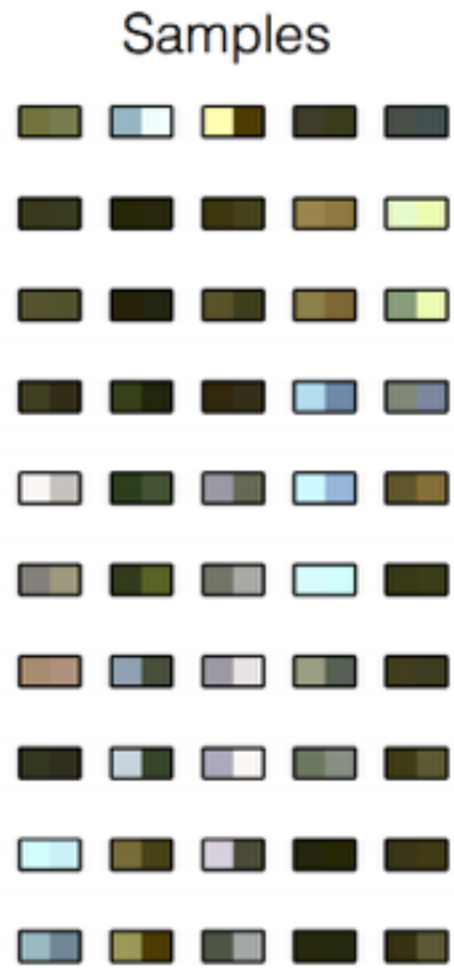
Is PMI informative about object boundaries?

Yes!

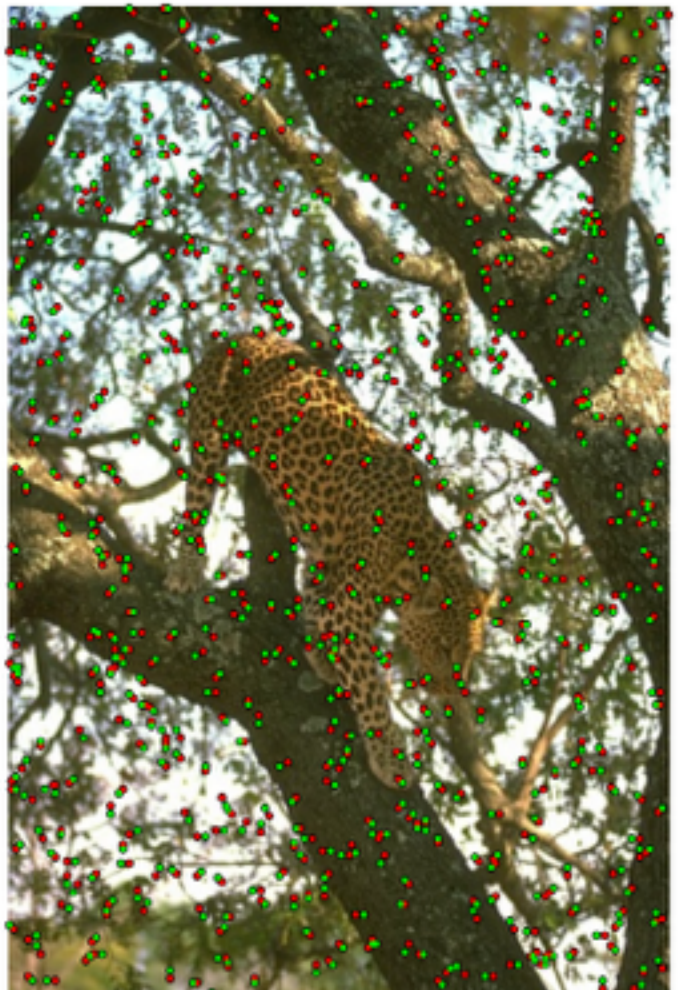


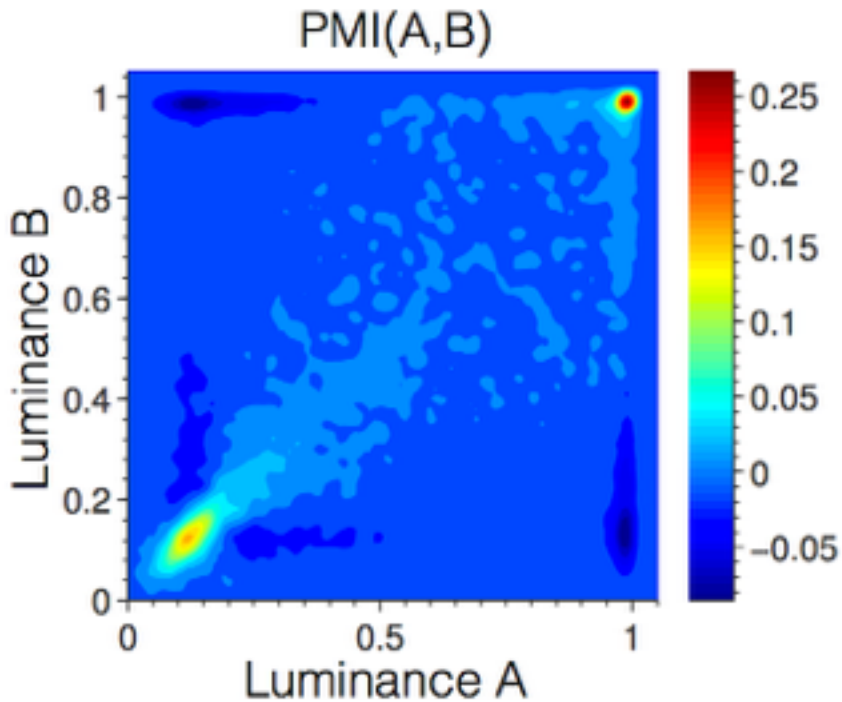
Step 1: Estimate feature co-occurrence distribution $P(A, B)$





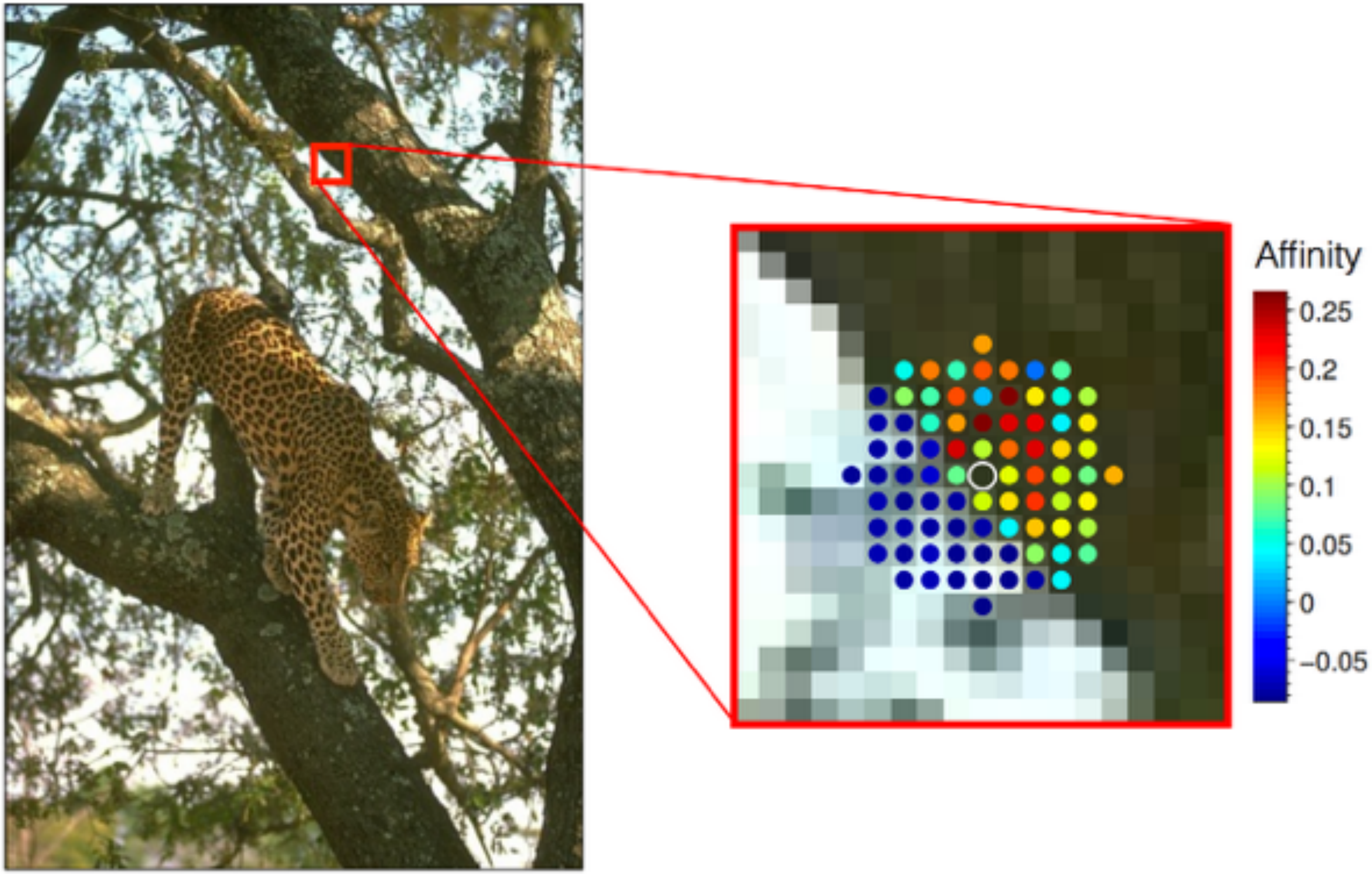
Step 2: Derive PMI(A,B) from feature co-occurrence distribution



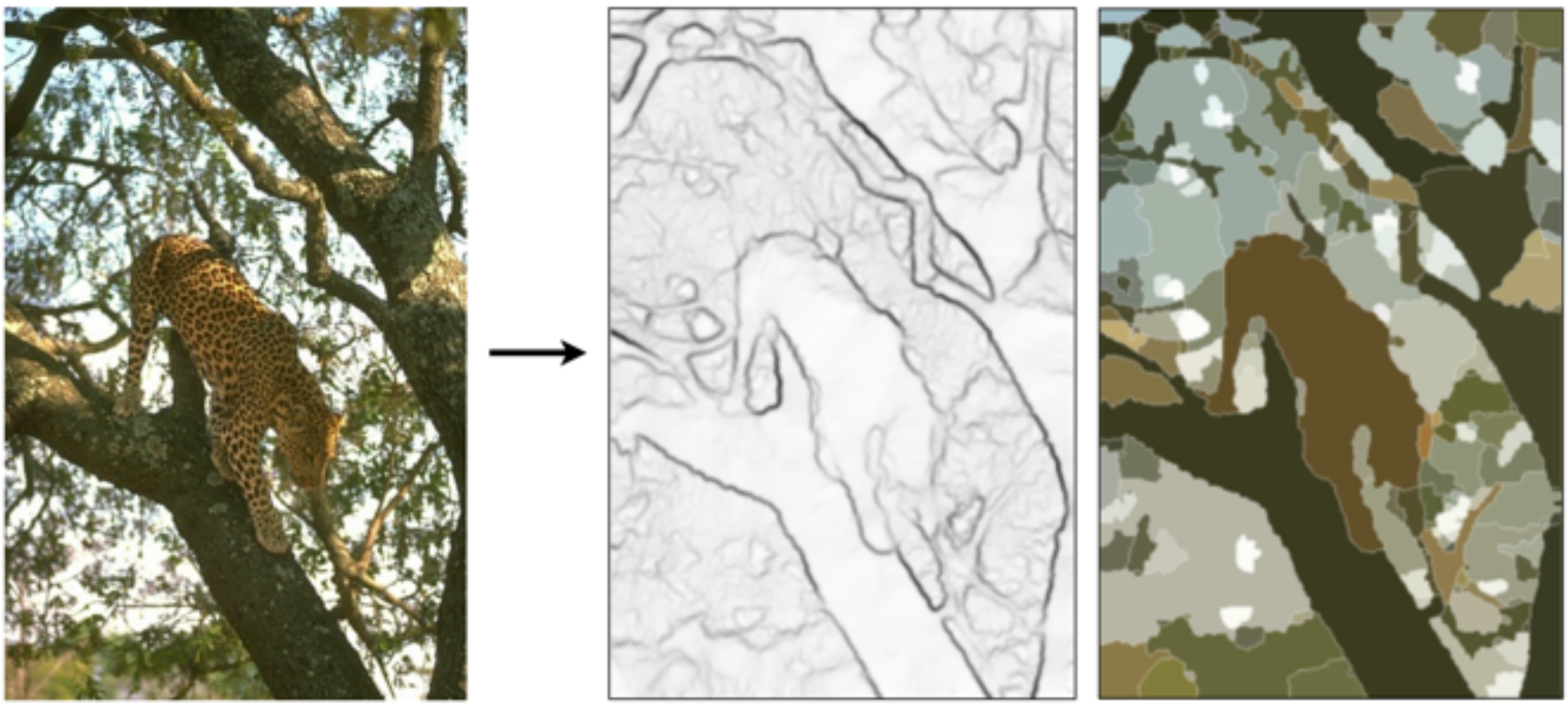


$$\mathrm{PMI}_\rho(A,B) = \log \frac{P(A,B)^\rho}{P(A)P(B)}$$

Step 3: Use PMI as affinity between each pair of nearby pixels



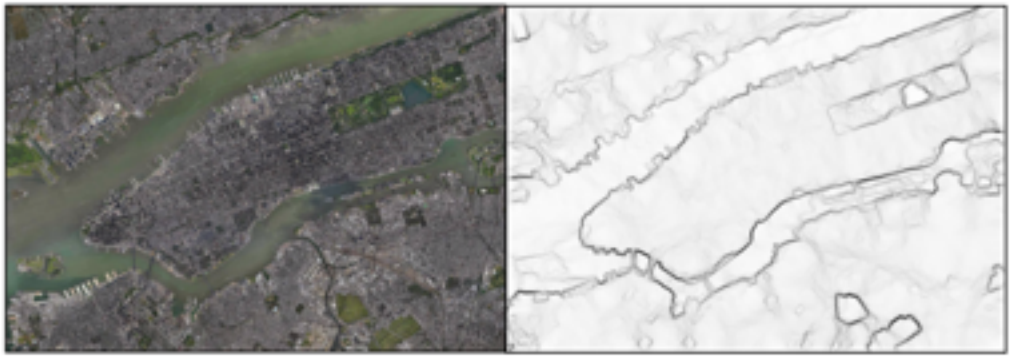
Step 4: Group pixels based on affinity (spectral clustering)



Works on diverse stimuli

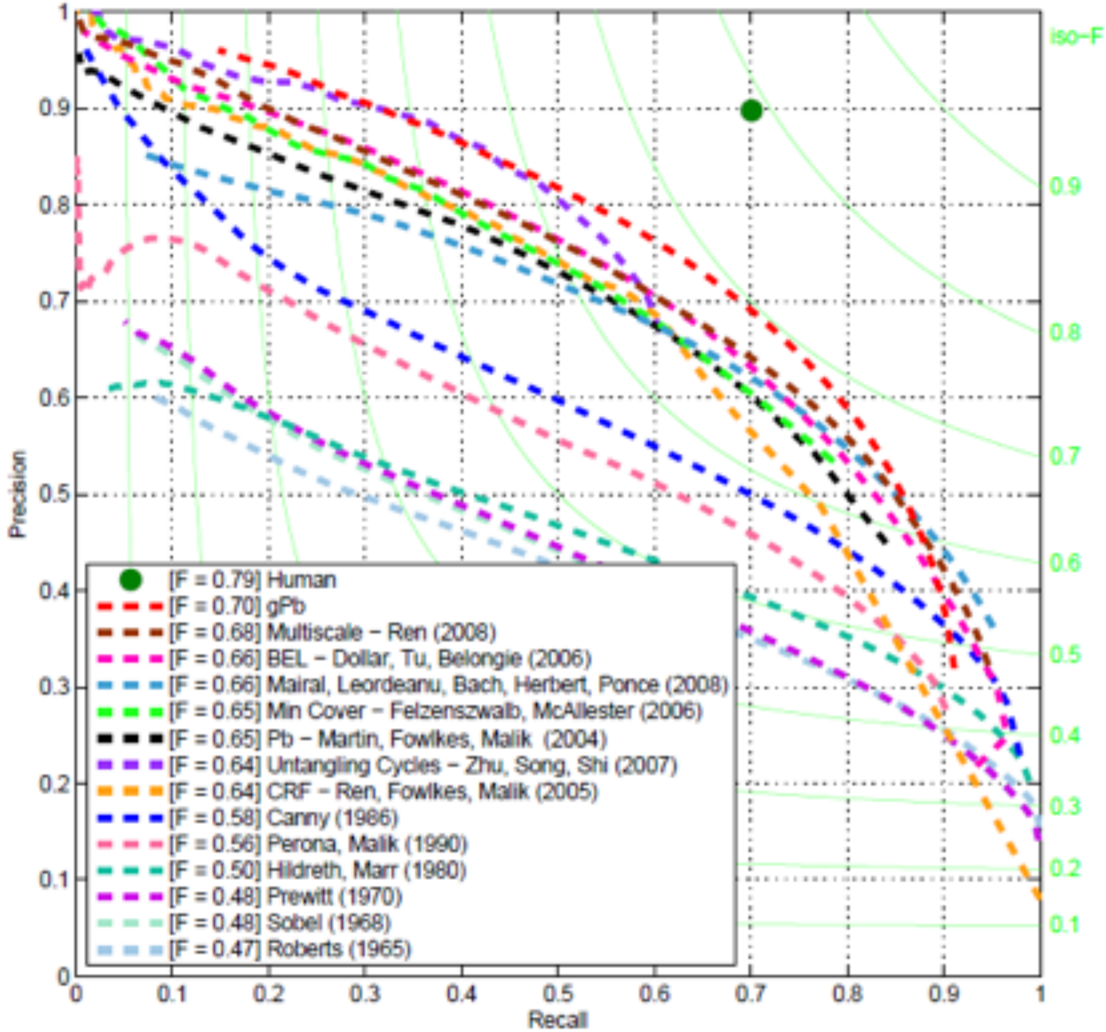
Cellphone photo

Satellite imagery



Art

45 years of boundary detection



Source: Arbelaez, Maire, Fowlkes, and Malik. TPAMI 2011 (pdf)

State of edge detection

- Local edge detection works well
  - But many false positives from illumination and texture edges
- Some methods to take into account longer contours, but could probably do better
- Few methods that actually “learn” from data. For example, Sketch Tokens, will do so.
- Poor use of object and high-level information

Questions

Take-home reading and demo code

- Szeliski Chapter 4.2 Edges
- Original PB paper:
- http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/papers/mfm-pami-boundary.pdf
- Crispy Boundary Paper and code:
- http://web.mit.edu/phillipi/pmi-boundaries/
- Edge detection with Skimage:
- http://scikit-image.org/docs/dev/user_guide/tutorial_segmentation.html