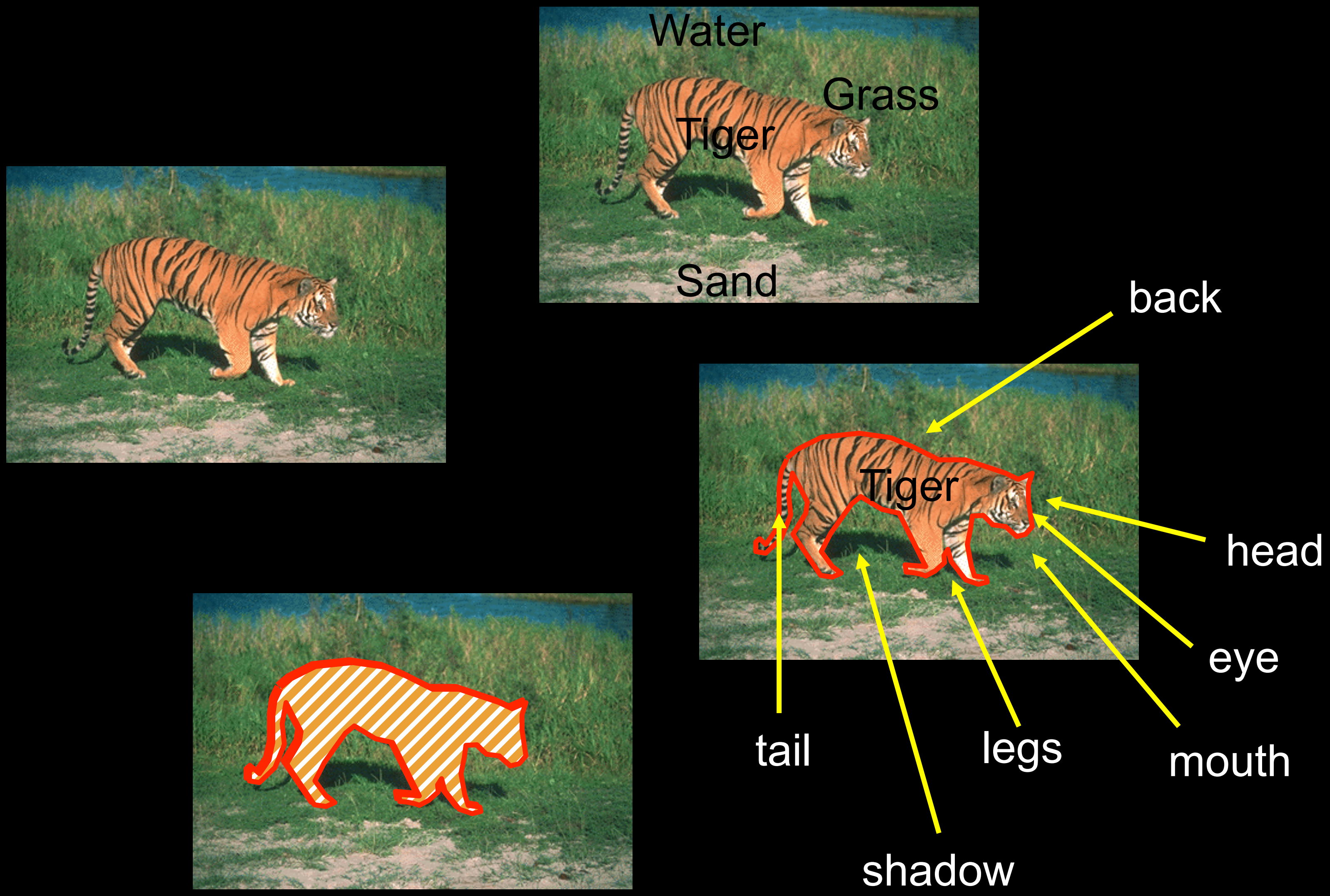
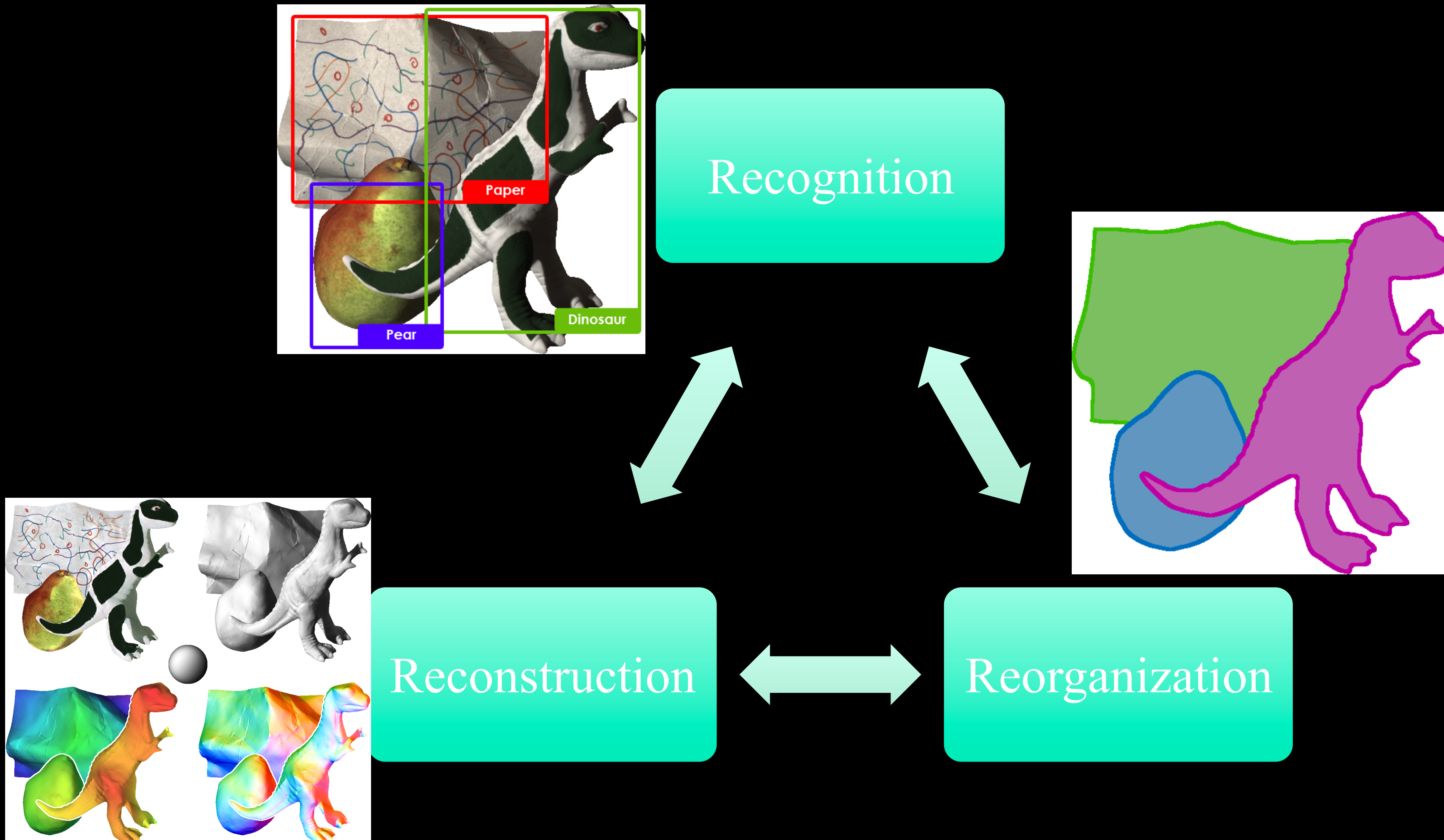
Visual Grouping:
Contours and Regions in Natural Images

Jitendra Malik
University of California at Berkeley

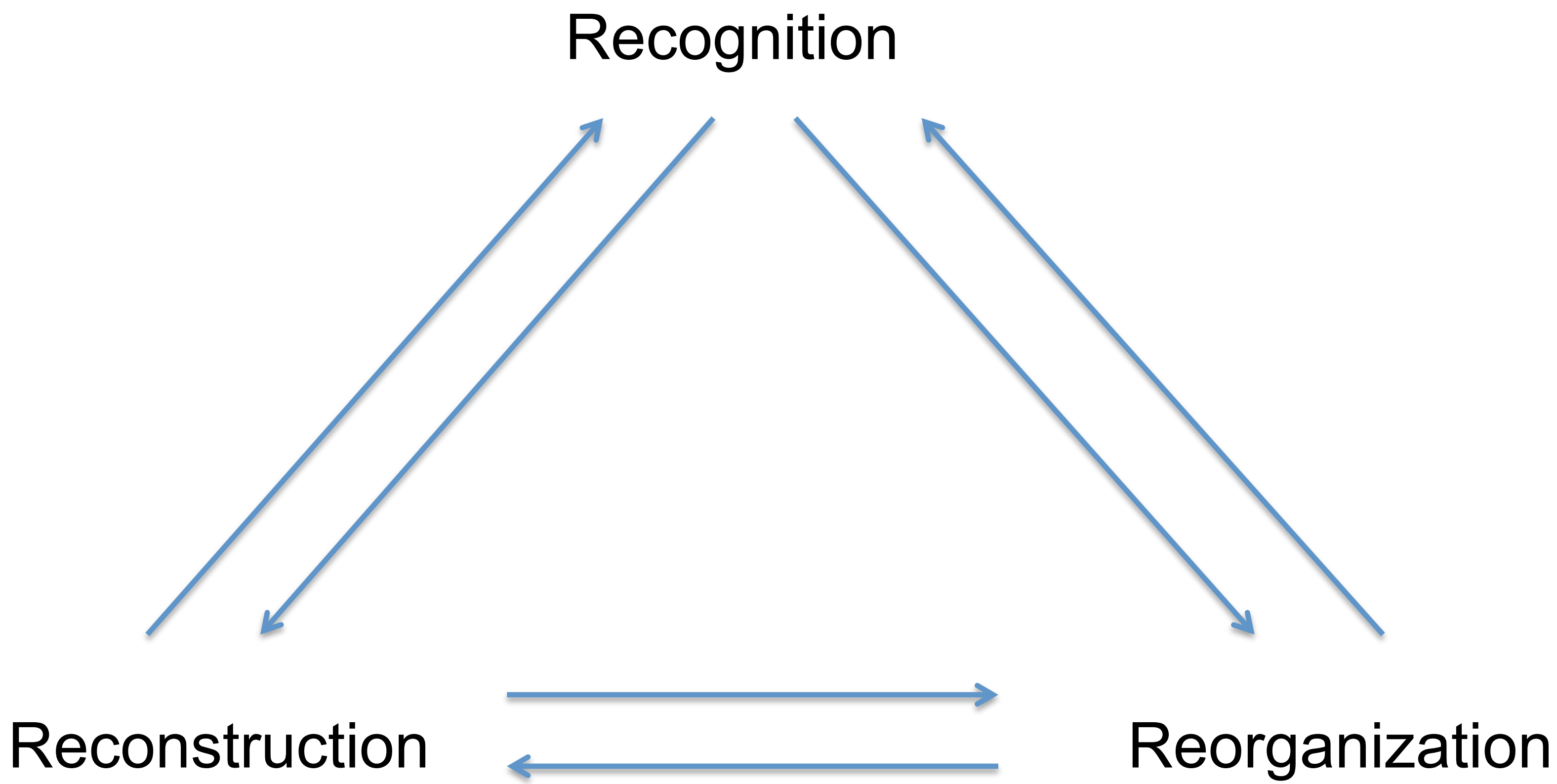
From Pixels to Perception



Recognition, Reconstruction & Reorganization



The Three R's of Vision

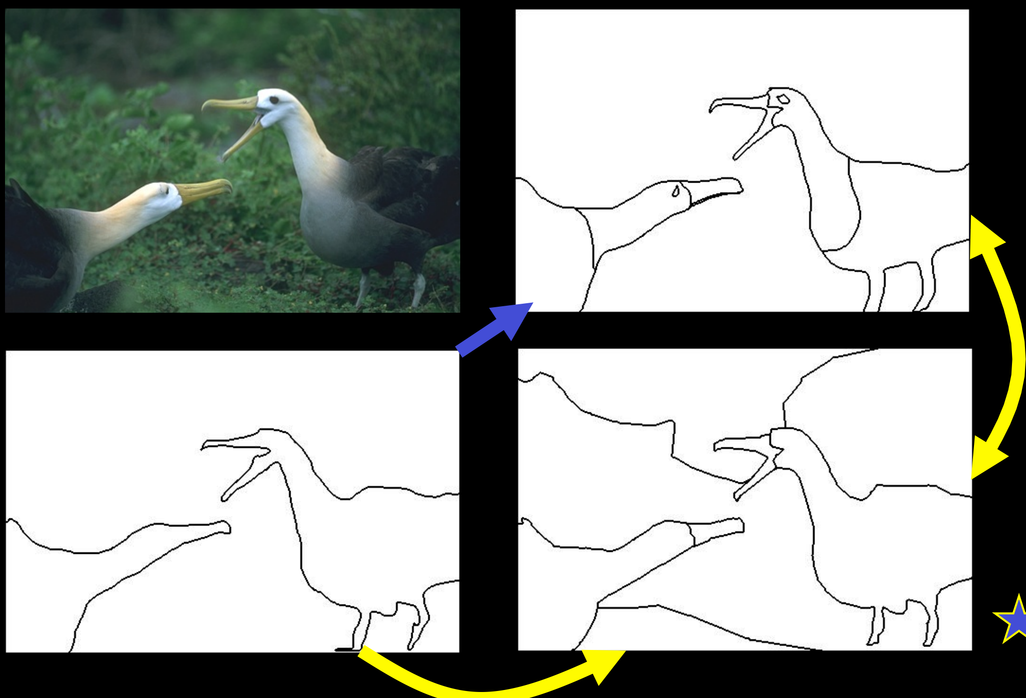


Each of the 6 directed arcs in this diagram is a useful direction of information flow

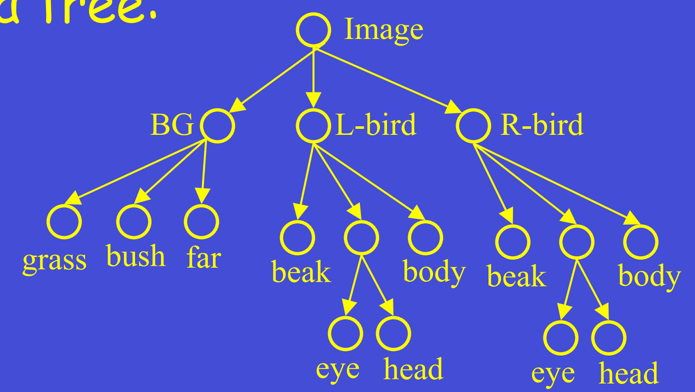
Berkeley Segmentation DataSet [BSDS]

D. Martin, C. Fowlkes, D. Tal, J. Malik. "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics", ICCV, 2001

Consistency

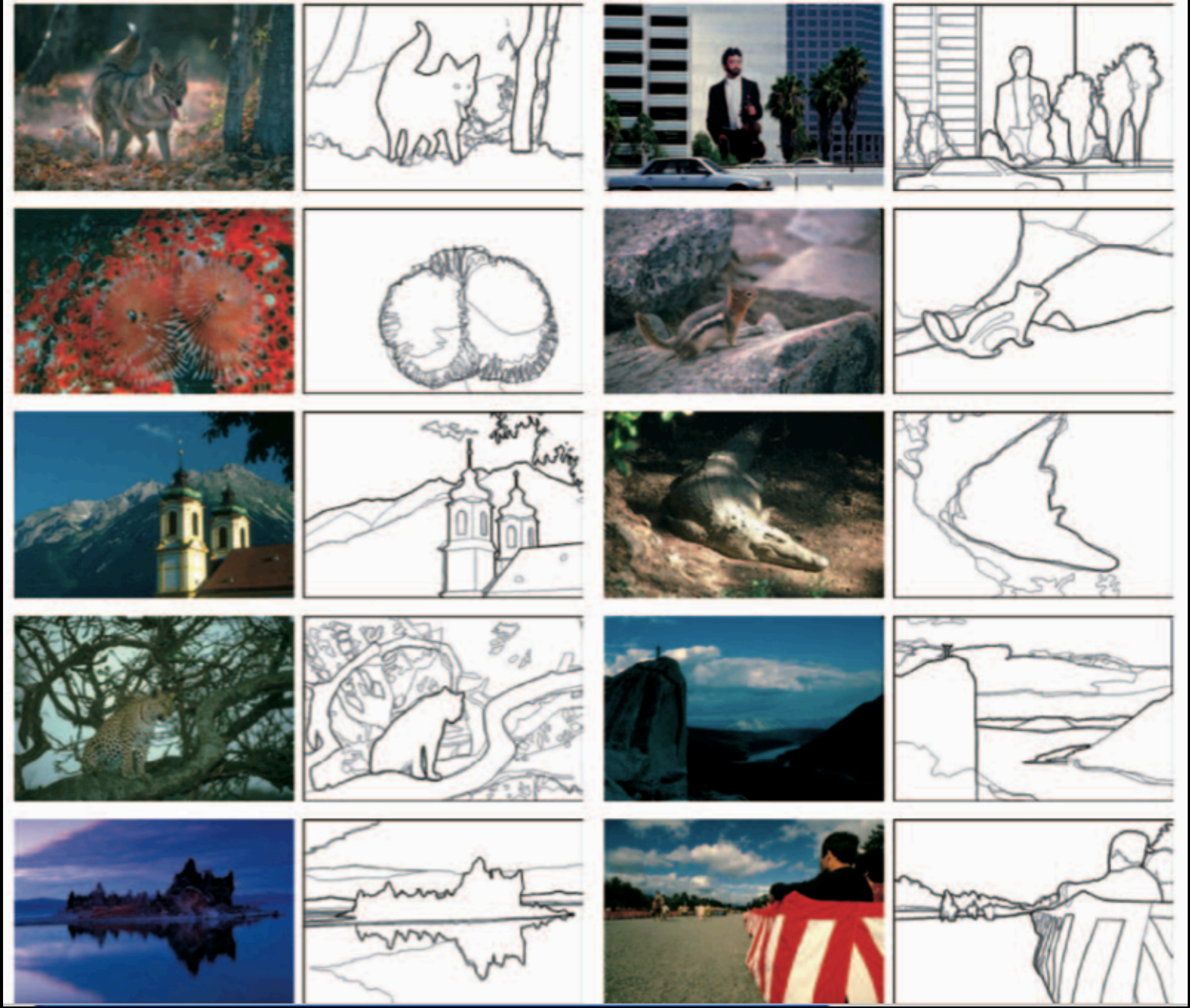


Perceptual organization forms a tree:

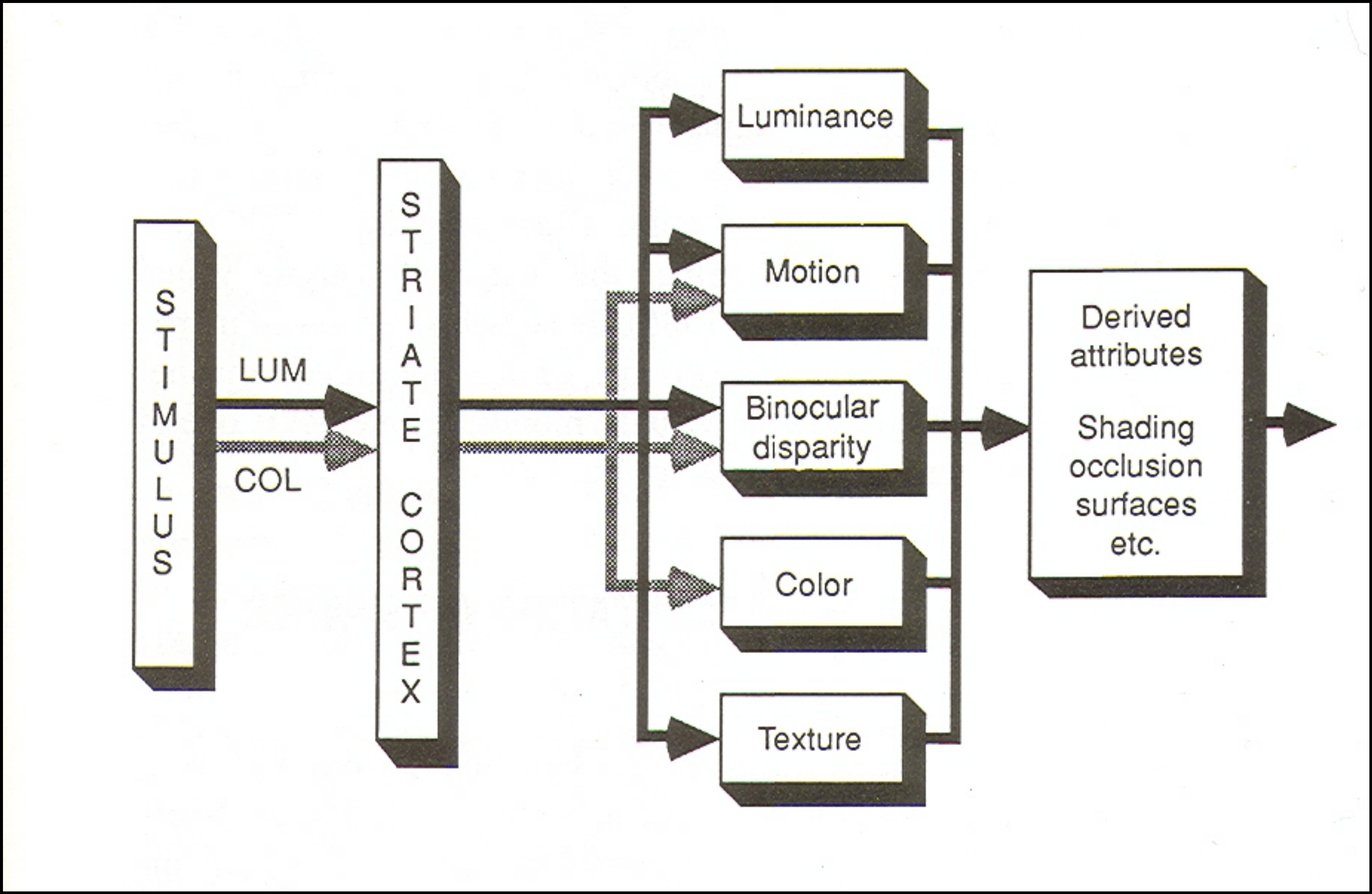


- A,C are refinements of B
- A,C are mutual refinements
- A,B,C represent the same percept
  - Attention accounts for differences

★ Two segmentations are consistent when they can be explained by the same segmentation tree (i.e. they could be derived from a single perceptual organization).



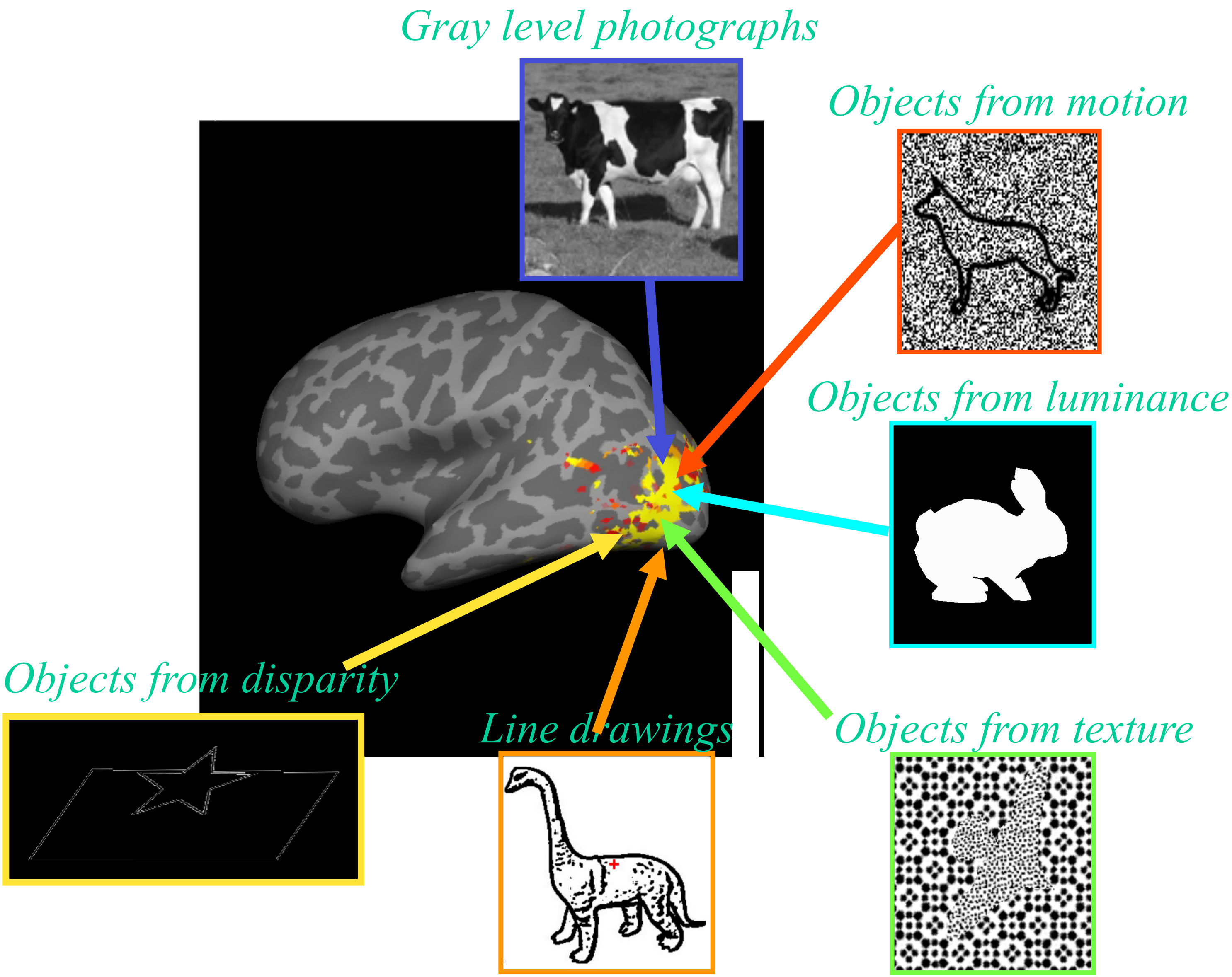
Contours can be defined by any of a number of cues (P. Cavanagh)



Boundaries of image regions defined by a number of cues

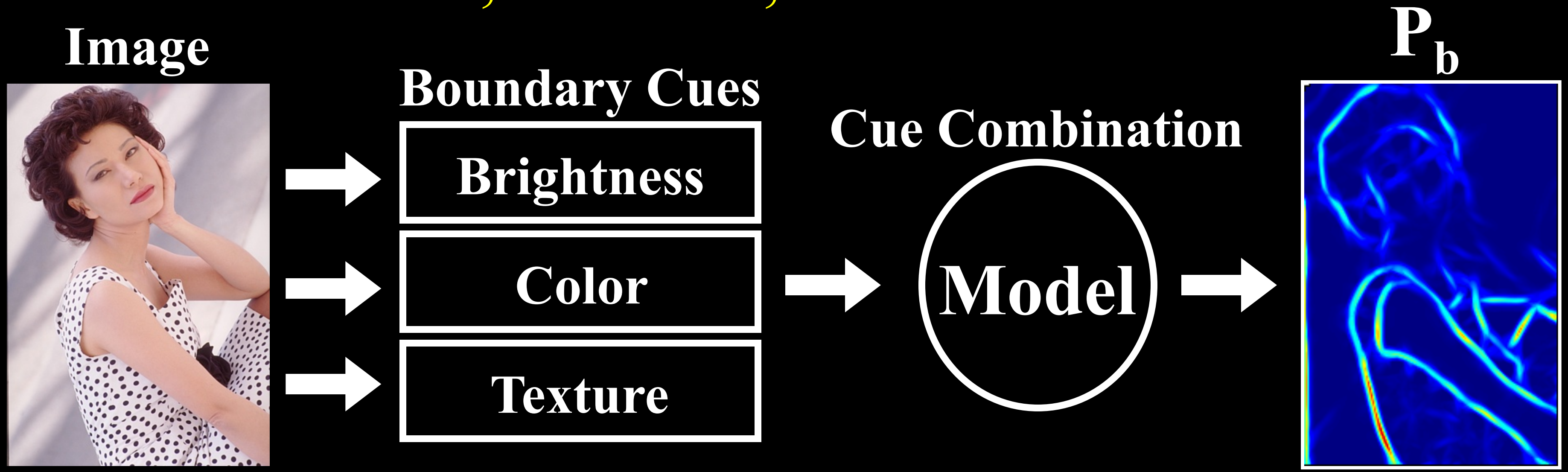
- Brightness
- Color
- Texture
- Motion (in video)
- Binocular Diparity (if available)
- Familiar objects

Cue-Invariant Representations



Grill-Spector et al. , Neuron 1998

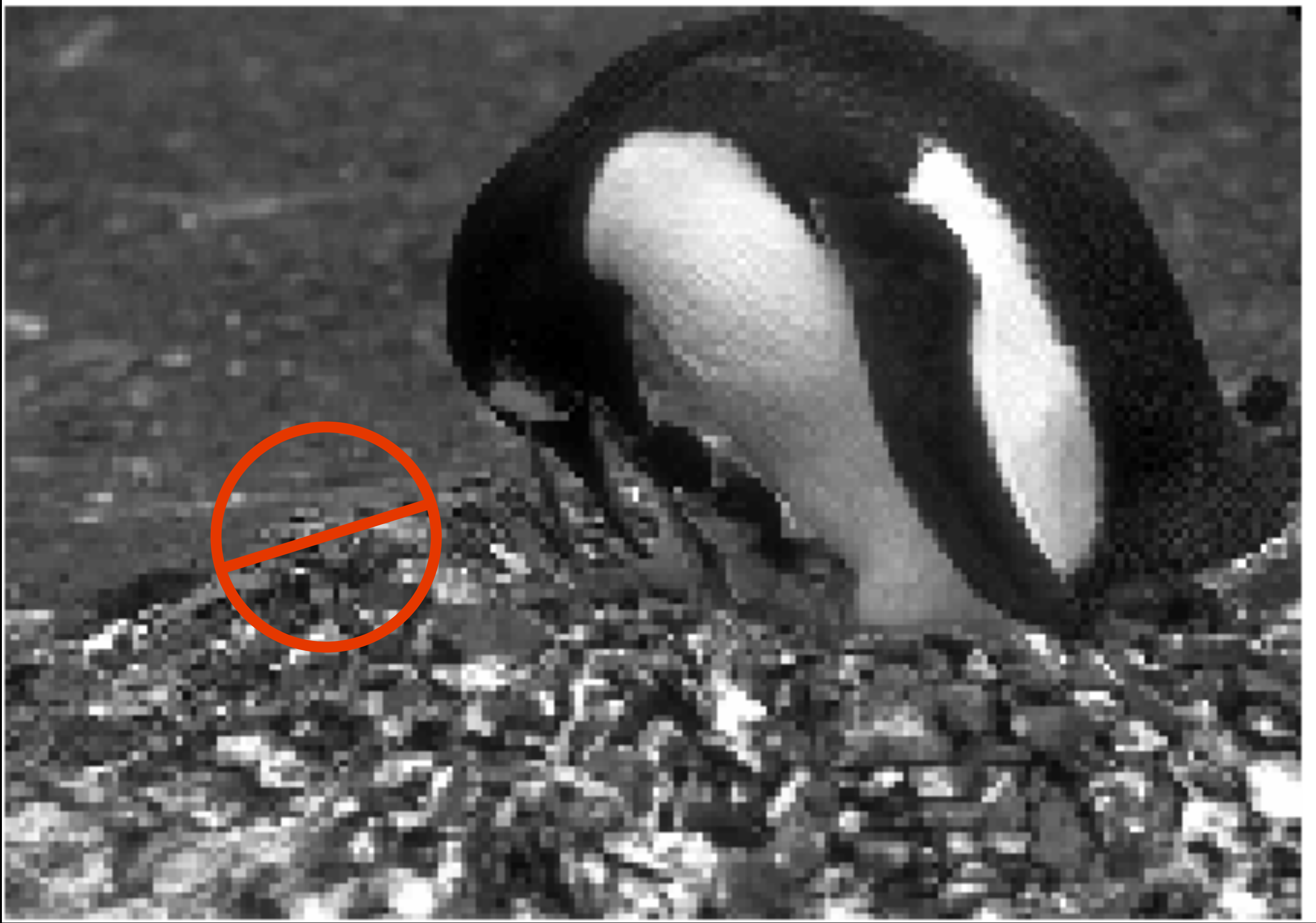
Martin, Fowlkes, Malik PAMI 04



Challenges: texture cue, cue combination

Goal: learn the posterior probability of a boundary $P_b(x,y,\theta)$ from local information only

Oriented Feature Gradient

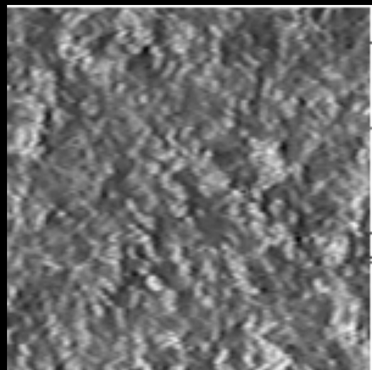


Visual Texture

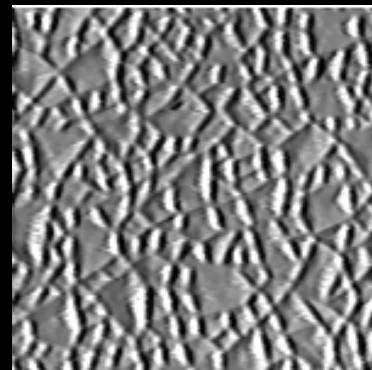
Jitendra Malik

University of California, Berkeley

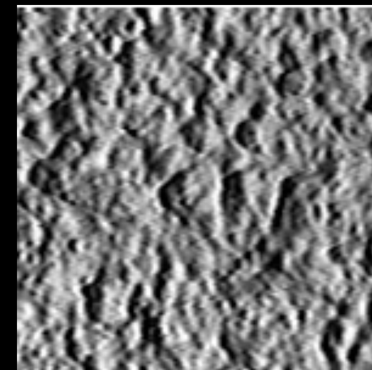
Example Natural Materials



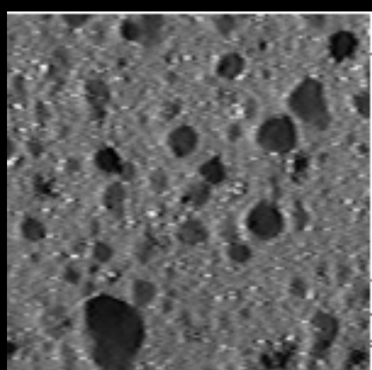
Terrycloth



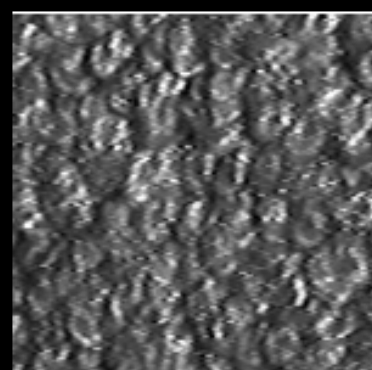
Rough Plastic



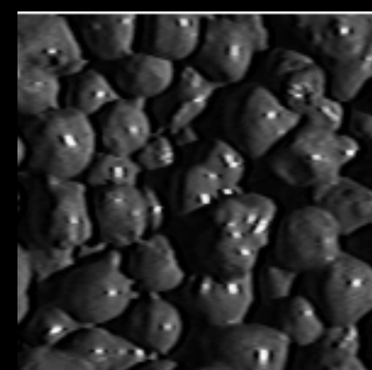
Plaster-b



Sponge



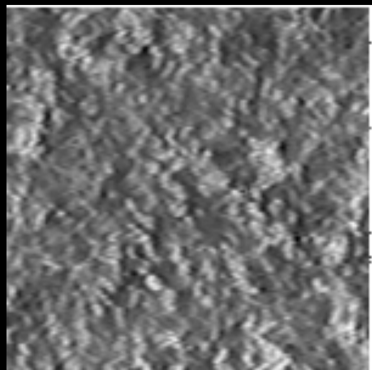
Rug-a

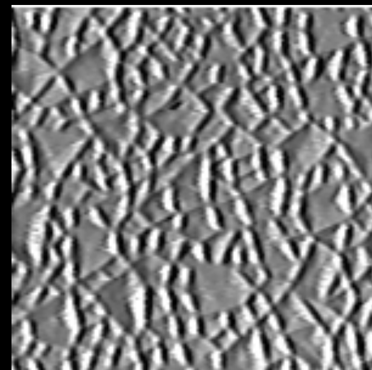


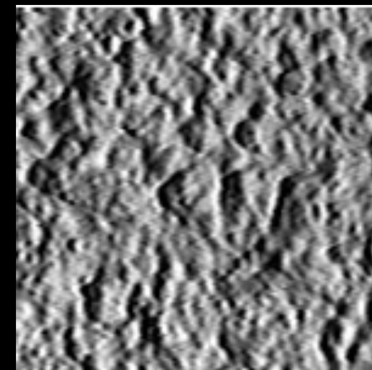
Painted Spheres

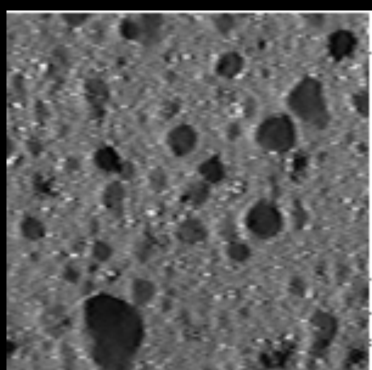
Columbia-Utrecht Database (http://www.cs.columbia.edu/CAVE)

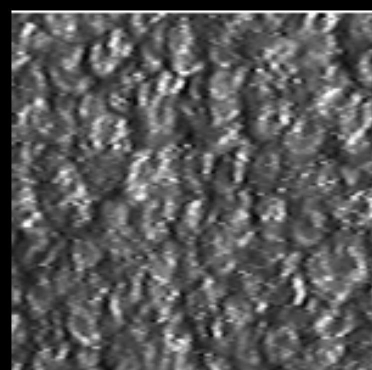
Example Natural Materials

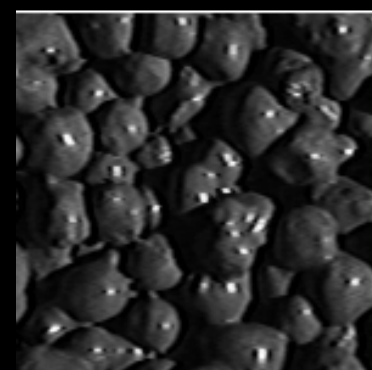

Terrycloth


Rough Plastic


Plaster-b


Sponge

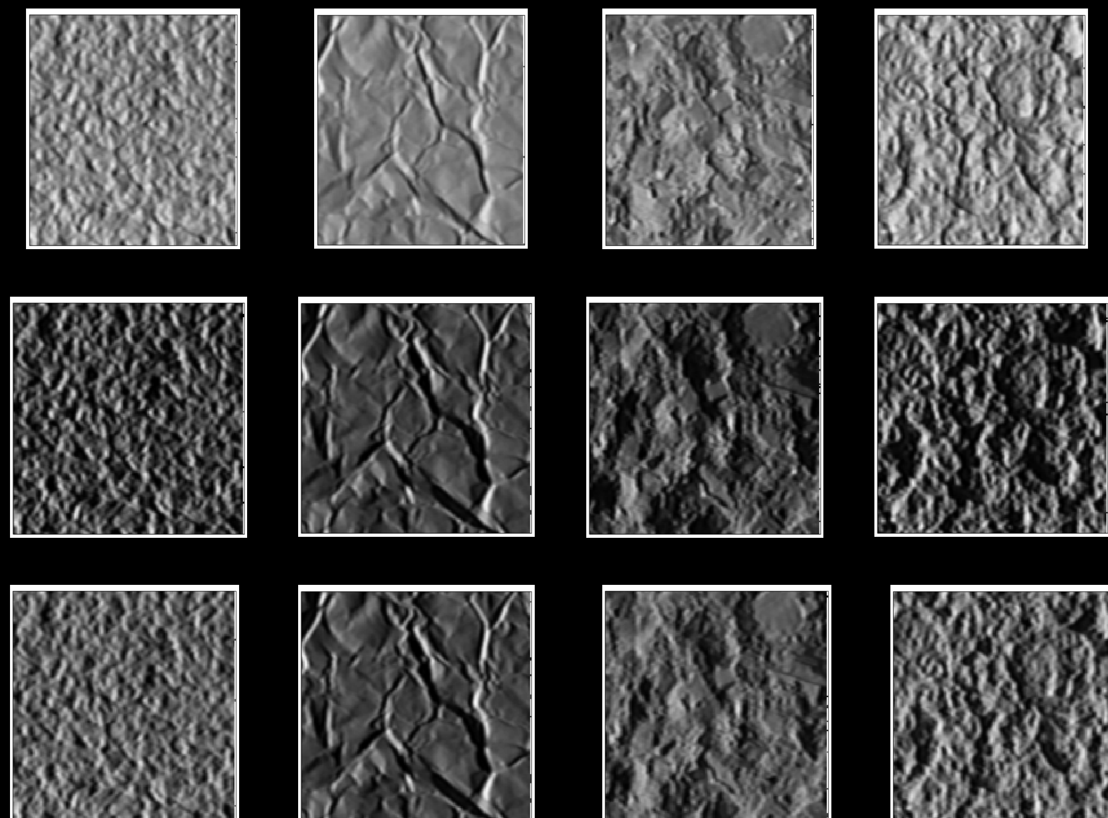

Rug-a


Painted Spheres

Columbia-Utrecht Database (http://www.cs.columbia.edu/CAVE)

Materials under different illumination and viewing directions

Different illumination and viewing directions



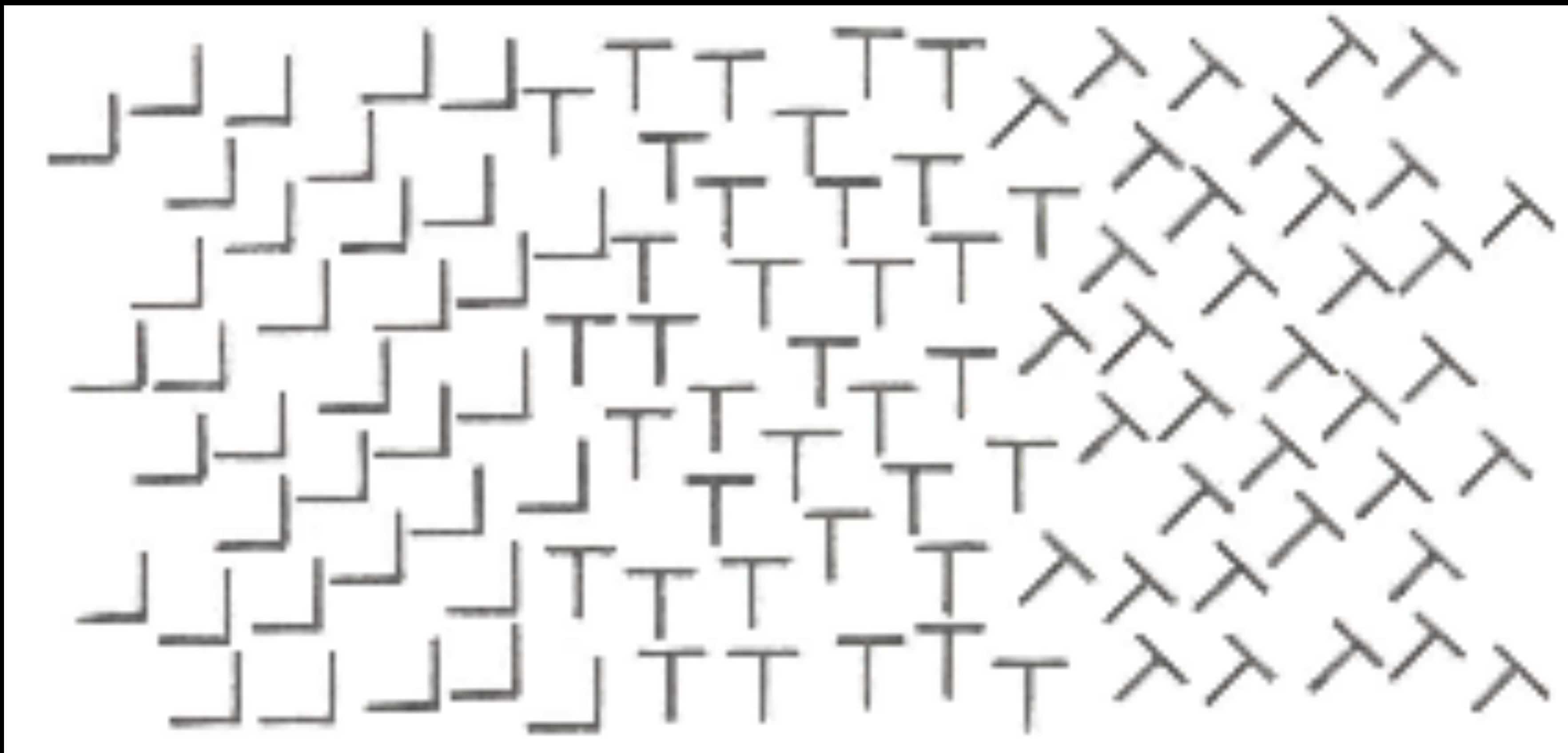
Plaster-a | Crumpled Paper | Concrete | Plaster-b (zoomed)

Julesz’ s texton theory

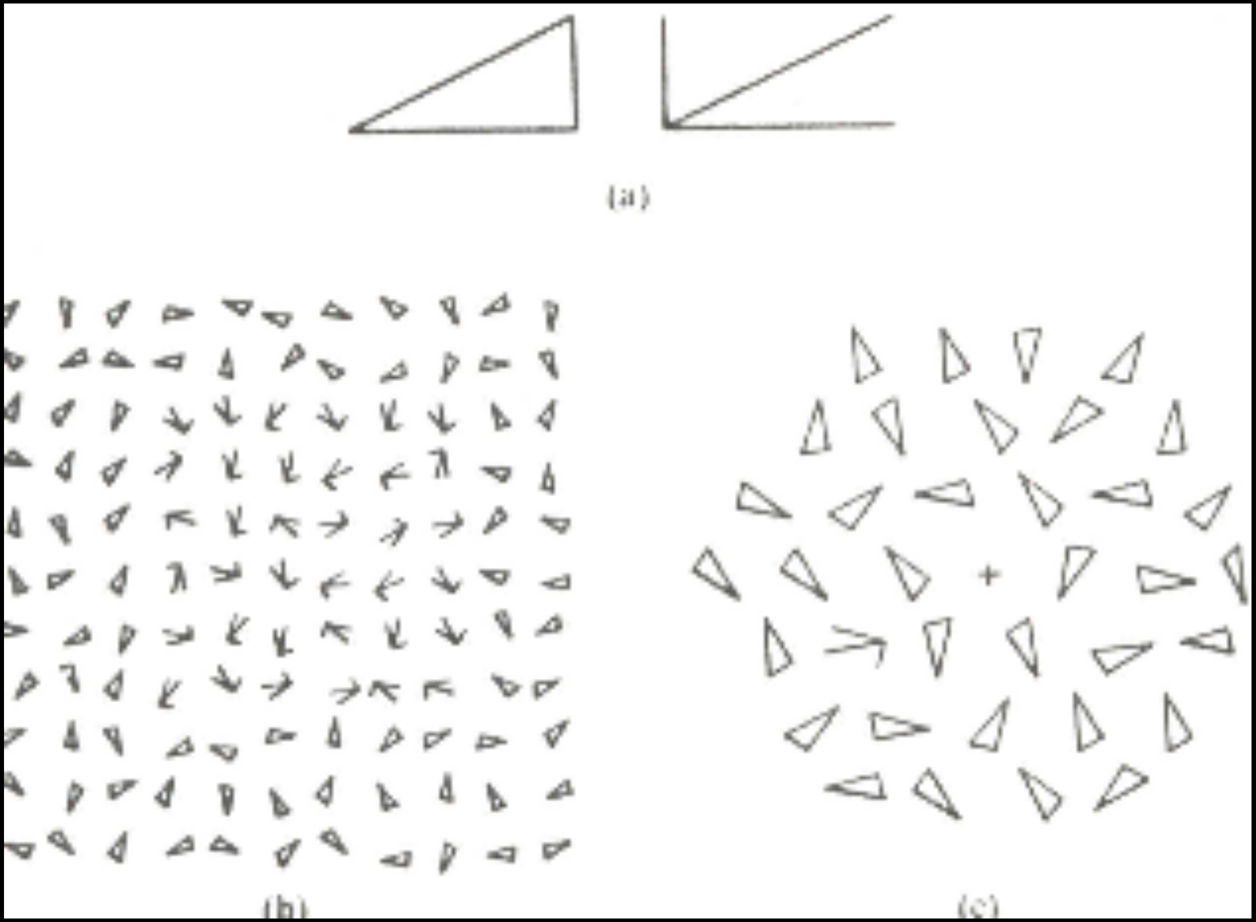
1. Human Vision operates in two distinct modes
   1. Pre-attentive vision - parallel, instantaneous, without scrutiny, independent of the number of patterns
   2. Attentive vision - serial search by focal attention in 50 ms steps limited to a small aperture as in form recognition

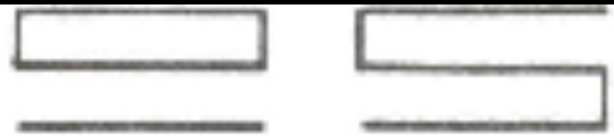
2. Textons are
   1. Elongated blobs - e.g. rectangles, ellipses, line segments with specific orientations, widths and lengths
   2. Terminators - ends of line segments
   3. Crossings of line segments

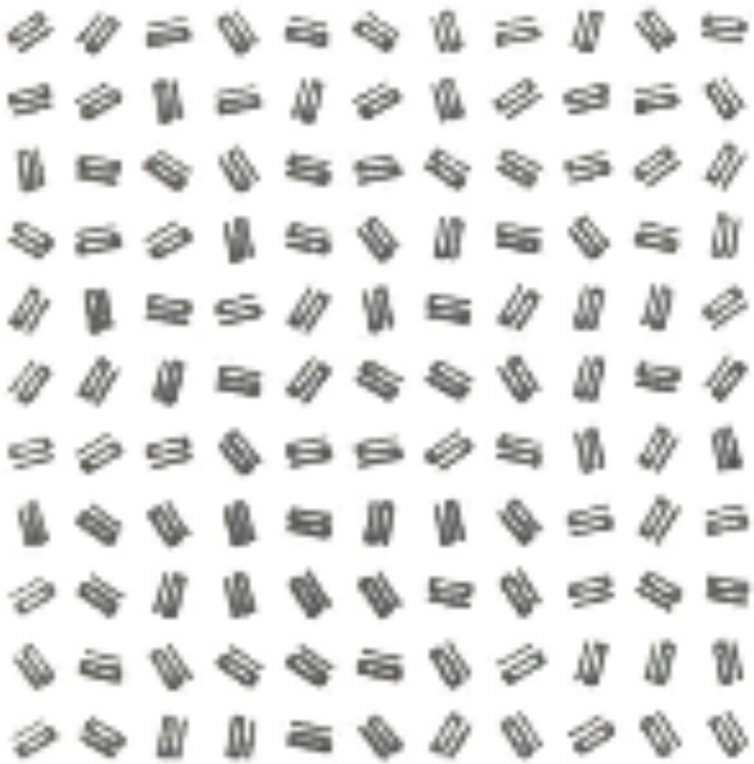
Orientation is a texton

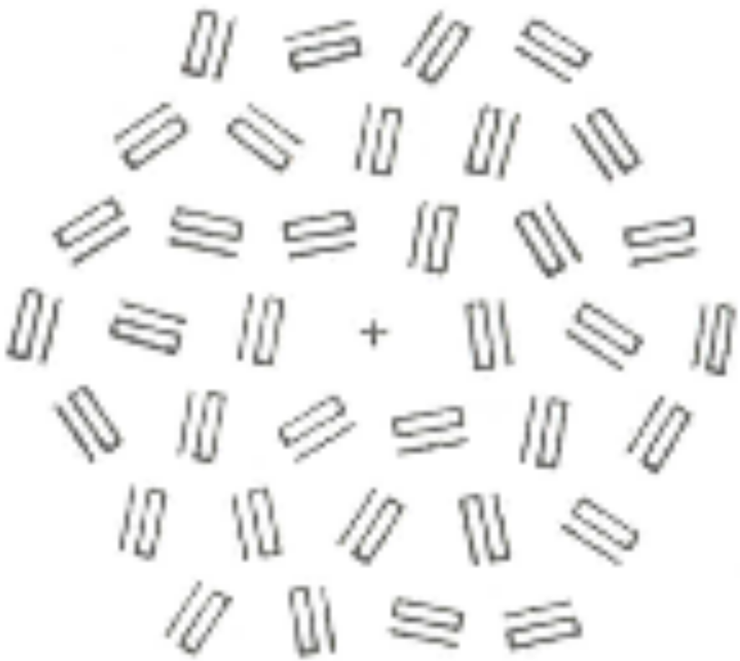


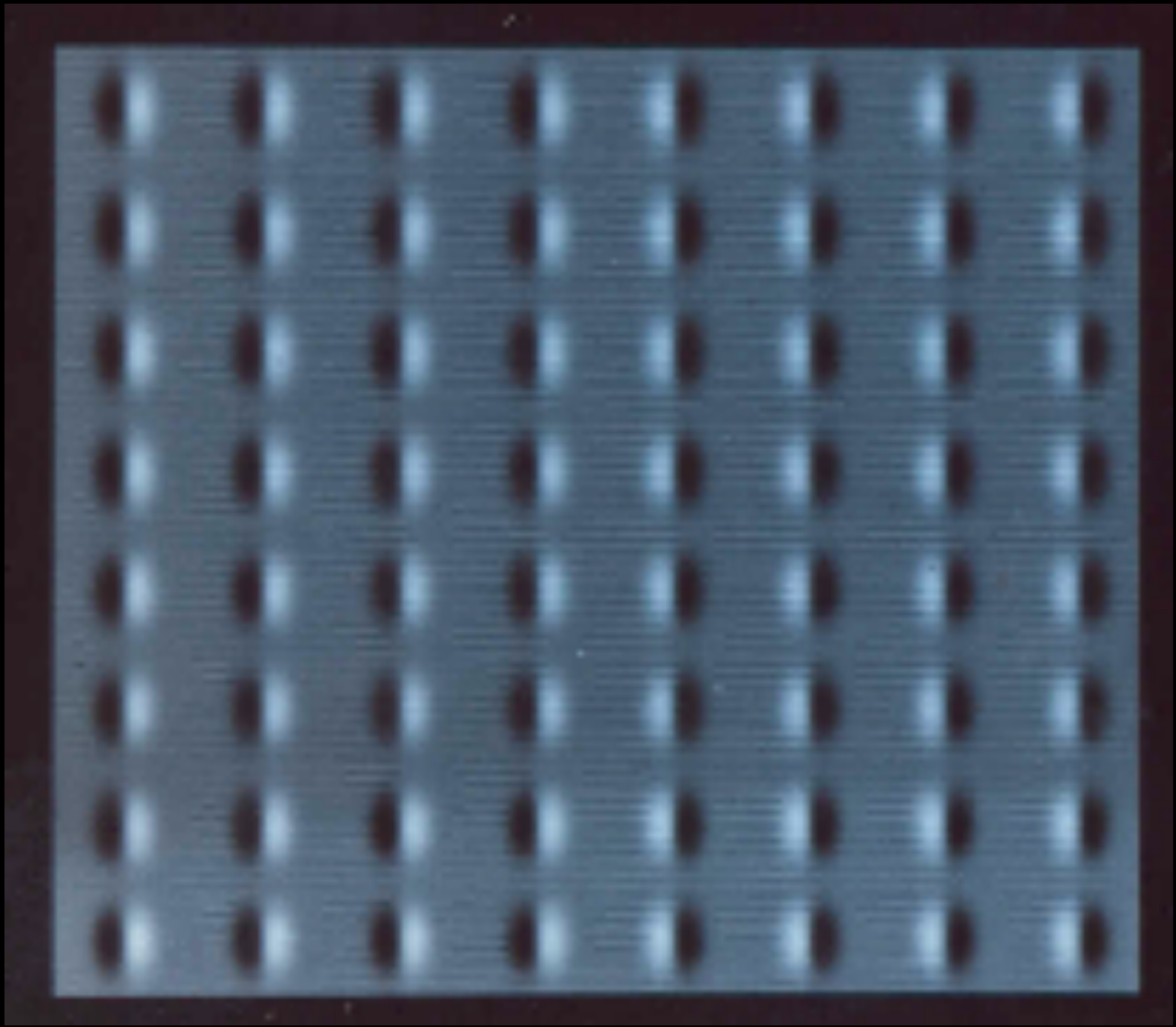
Terminators are textons

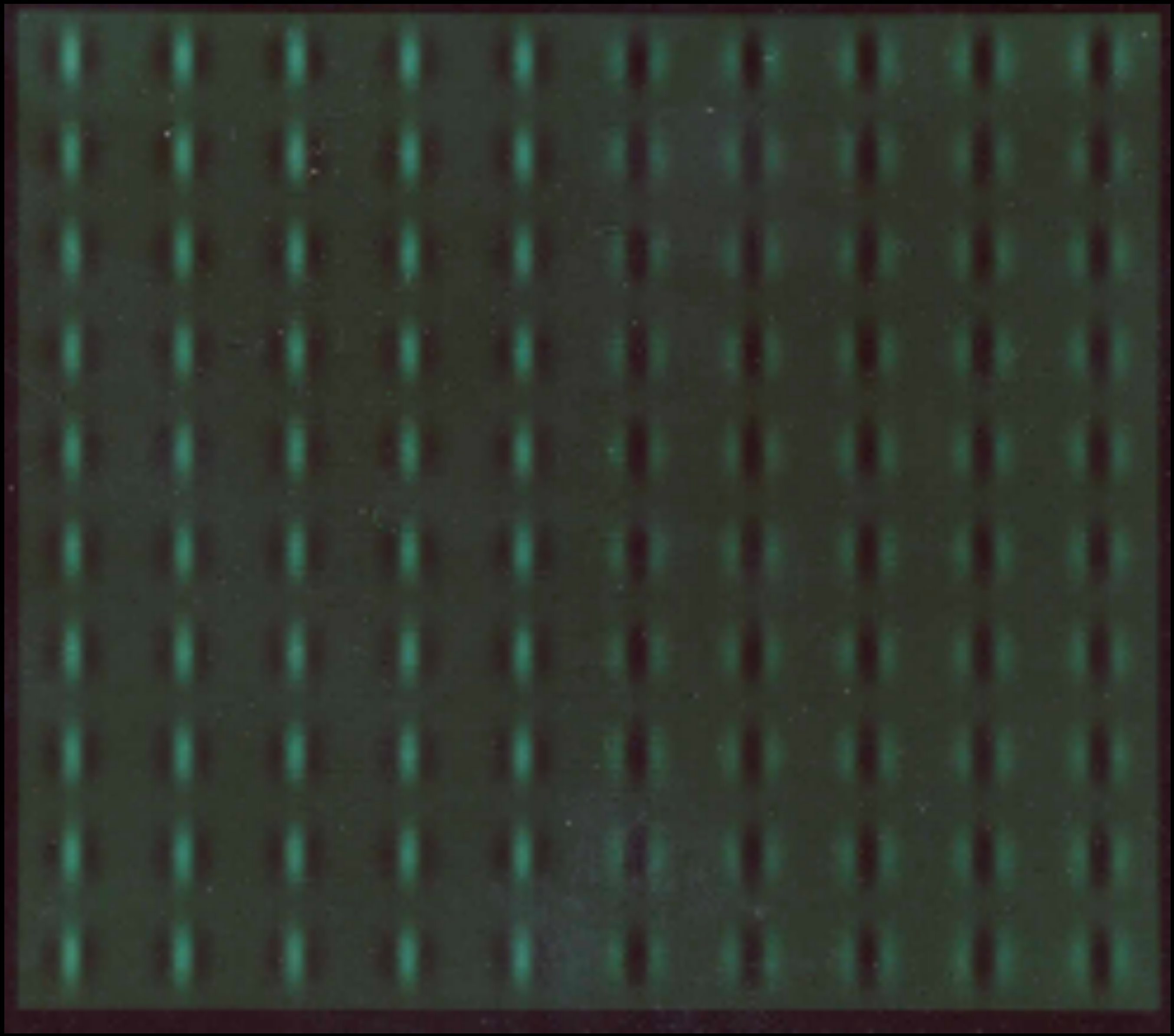


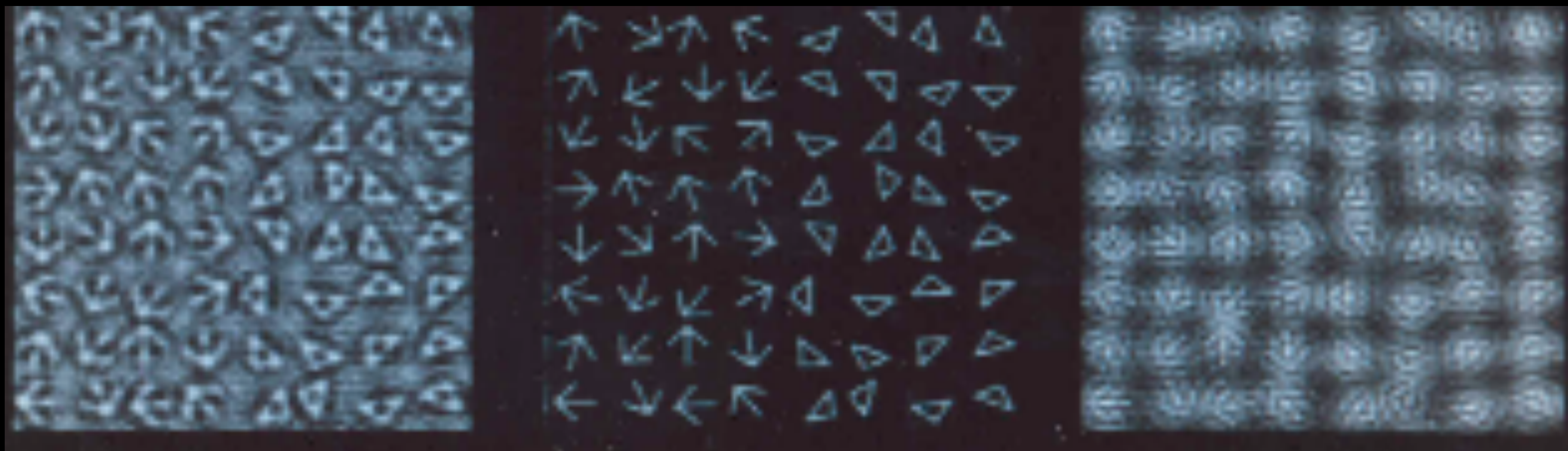

(a)

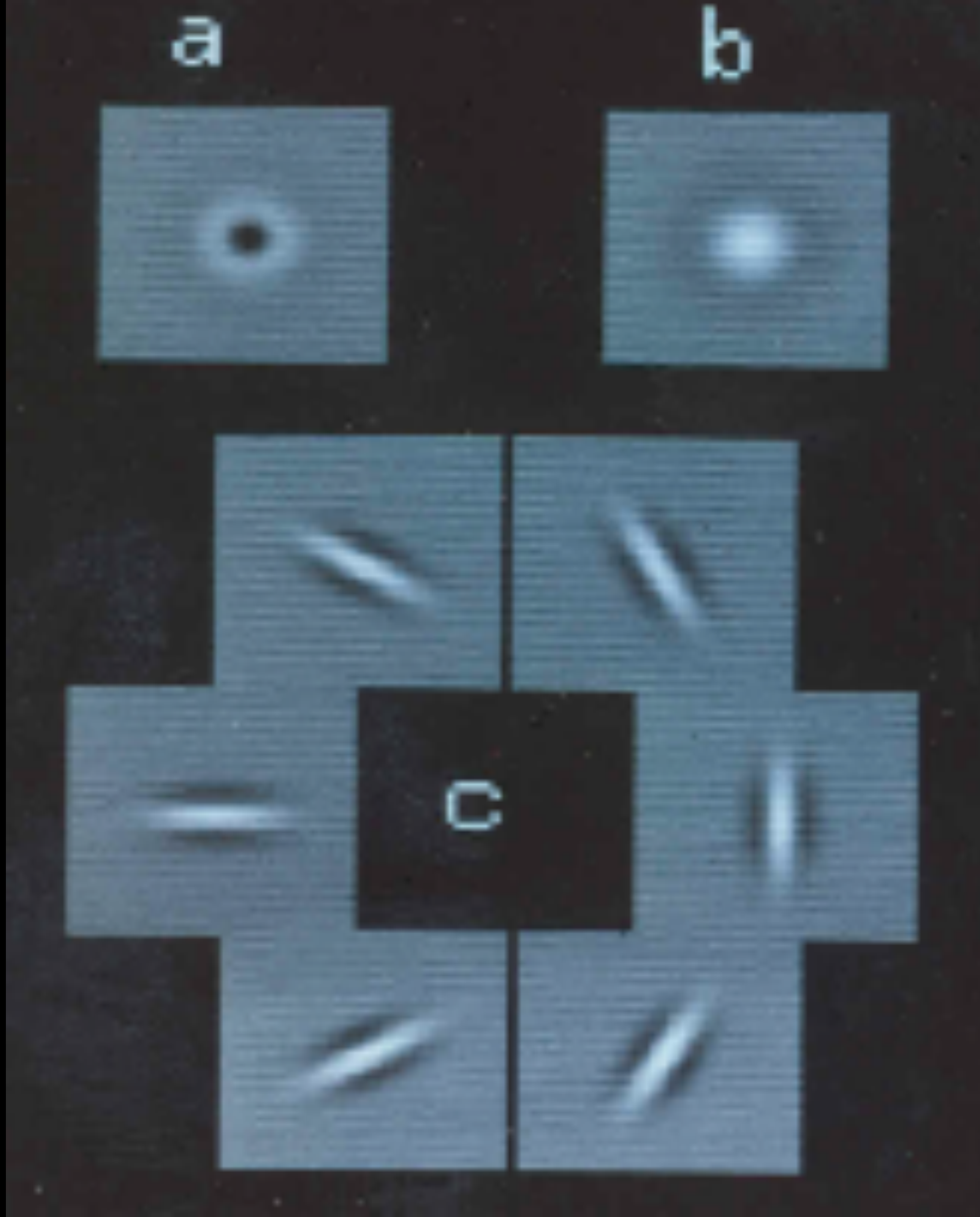

(b)

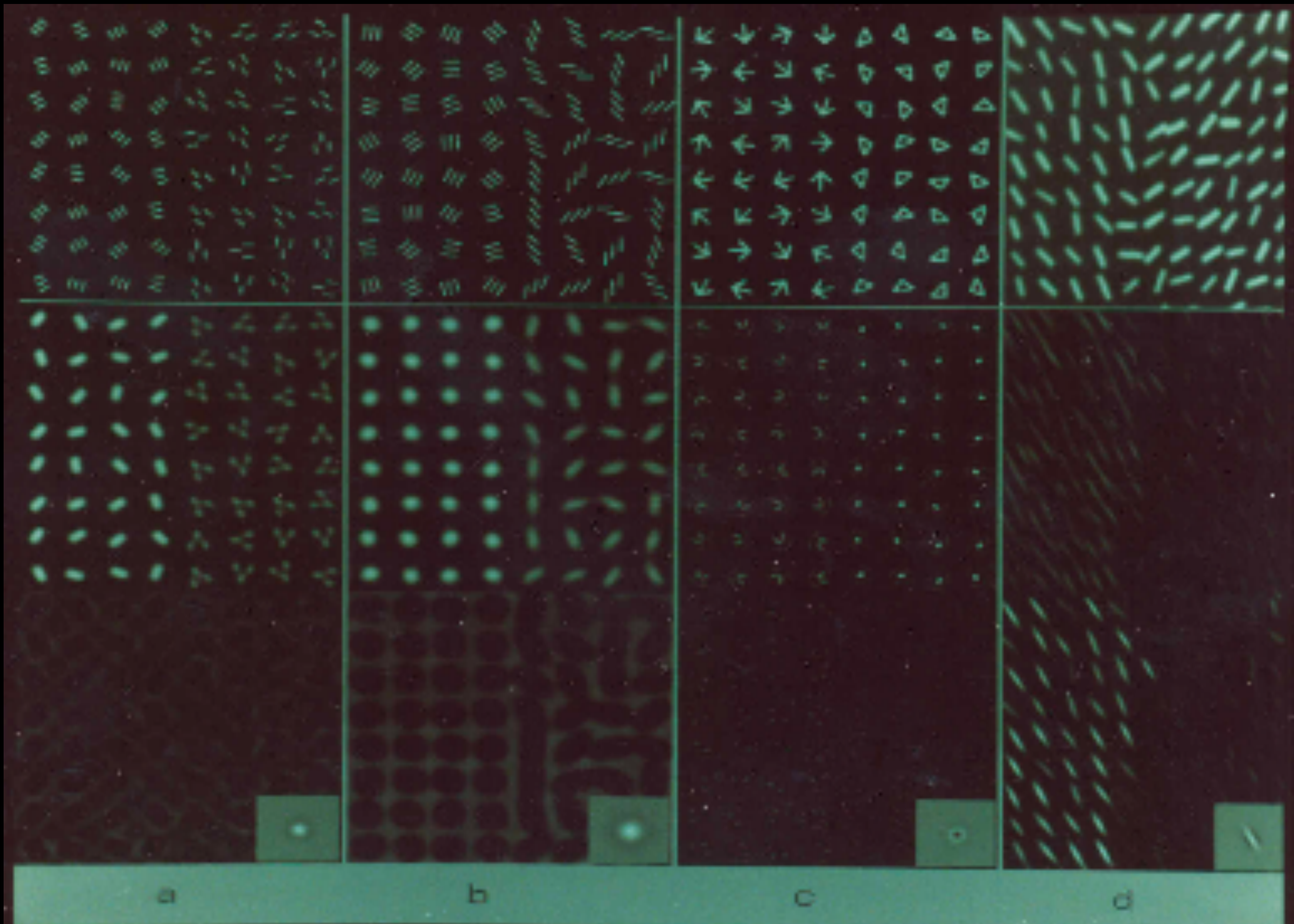

(c)



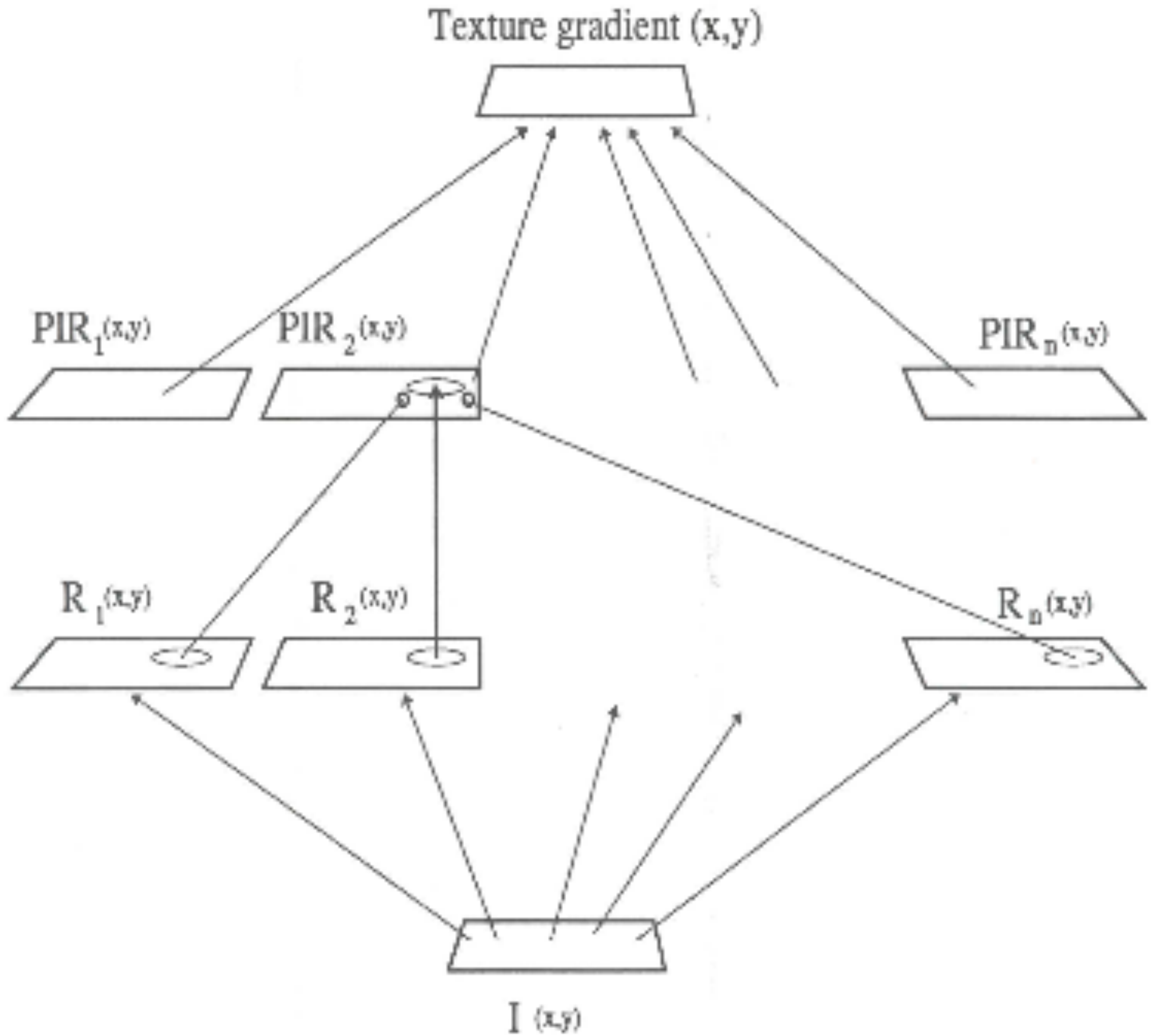


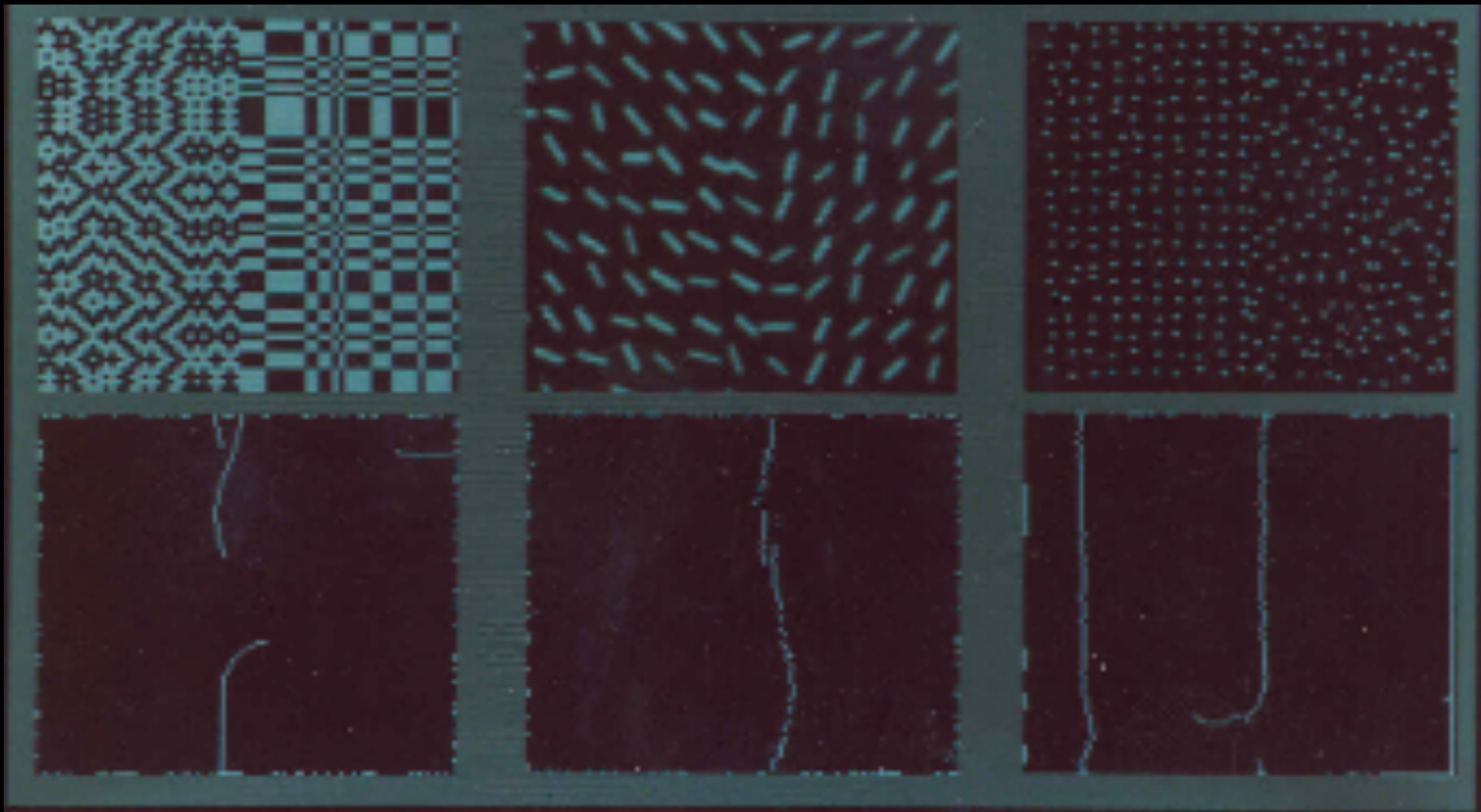



a
b
c


a
b
c
d

Texture gradient: $\max_i(|\nabla PIR_i * G_\sigma|)$





Six sample textures

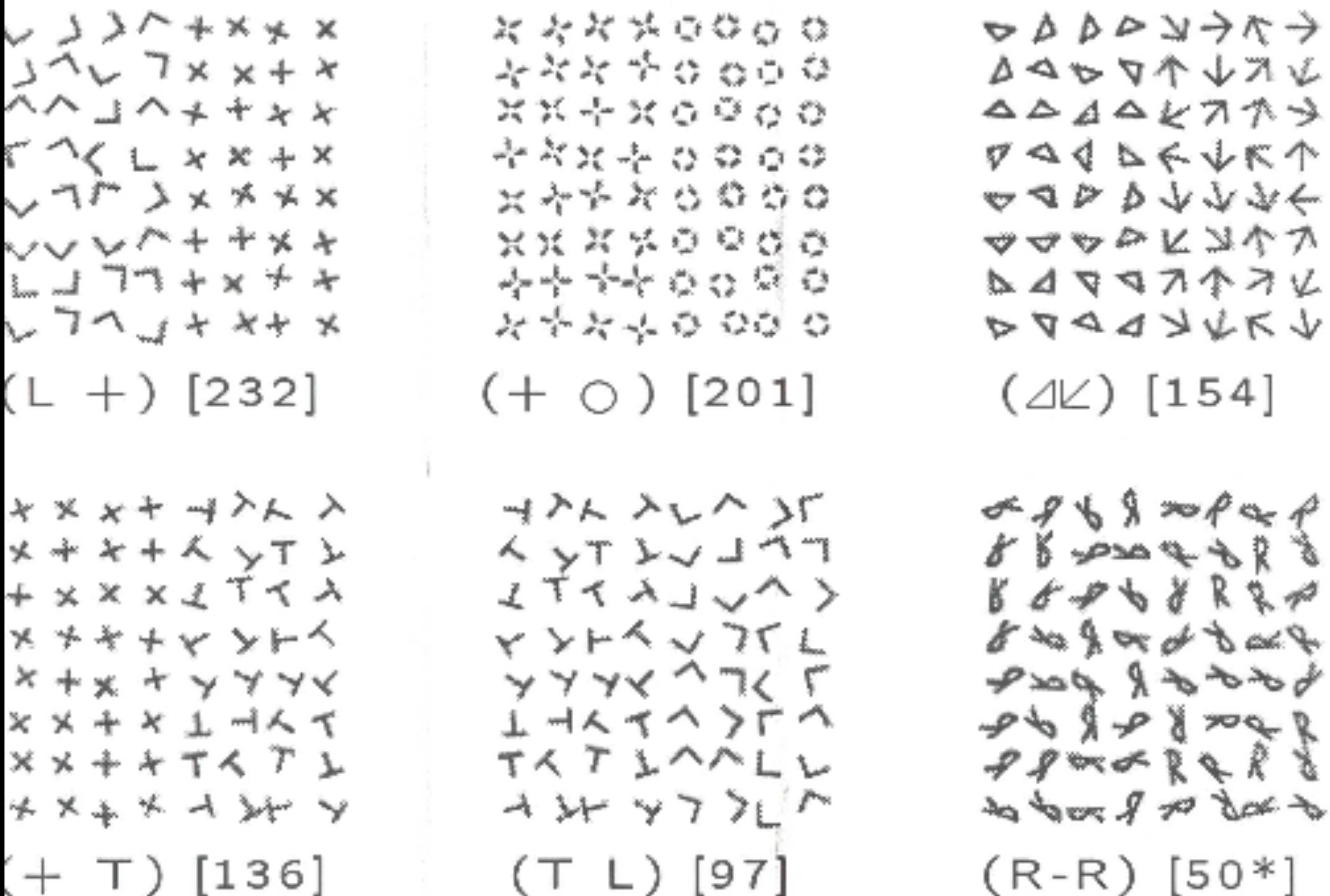
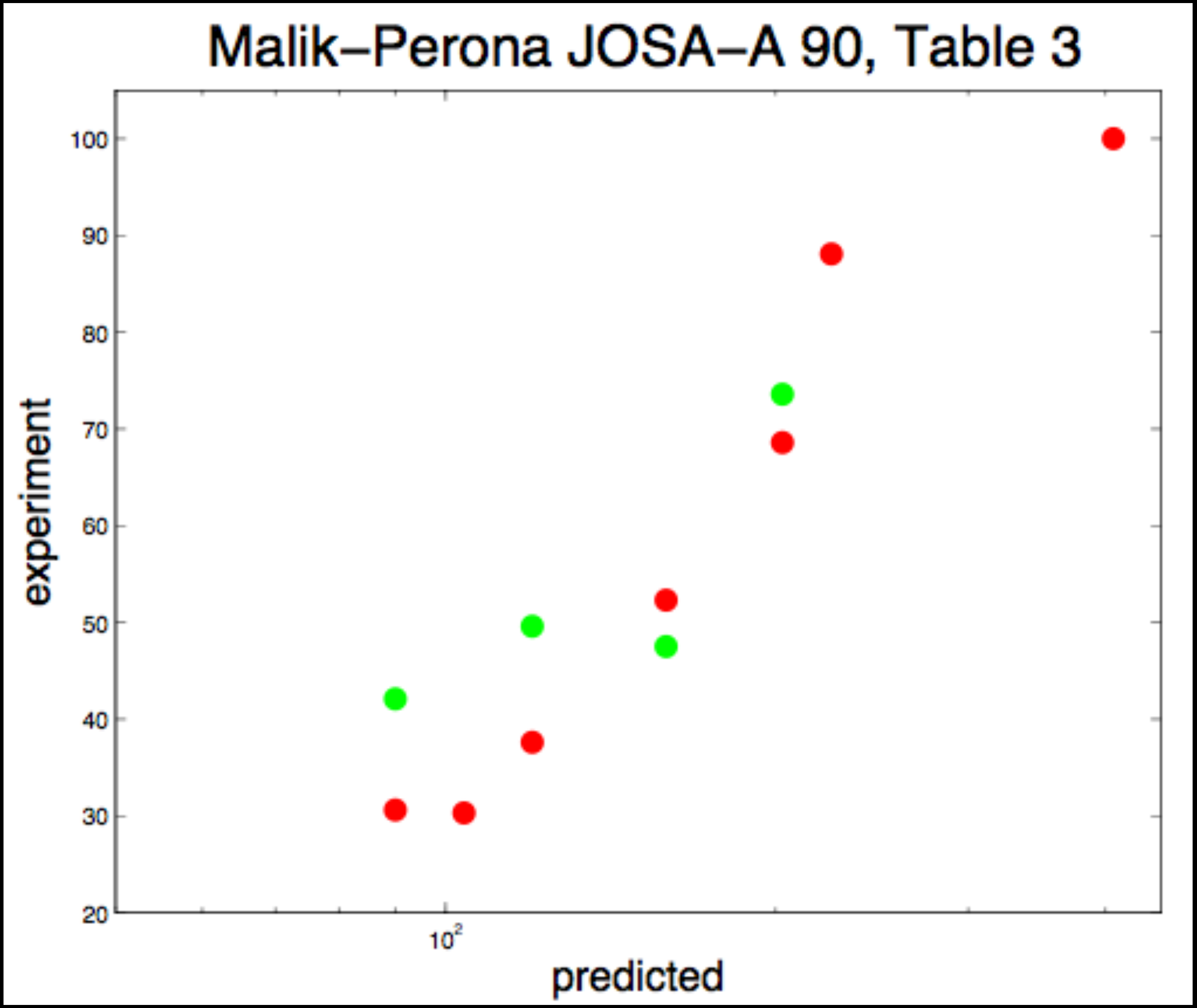
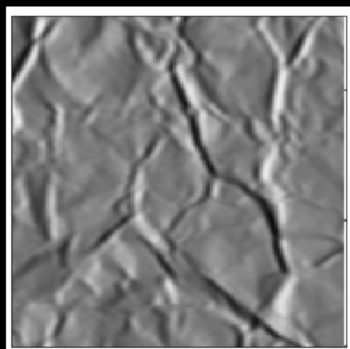


Table 3. Comparison of Predictions from Texture Segmentation Algorithm with Two Sets of Psychophysical Data[a]

Texture Pair	Discriminability Data Refs. 41 and 42	Discriminability Data Ref. 43	Discriminability Predicted Data
+ ○	100(saturated)	n.a.	407
+ □	88.1	n.a.	225
L +	68.6	0.736	203
L M	n.a.	n.a.	165
Δ↓	52.3	0.4–0.55	159
+ T	37.6	0.496	120
+ X	30.3	n.a.	104
T L	30.6	0.421	90*
L_L M_L	n.a.	n.a.	85
R-mirror-R	n.a.	n.a.	50*


Malik–Perona JOSA–A 90, Table 3
100
90
80
70
60
50
40
30
20
experiment
10^2
predicted

Texture Recognition



?

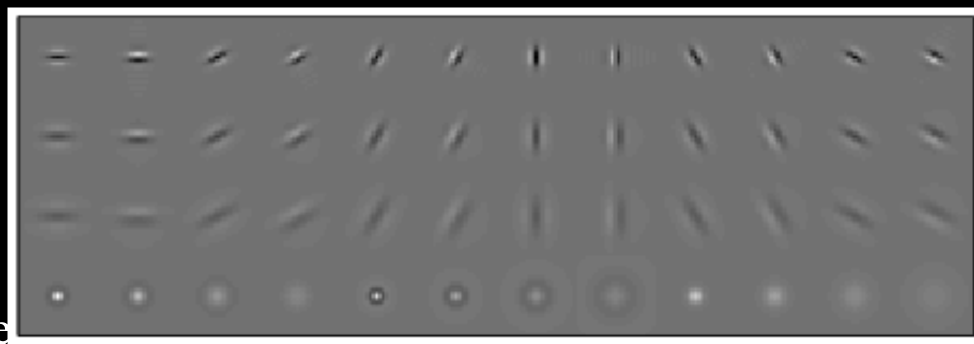
?

Felt?
Polyester?
Terrycloth?
Rough Plaster?
Leather?
Plaster?
Concrete?
Crumpled Paper?
Sponge?
Limestone?
Brick?

2D Textons

- Goal: find canonical local features in a texture;

    1) Filter image with linear filters:

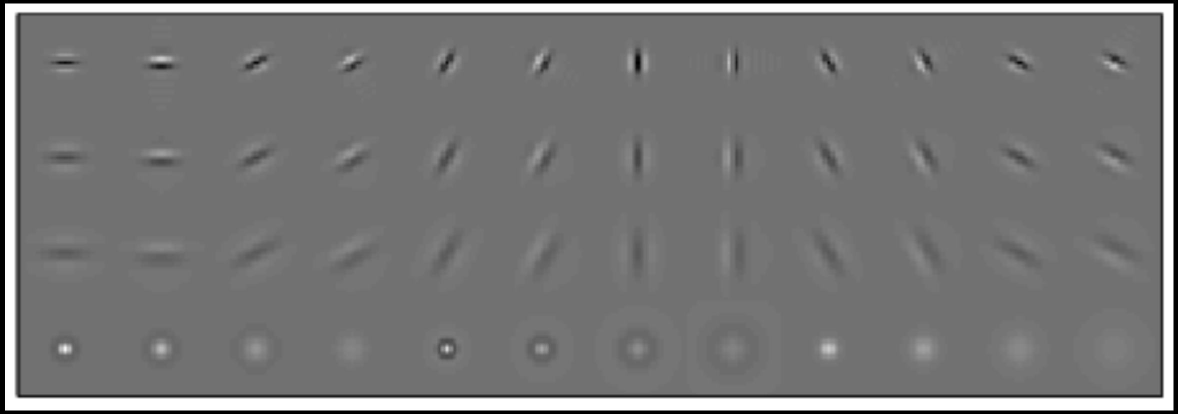


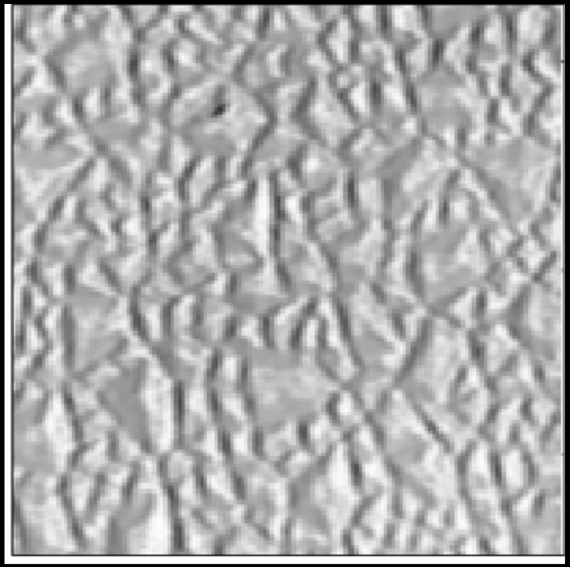
    2) Ve

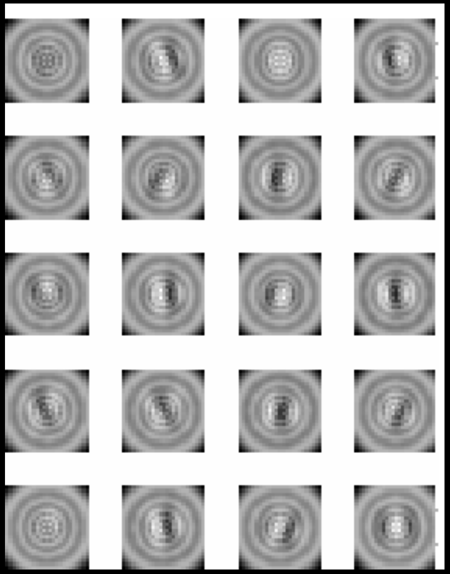
    3) Quantization centers are the textons.

- Spatial distribution of textons defines the texture;

2D Textons (cont’d)

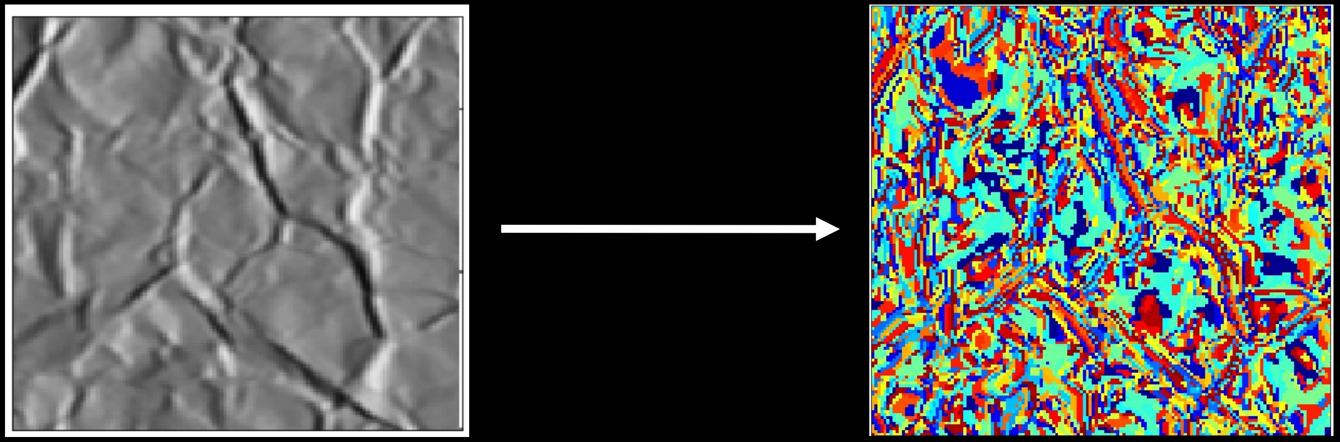






Texton Labeling

- Each pixel labeled to texton *i* (1 to K) which is *most similar in appearance*;
- Similarity measured by the Euclidean distance between the filter responses;

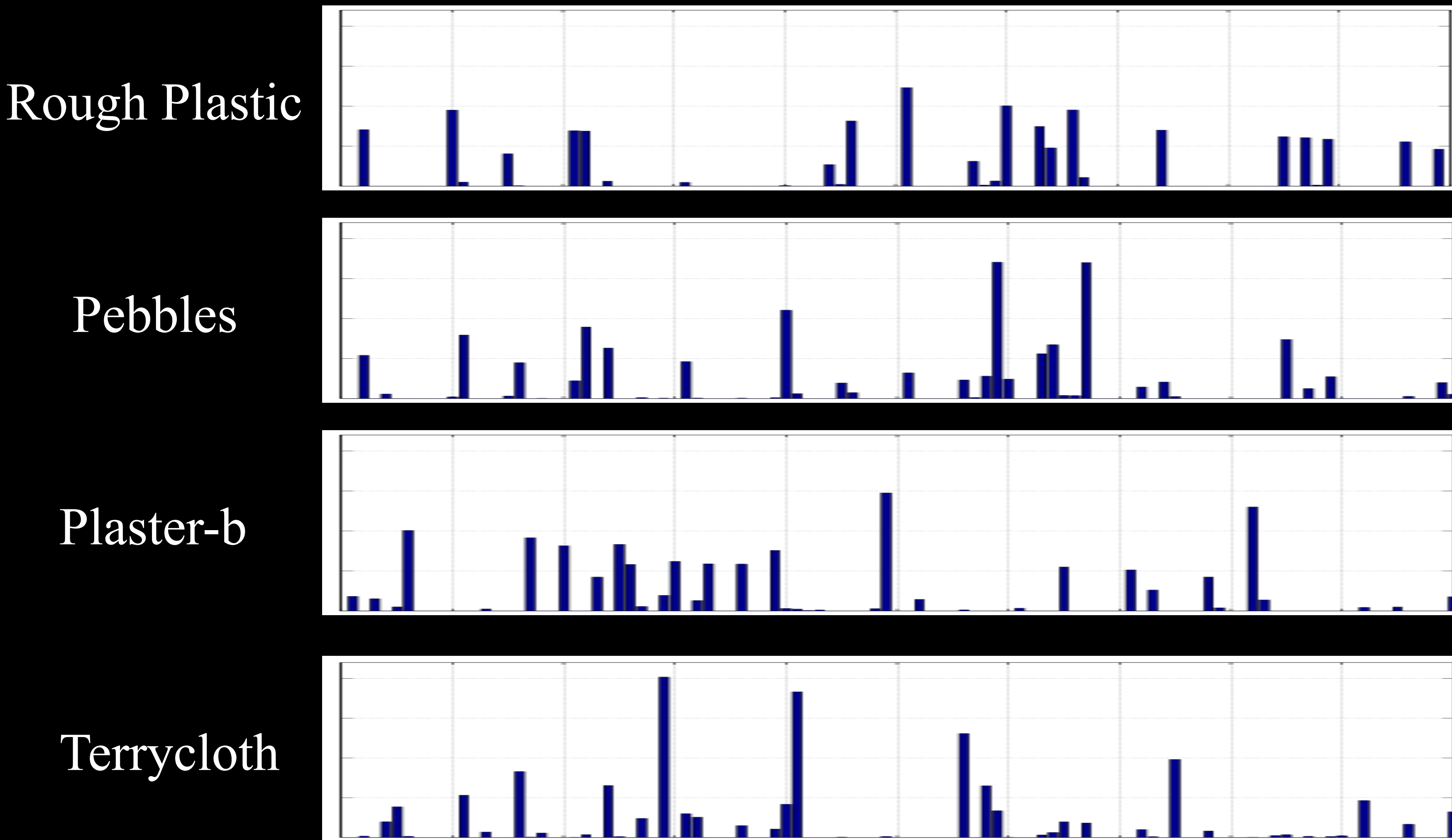


Material Representation

- Each material is now represented as a spatial arrangement of symbols from the texton vocabulary;

- Recognition --- ignore spatial arrangement,
  use histogram (K=100);

Histogram Models for Recognition (Leung & Malik, 1999)

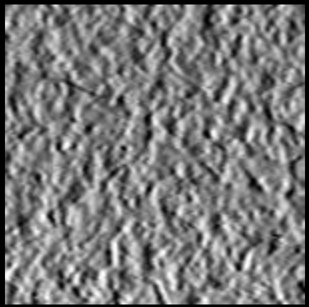


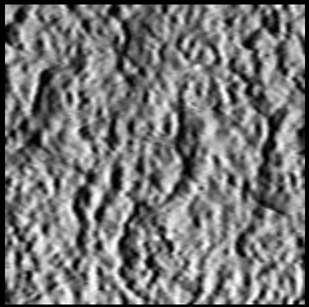
Similarity of materials

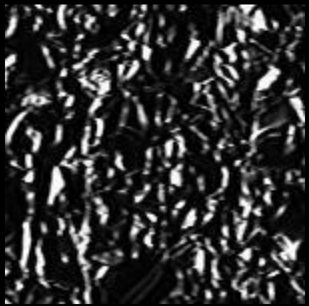
- Similarity between histograms measured using chi-square difference:

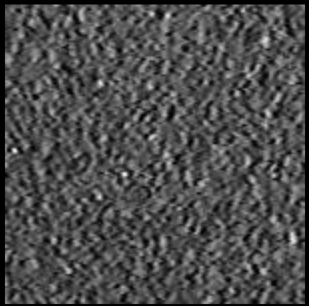
Similarity Matrix

	Felt	Terrycloth	Rough Plastic	Leather	Sandpaper	Pebbles	Plastera	Plasterb	Rough Paper	Artificial Grass	Rough Shingle	Aluminum Foil	Cork	Rough Tile
Felt	**0.6**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Terrycloth	0.0	**1.0**	0.0	0.0	0.3	0.0	0.1	0.2	0.0	0.0	0.0	0.0	0.0	0.0
Rough Plastic	0.0	0.0	**0.9**	0.0	0.0	0.0	0.2	0.1	0.0	0.0	0.0	0.0	0.0	0.0
Leather	0.2	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Sandpaper	0.0	0.1	0.0	0.0	**1.0**	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pebbles	0.0	0.0	0.0	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0
Plastera	0.0	0.1	0.2	0.0	0.1	0.0	**1.0**	0.7	0.0	0.0	0.0	0.0	0.0	0.0
Plasterb	0.0	0.2	0.1	0.0	0.0	0.0	0.8	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0
Rough Paper	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	**0.5**	0.0	0.0	0.0	0.0	0.0
Artificial Grass	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.1	0.1	0.0	0.0
Roof Shingle	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	**1.0**	0.1	0.0	0.0
Aluminum Foil	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	**1.0**	0.0	0.0
Cork	0.0	0.0	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.2
Rough Tile	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.9**

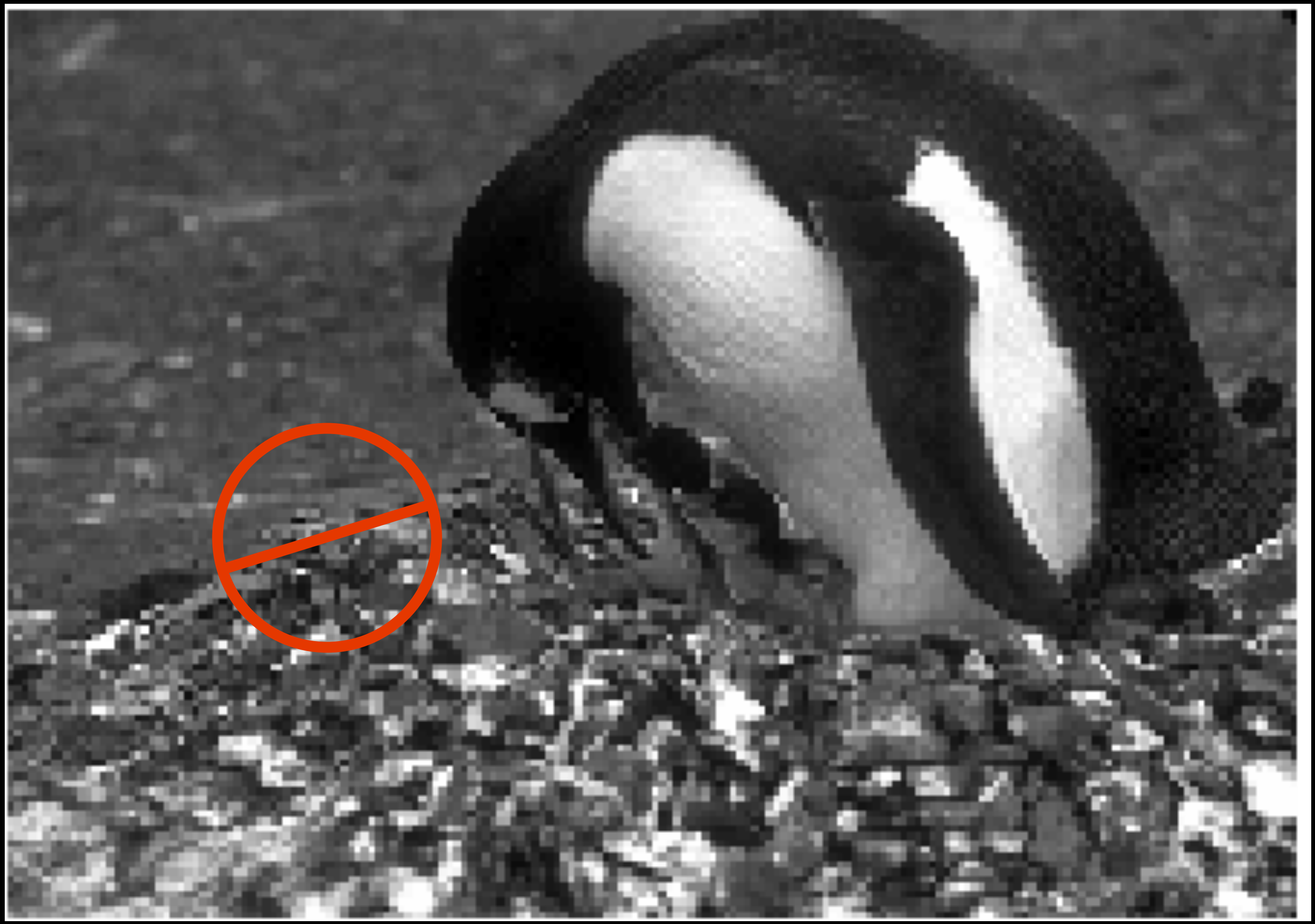

Plaster-a


Plaster-b


Aluminum Foil

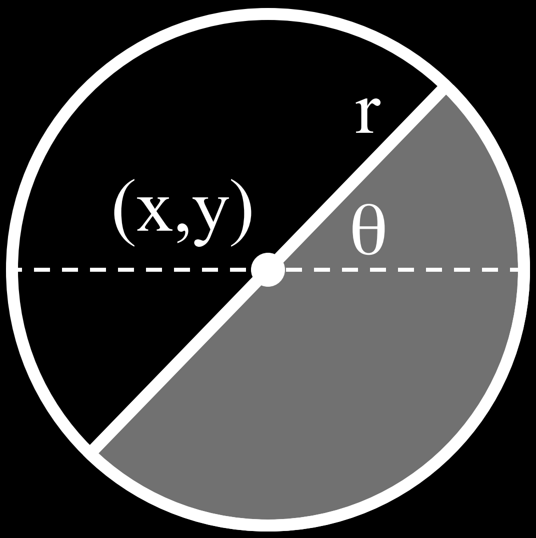

Cork

Oriented Feature Gradient

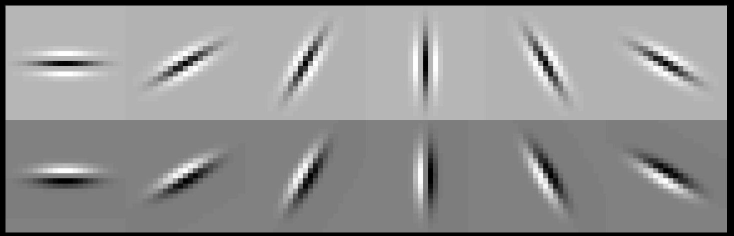


Individual Features

- 1976 CIE L*a*b* colorspace
- Brightness Gradient BG(x,y,r,θ)
  - Difference of L* distributions
- Color Gradient CG(x,y,r,θ)
  - Difference of a*b* distributions

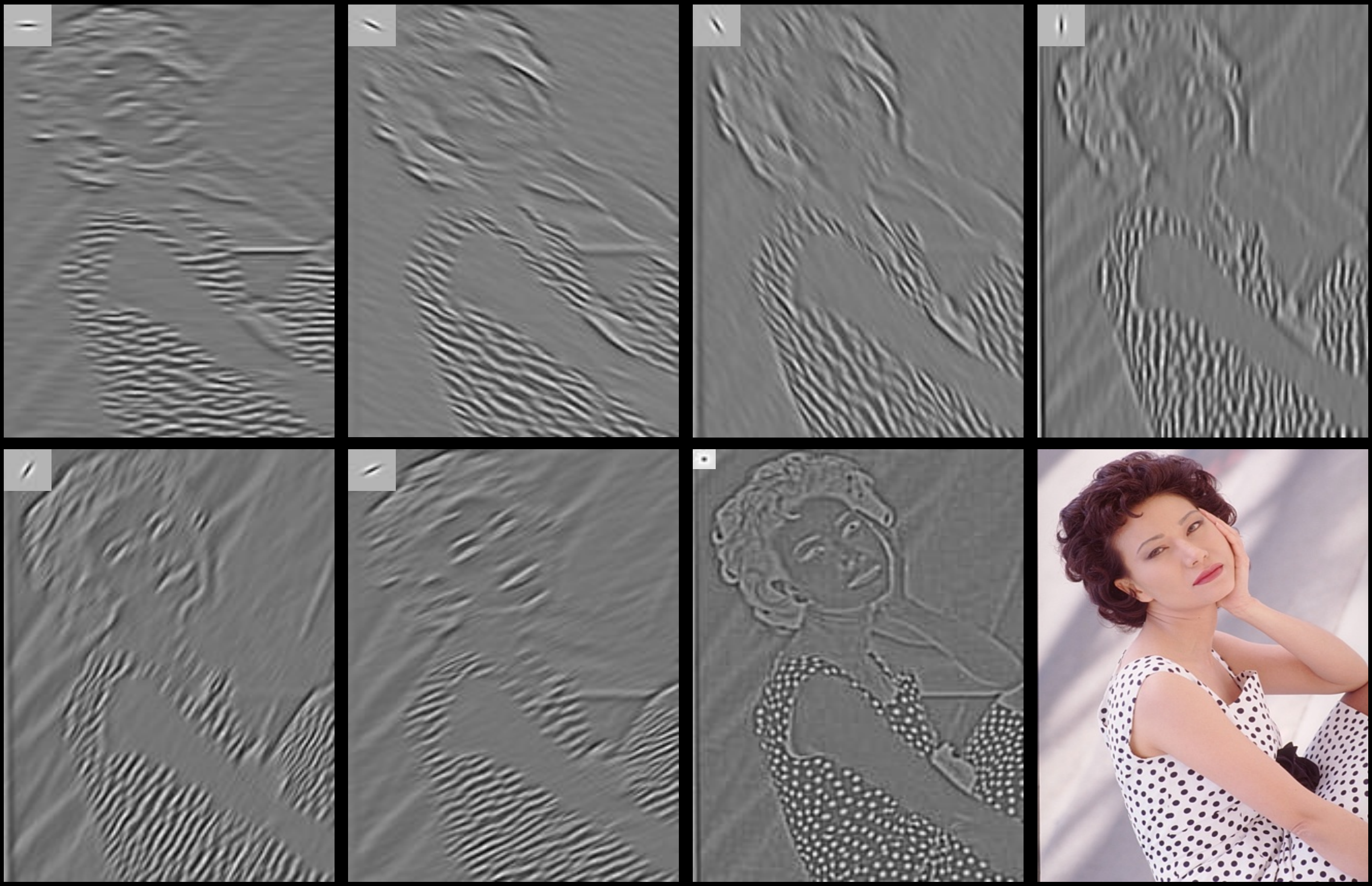


- Texture Gradient TG(x,y,r,θ)
  - Difference of distributions of V1-like filter responses

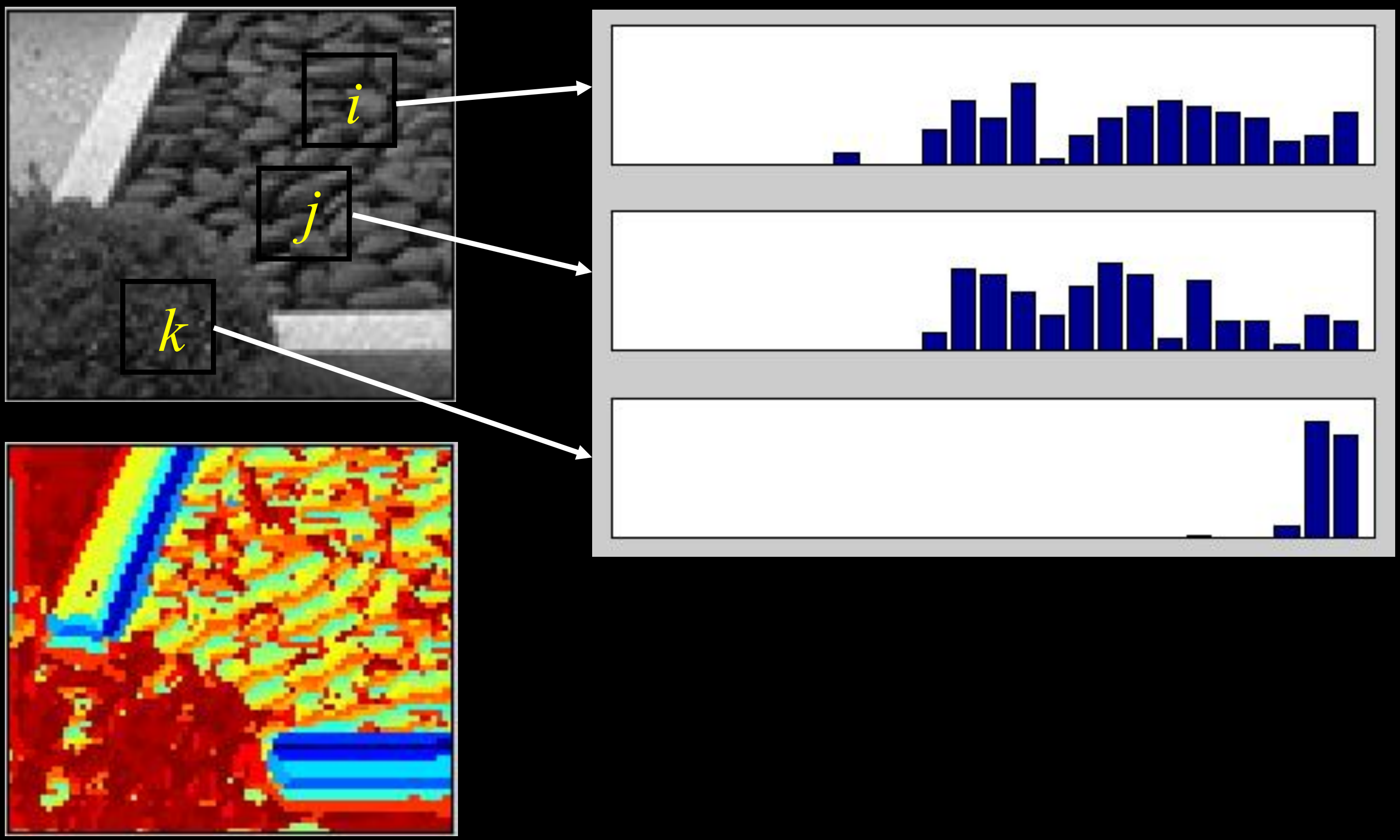


These are combined using logistic regression

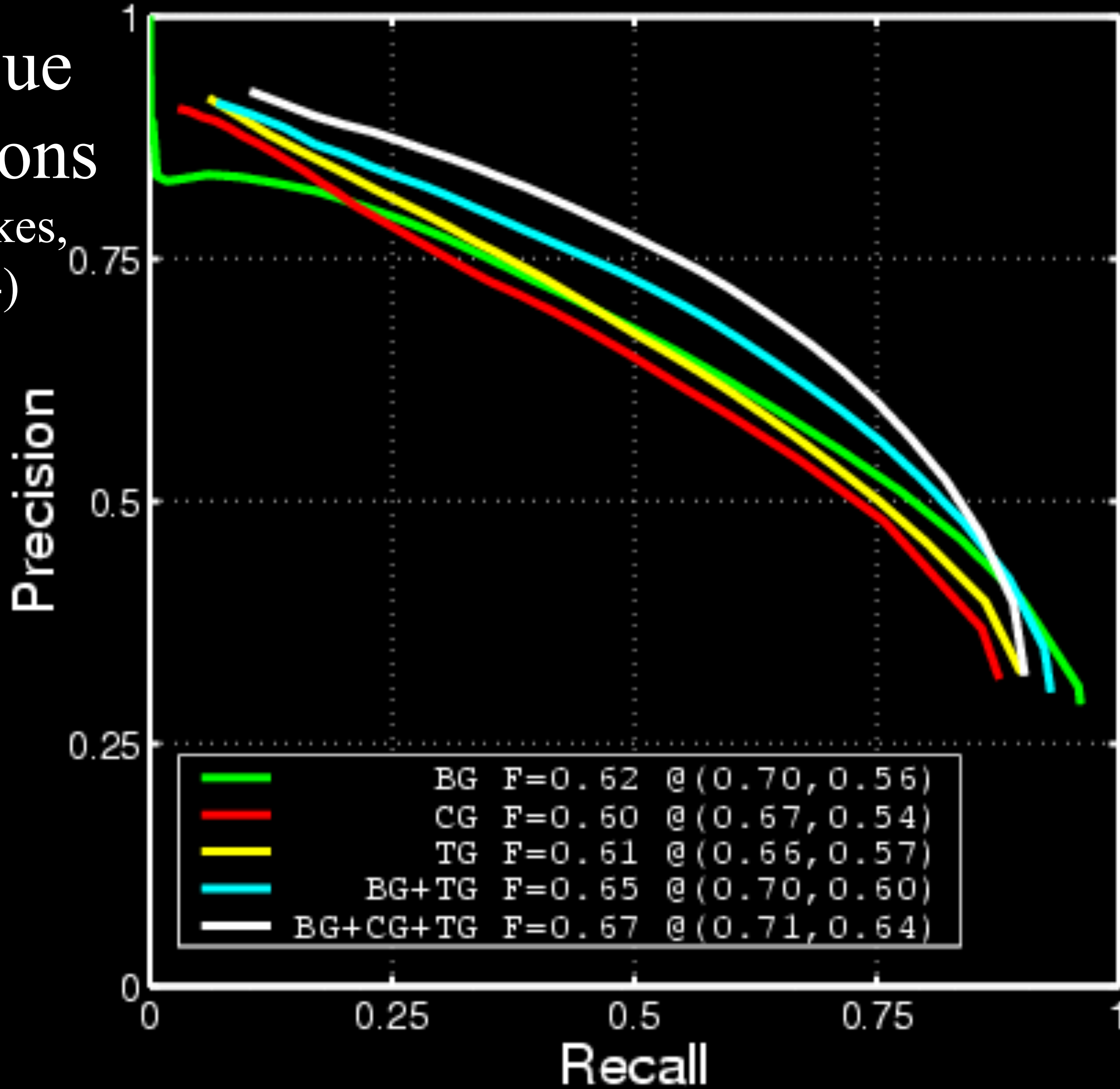
Filter Outputs



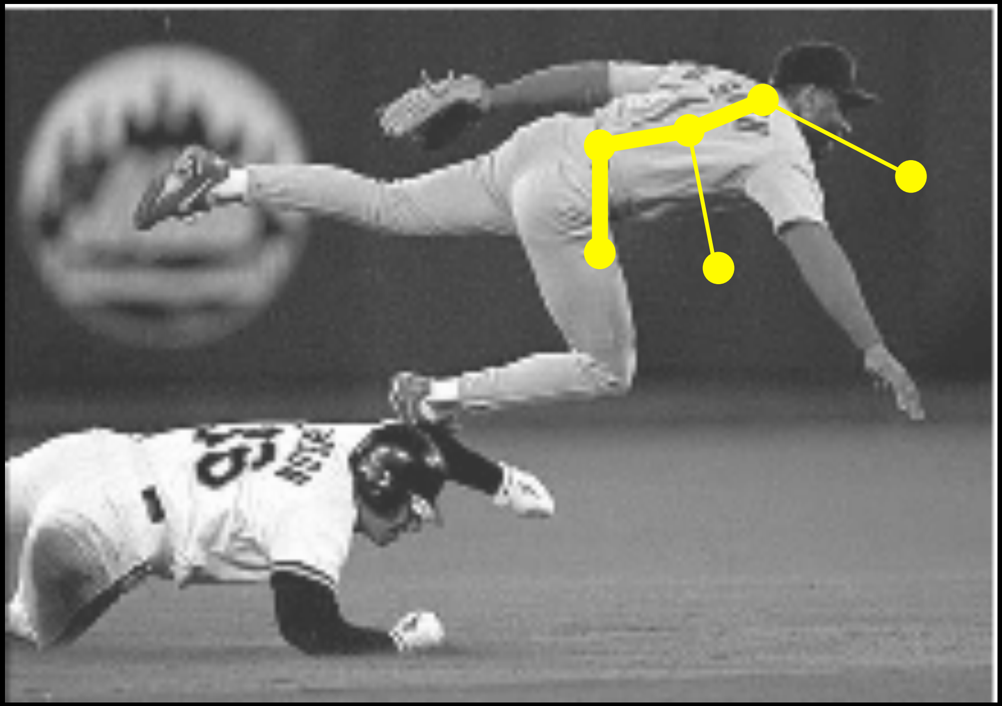
Texture gradient = Chi square distance between texton histograms in half disks across edge



Various Cue Combinations
(Martin, Fowlkes, Malik, 2004)



Exploiting global constraints: Image Segmentation as Graph Partitioning



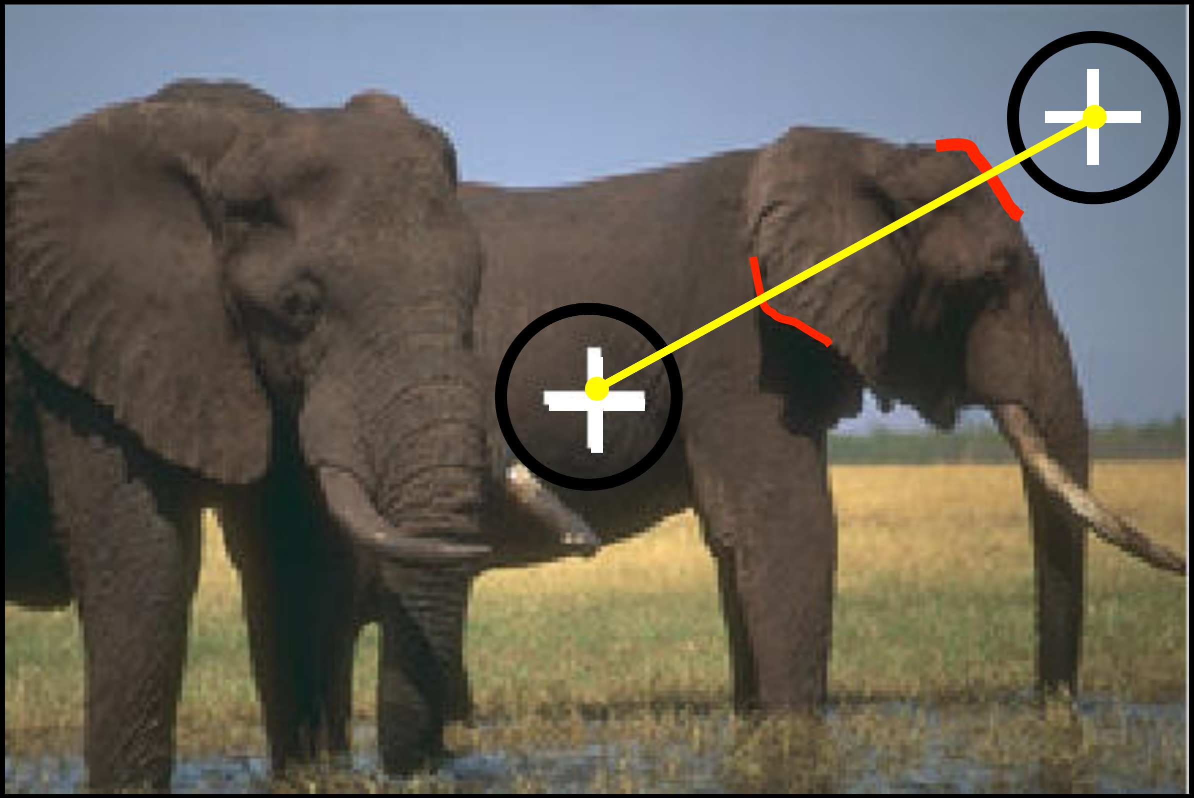
V: image pixels

E: connections between pairs of nearby pixels

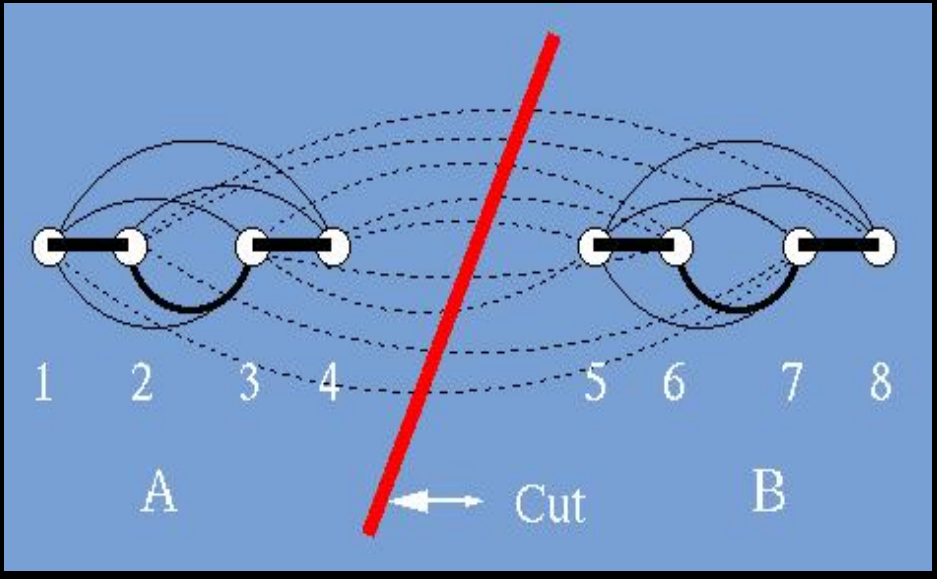
Partition graph so that similarity within group is large and similarity between groups is small -- ***Normalized Cuts*** [Shi & Malik 97]

Wij small when intervening contour strong, small when weak..

Cij = max Pb(x,y) for (x,y) on line segment ij; Wij = exp (- Cij / σ)



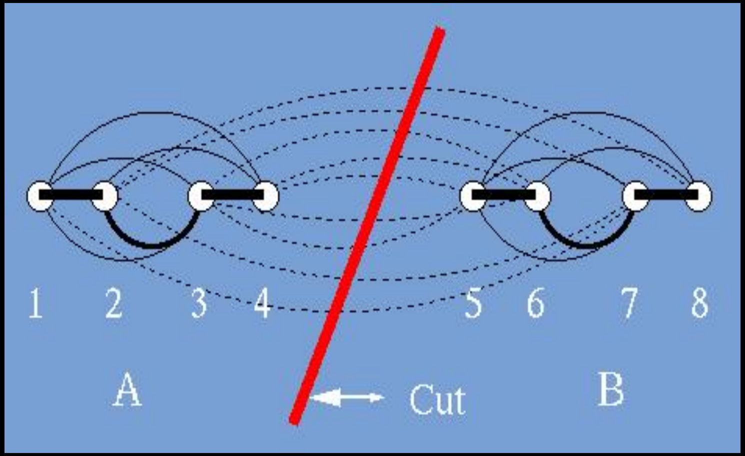
How to partition a graph



$$cut(A,B) = \sum_{u \in A, v \in B} w(u,v)$$

- We can find the minimum cut efficiently, but this tends to break the graph into isolated little pieces

Normalized Cut is a better measure ..



- We normalize by the total volume of connections

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$$

where $assoc(A,V) = \sum_{u \in A, t \in V} w(u,t)$

Solving the Normalized Cut problem

- Exact discrete solution to Ncut is NP-hard even on regular grid [Papadimitriou' 97]
- We first transform to

$$min_{\boldsymbol{x}} Ncut(\boldsymbol{x}) = min_{\boldsymbol{y}} \frac{\boldsymbol{y}^T(\mathbf{D}-\mathbf{W})\boldsymbol{y}}{\boldsymbol{y}^T\mathbf{D}\boldsymbol{y}}$$

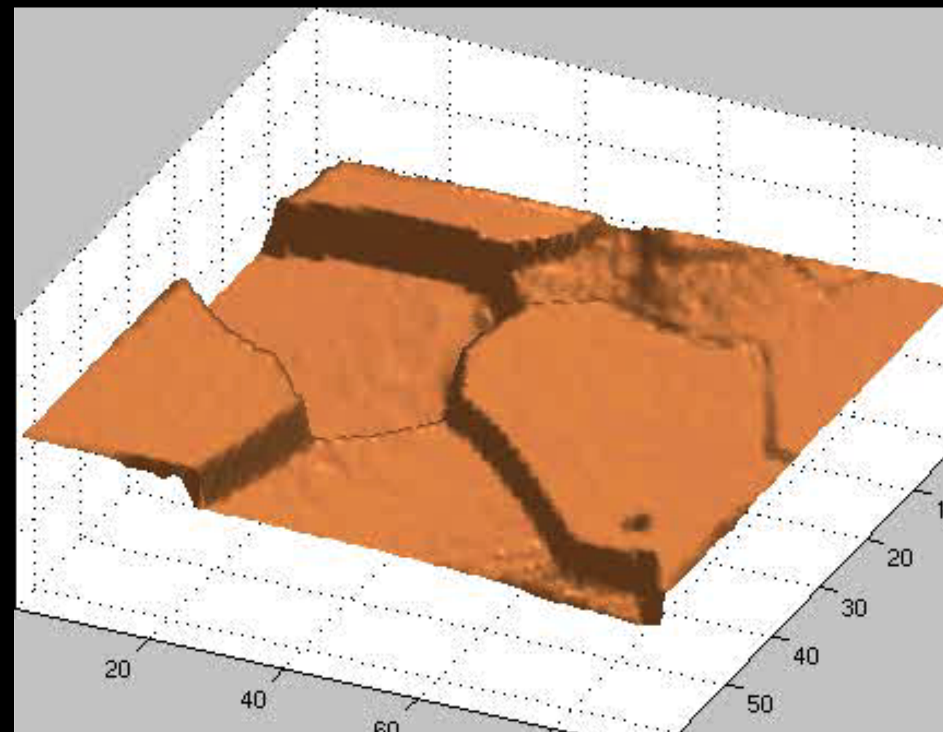
with the condition $y(i) \in \{1, -b\}$ and $y^T\mathbf{D}\mathbf{1} = 0$

- Drawing on spectral graph theory, good approximation can be obtained by solving a generalized eigenvalue problem.

$$(\mathbf{D}-\mathbf{W})\boldsymbol{y} = \lambda\mathbf{D}\boldsymbol{y}.$$

Normalized Cuts as a Spring-Mass system

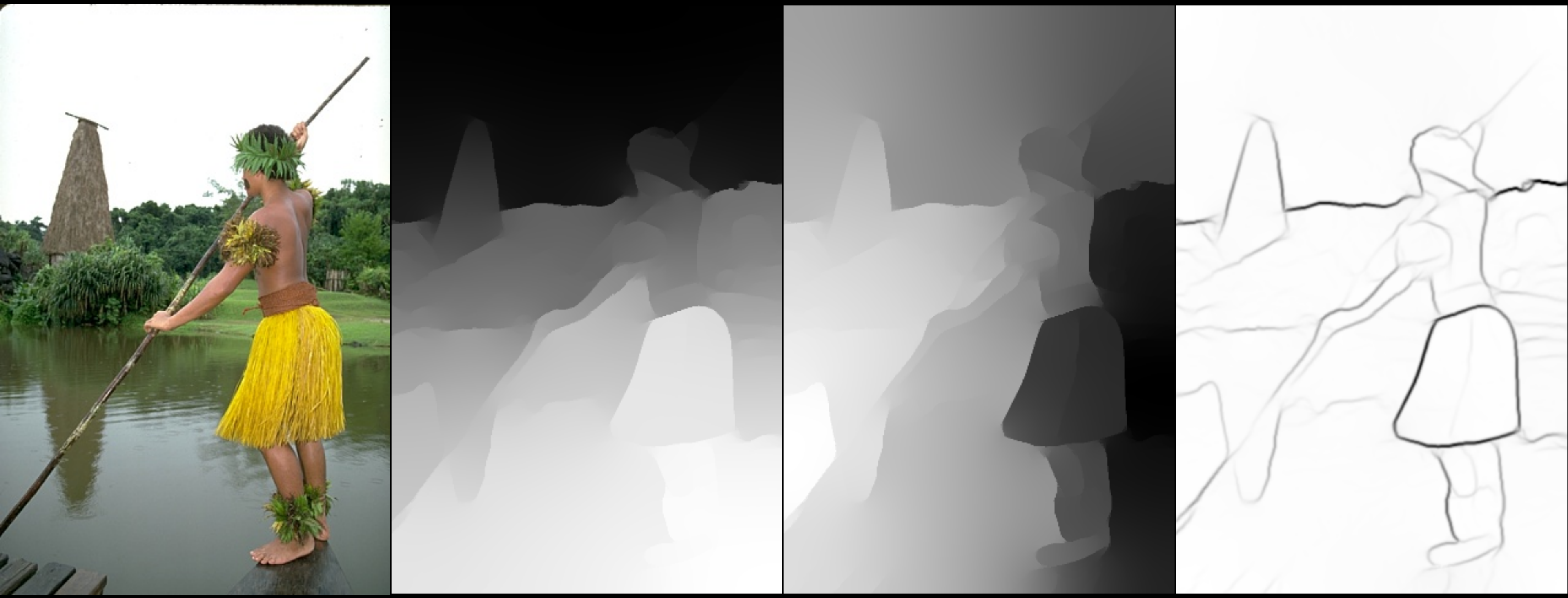
- Each pixel is a point mass; each connection is a spring:

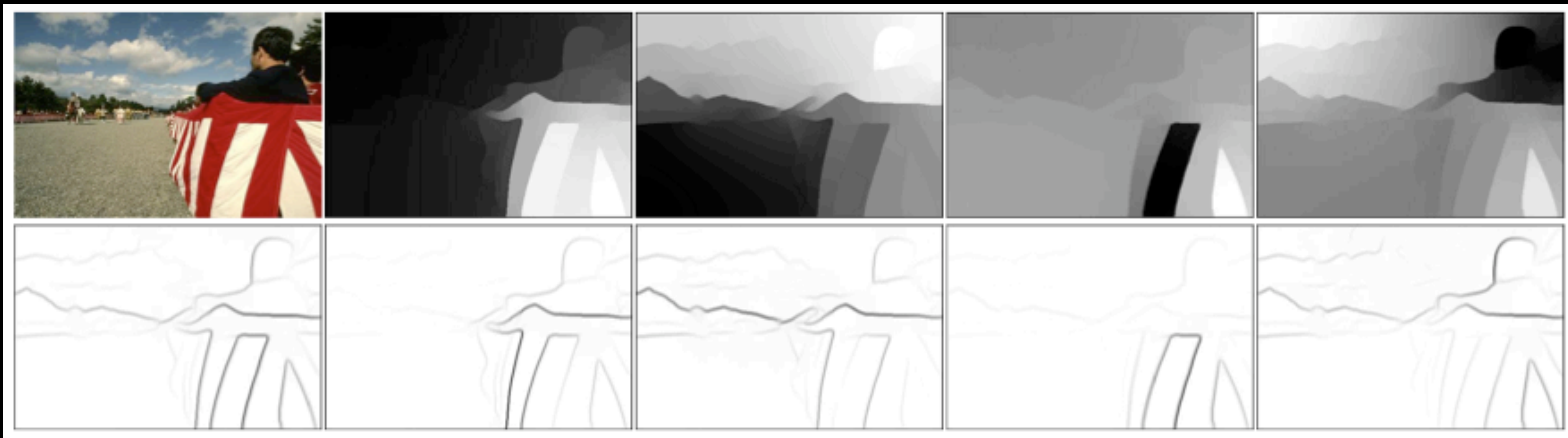


- Fundamental modes are generalized eigenvectors of

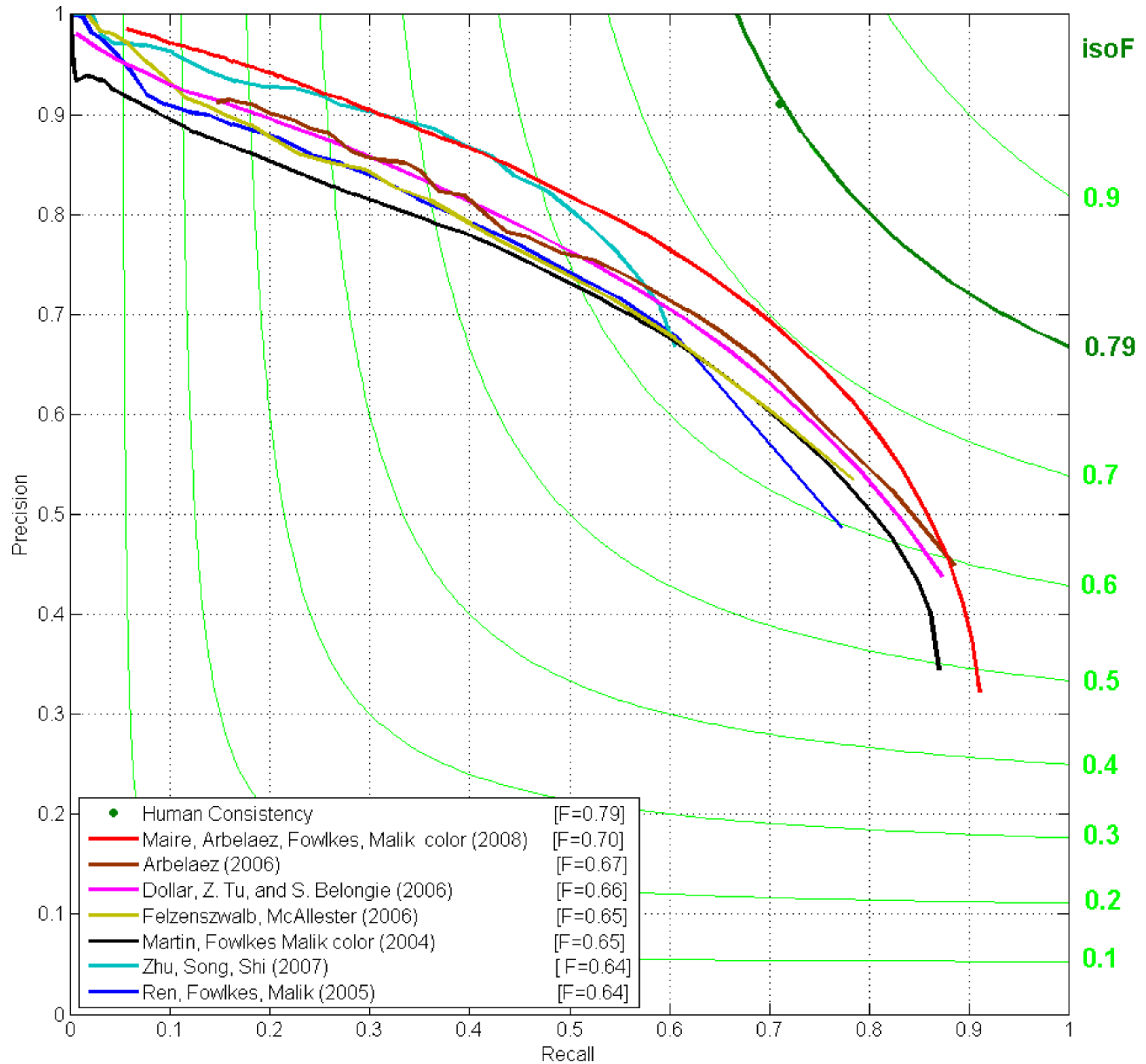
$$(D - W)\, y = \lambda D y$$

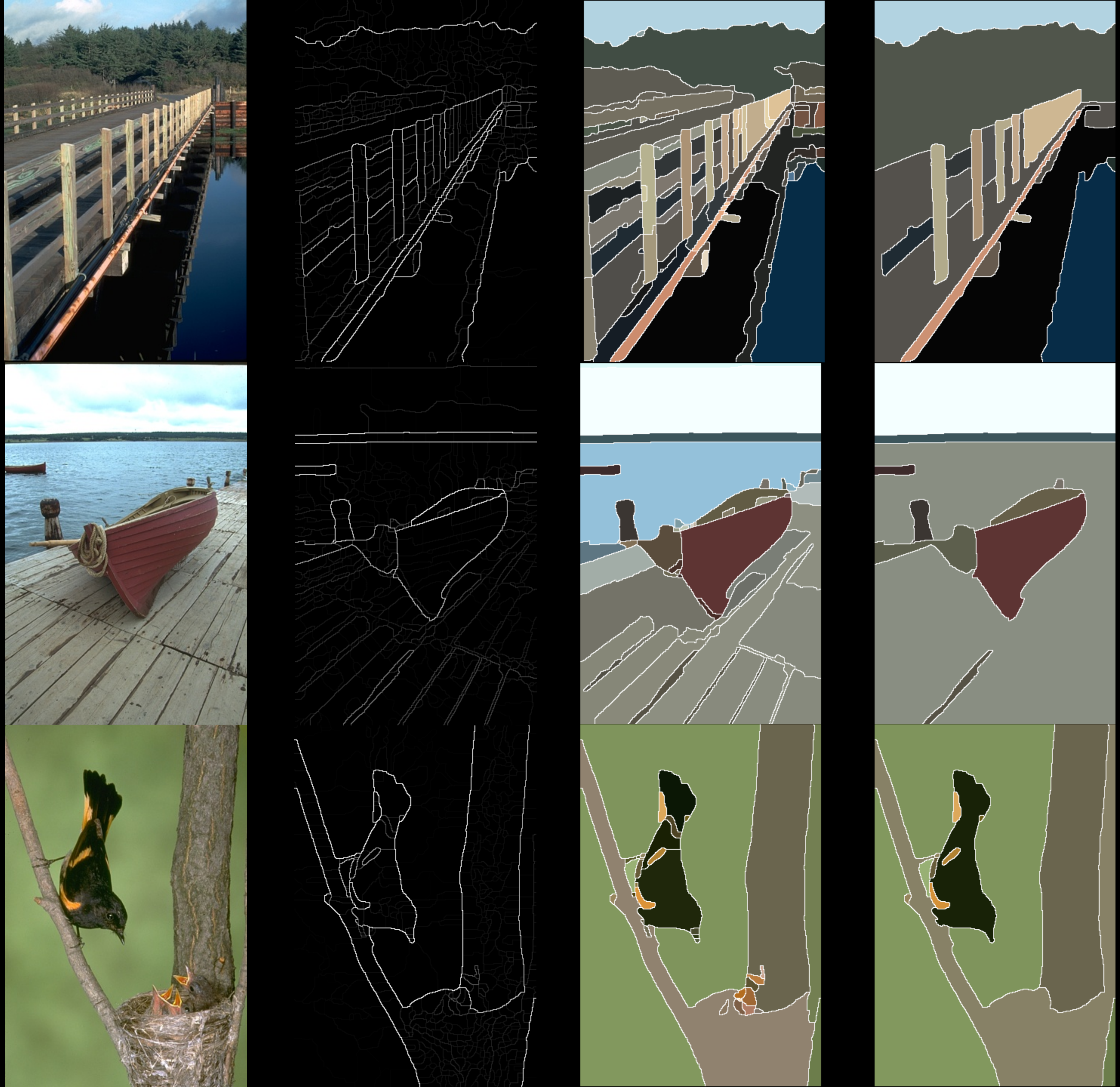
Eigenvectors carry contour information

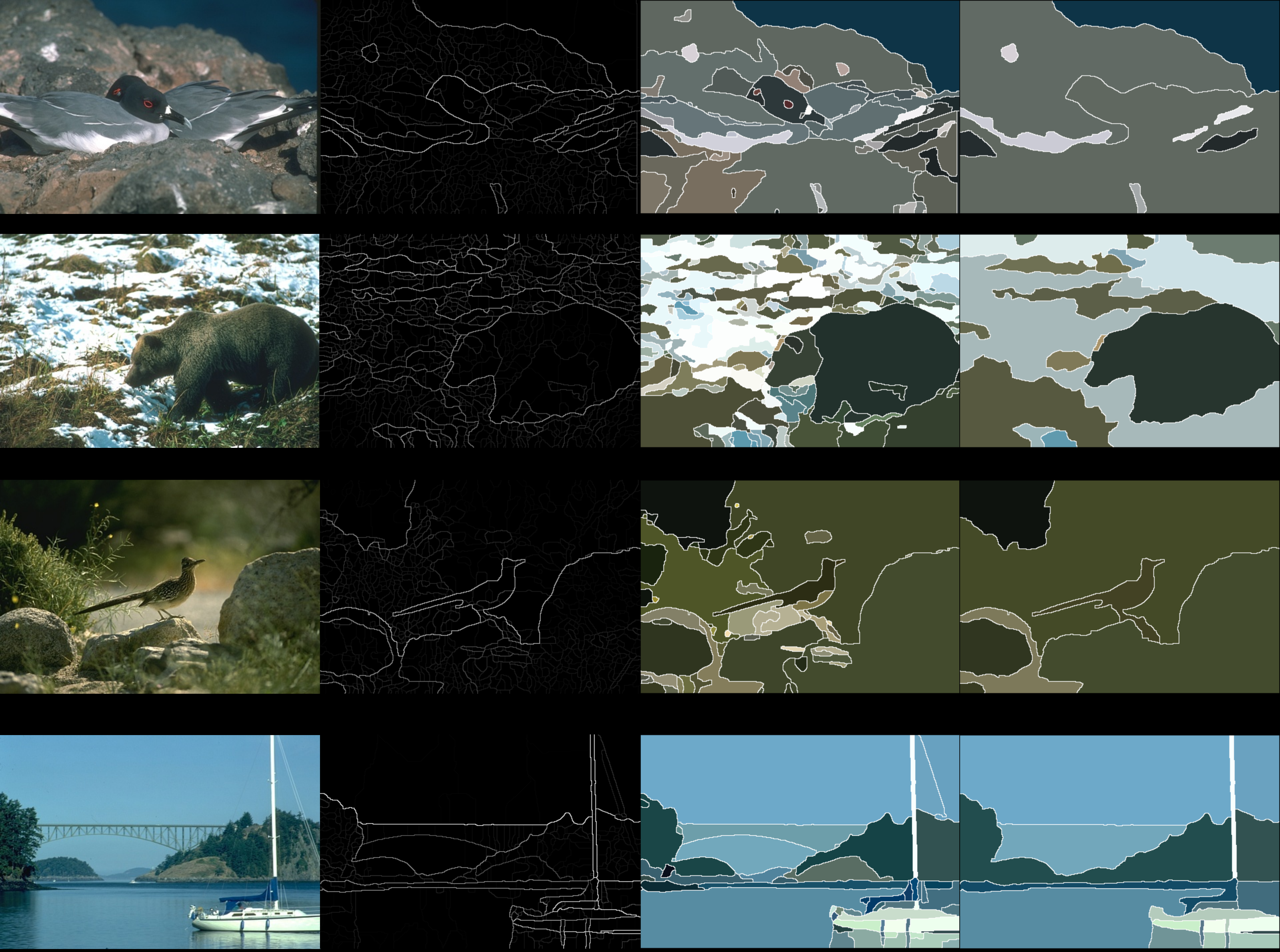


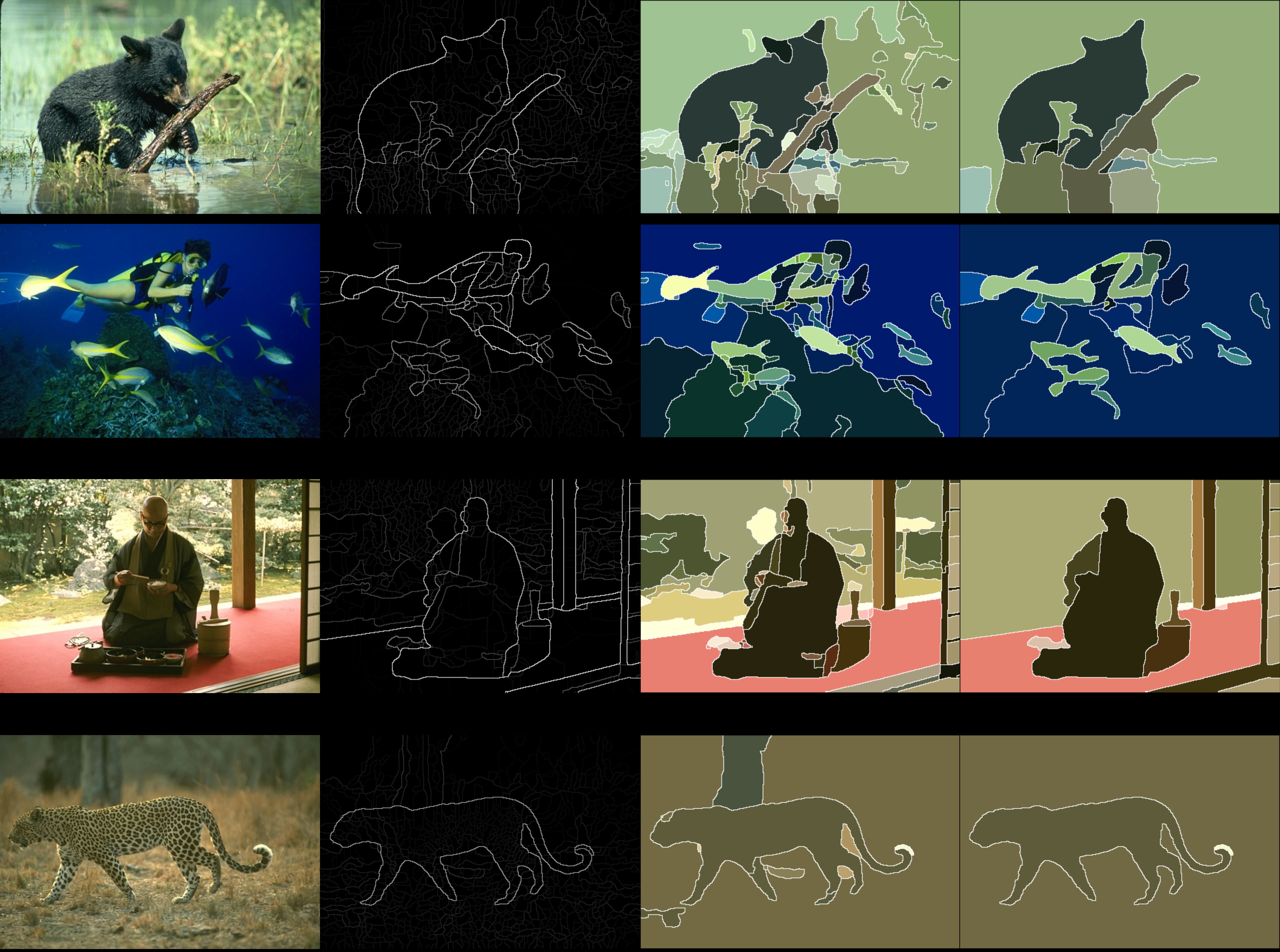


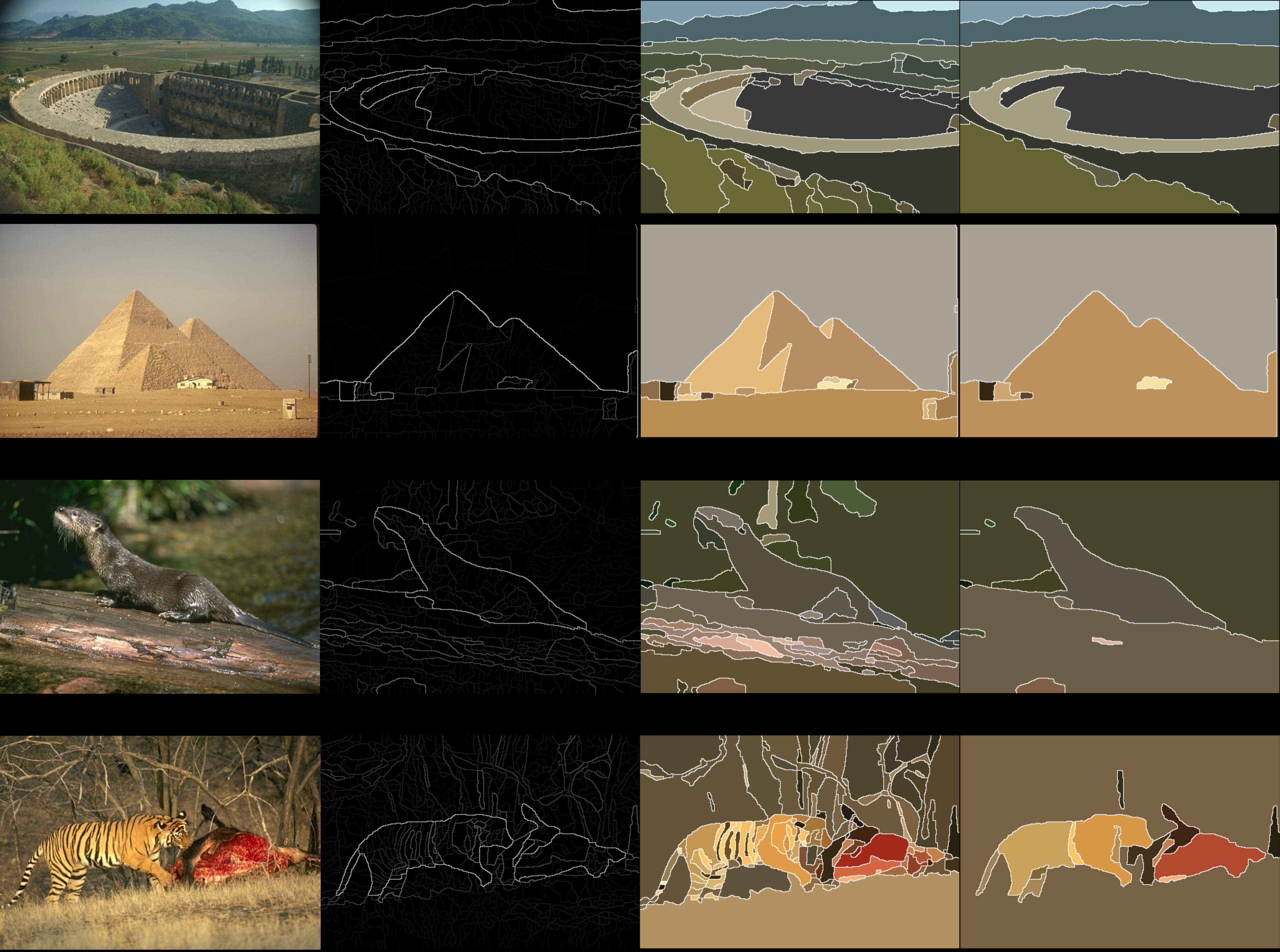
Comparison to other approaches (2009)











Fast Edge Detection Using Structured Forests

Piotr Dollár and C. Lawrence Zitnick
Microsoft Research
{pdollar,larryz}@microsoft.com

Abstract—Edge detection is a critical component of many vision systems, including object detectors and image segmentation algorithms. Patches of edges exhibit well-known forms of local structure, such as straight lines or T-junctions. In this paper we take advantage of the structure present in local image patches to learn both an accurate and computationally efficient edge detector. We formulate the problem of predicting local edge masks in a structured learning framework applied to random decision forests. Our novel approach to learning decision trees robustly maps the structured labels to a discrete space on which standard information gain measures may be evaluated. The result is an approach that obtains realtime performance that is orders of magnitude faster than many competing state-of-the-art approaches, while also achieving state-of-the-art edge detection results on the BSDS500 Segmentation dataset and NYU Depth dataset. Finally, we show the potential of our approach as a general purpose edge detector by showing our learned edge models generalize well across datasets.

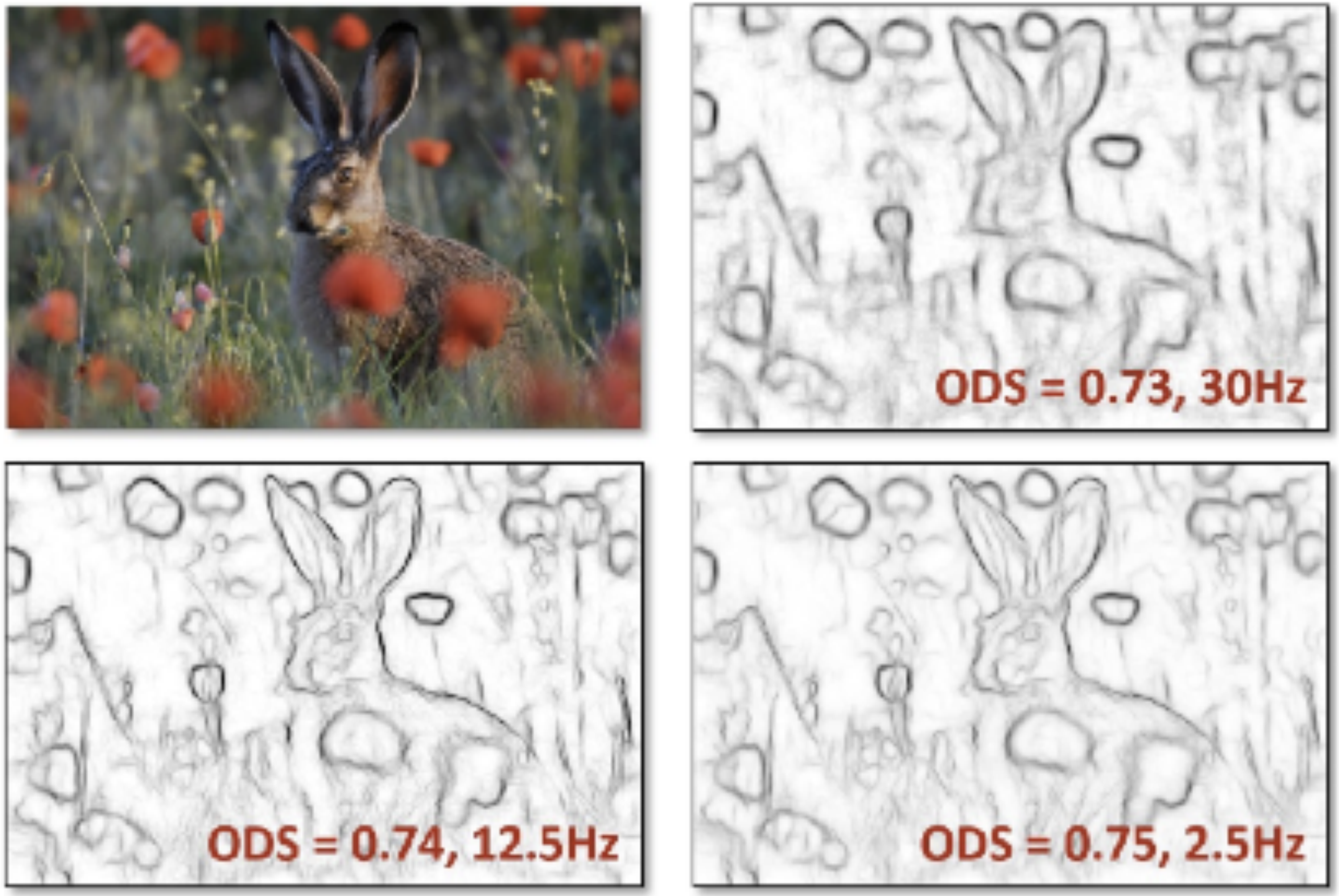


Fig. 1. Edge detection results using three versions of our Structured Edge (SE) detector demonstrating tradeoffs in accuracy vs. runtime. We obtain realtime performance while simultaneously achieving state-of-the-art results. ODS numbers were computed on BSDS [1] on which the popular gPb detector [1] achieves a score of .73. The variants shown include SE, SE+SH, and SE+MS+SH, see §4 for details.

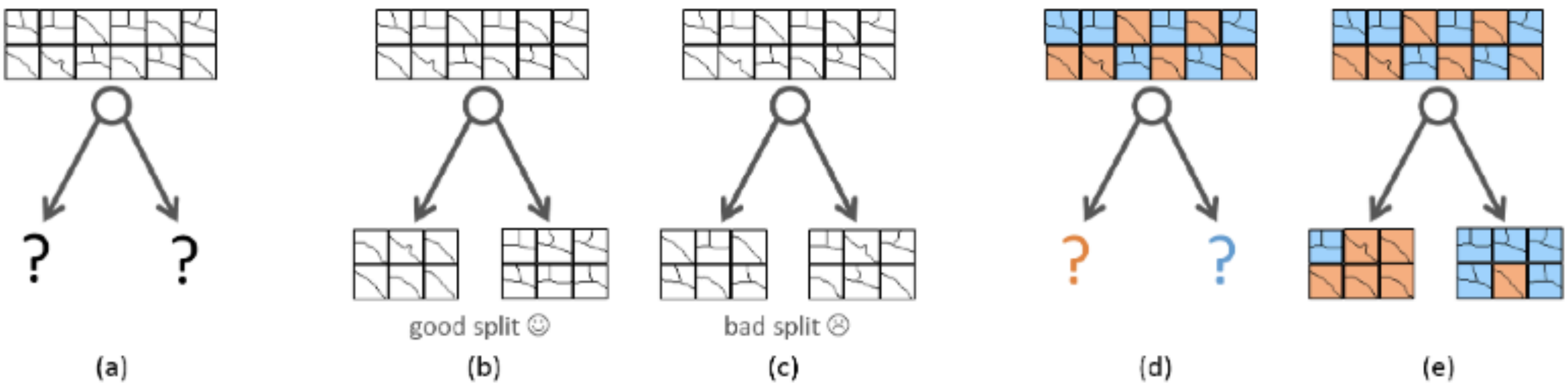


Fig. 2. Illustration of the decision tree node splits: (a) Given a set of structured labels such as segments, a splitting function must be determined. Intuitively a good split (b) groups similar segments, whereas a bad split (c) does not. In practice we cluster the structured labels into two classes (d). Given the class labels, a standard splitting criterion, such as Gini impurity, may be used (e).

$x \in \mathbb{R}^{32\times32\times K}$ where K is the number of channels. We use features of two types: pixel lookups $x(i,j,k)$ and pairwise differences $x(i_1,j_1,k) - x(i_2,j_2,k)$, see §2.

Inspired by Lim et al. [31], we use a similar set of color and gradient channels (originally developed for fast pedestrian detection [12]). We compute 3 color channels in CIE-LUV color space along with normalized gradient magnitude at 2 scales (original and half resolution). Additionally, we split each gradient magnitude channel into 4 channels based on orientation. The result is 3 color, 2 magnitude and 8 orientation channels, for a total of 13 channels.

We blur the channels with a radius 2 triangle filter and downsample by a factor of 2, resulting in $32\cdot32\cdot13/4 = 3328$ candidate features x. Motivated by [31], we also compute pairwise difference features. We apply a large triangle blur to each channel (8 pixel radius), and downsample to a resolution of 5×5. Sampling all candidate pairs and computing their differences yields an additional $\binom{5\cdot5}{2} = 300$ candidate features per channel, resulting in 7228 total candidate features.

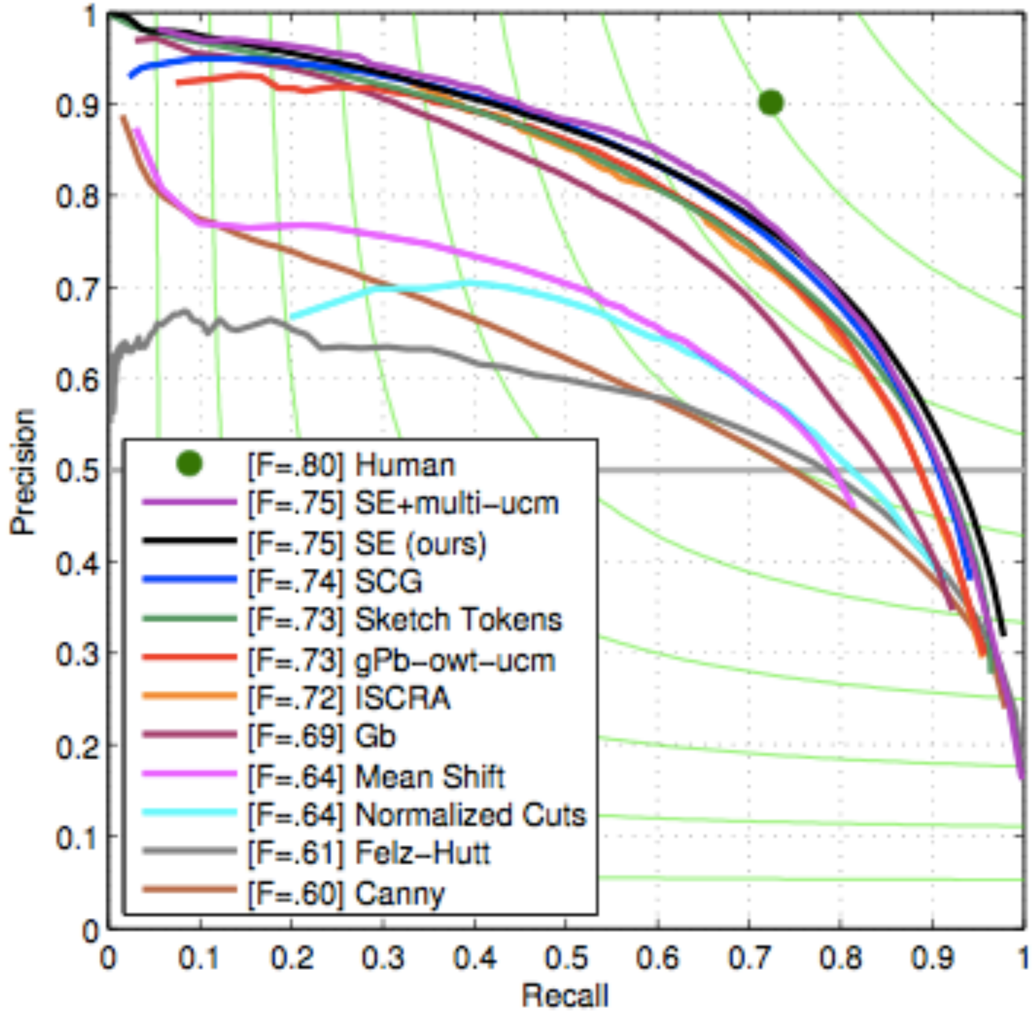
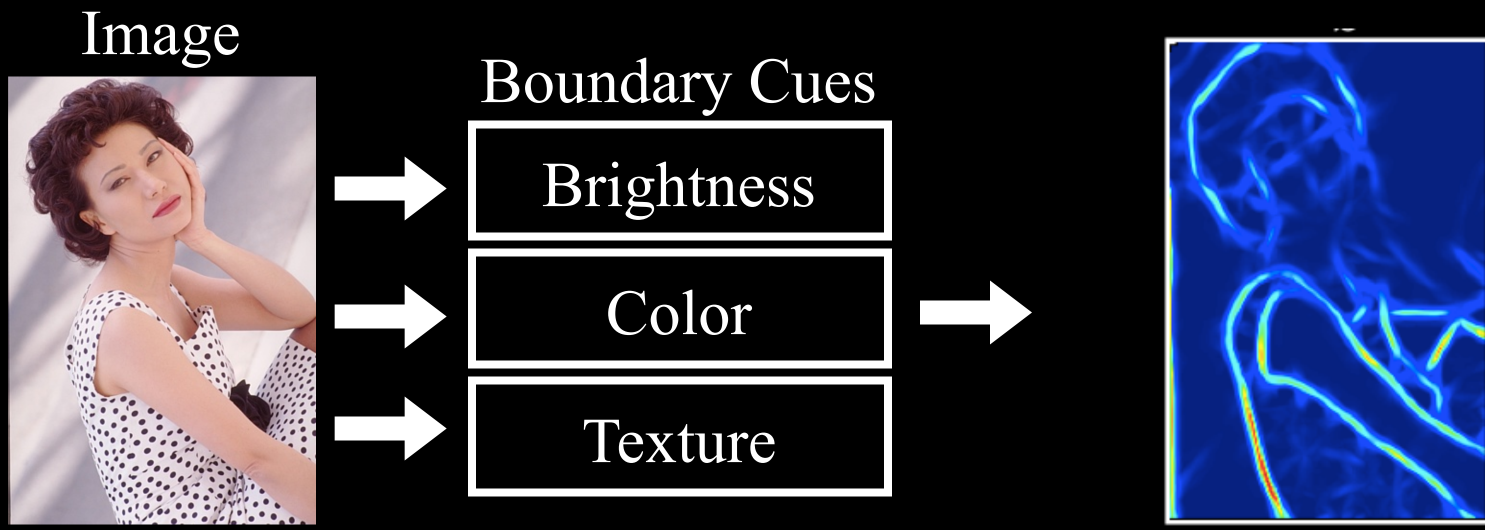


Fig. 9. Results on BSDS500. Structured edges (SE) and SE coupled with hierarchical multiscale segmentation (SE+multi-ucm) [2] achieve top results. For the SE result we report the SE+MS+SH variant. See Table 1 for additional details including method citations and runtimes. SE is orders of magnitude faster than nearly all edge detectors with comparable accuracy.

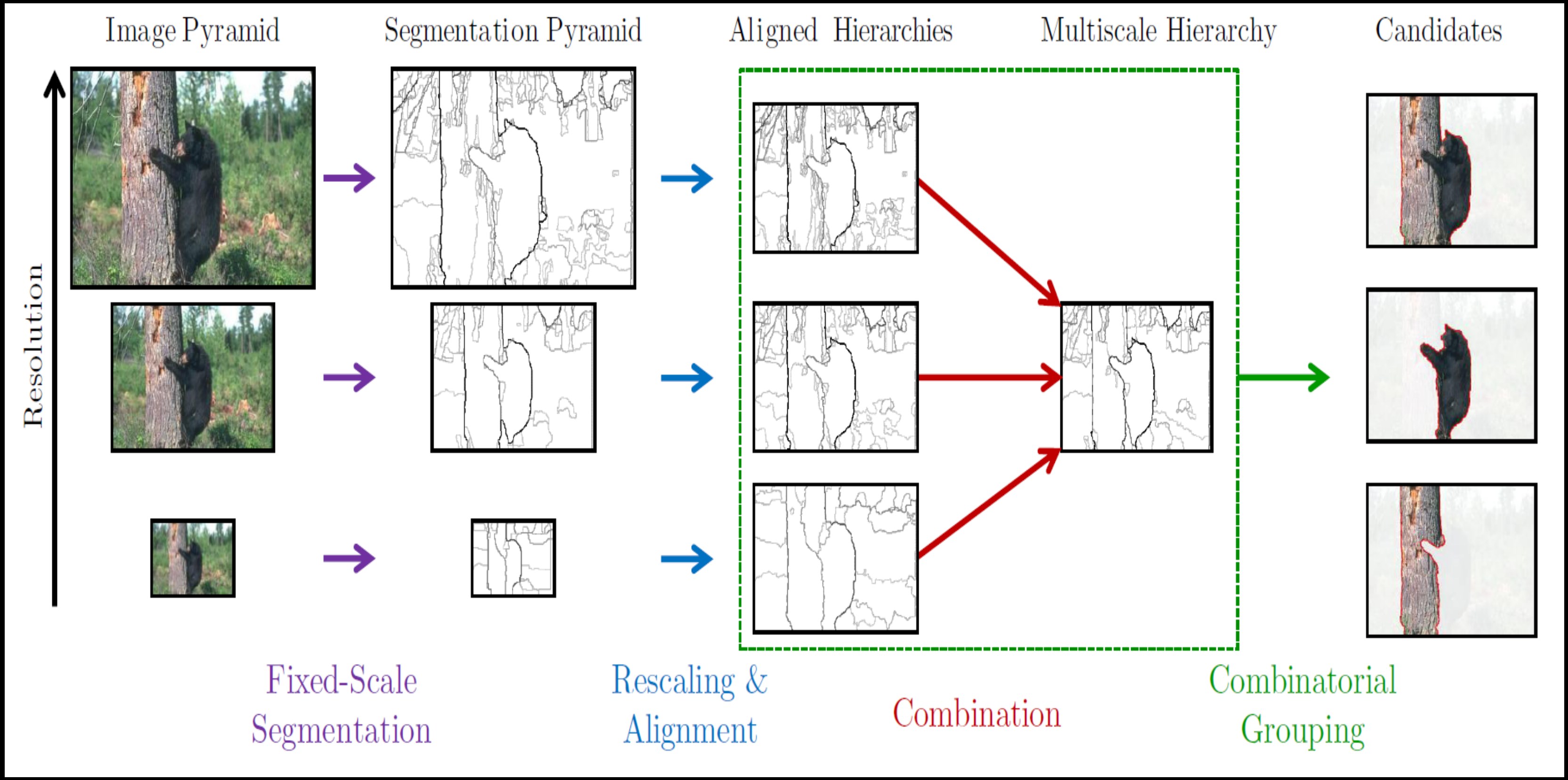
Martin, Fowlkes, Malik PAMI 04



We trained a detector that combine multiple cues to find the posterior probability of a boundary/edge $P_b(x,y,\theta)$. After that we used Normalized Cuts (Shi & Malik) to find regions.

Multiscale Combinatorial Regions

Arbelaez, Pont-Tuset, Barron, Marques & Malik, CVPR 2014



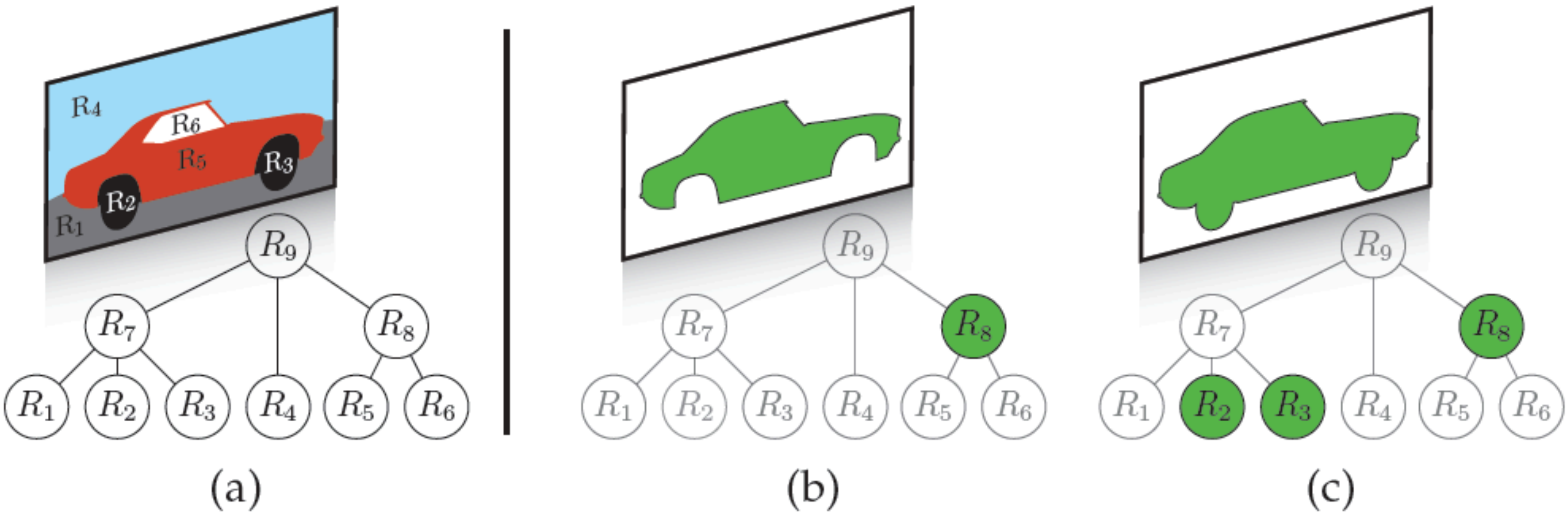
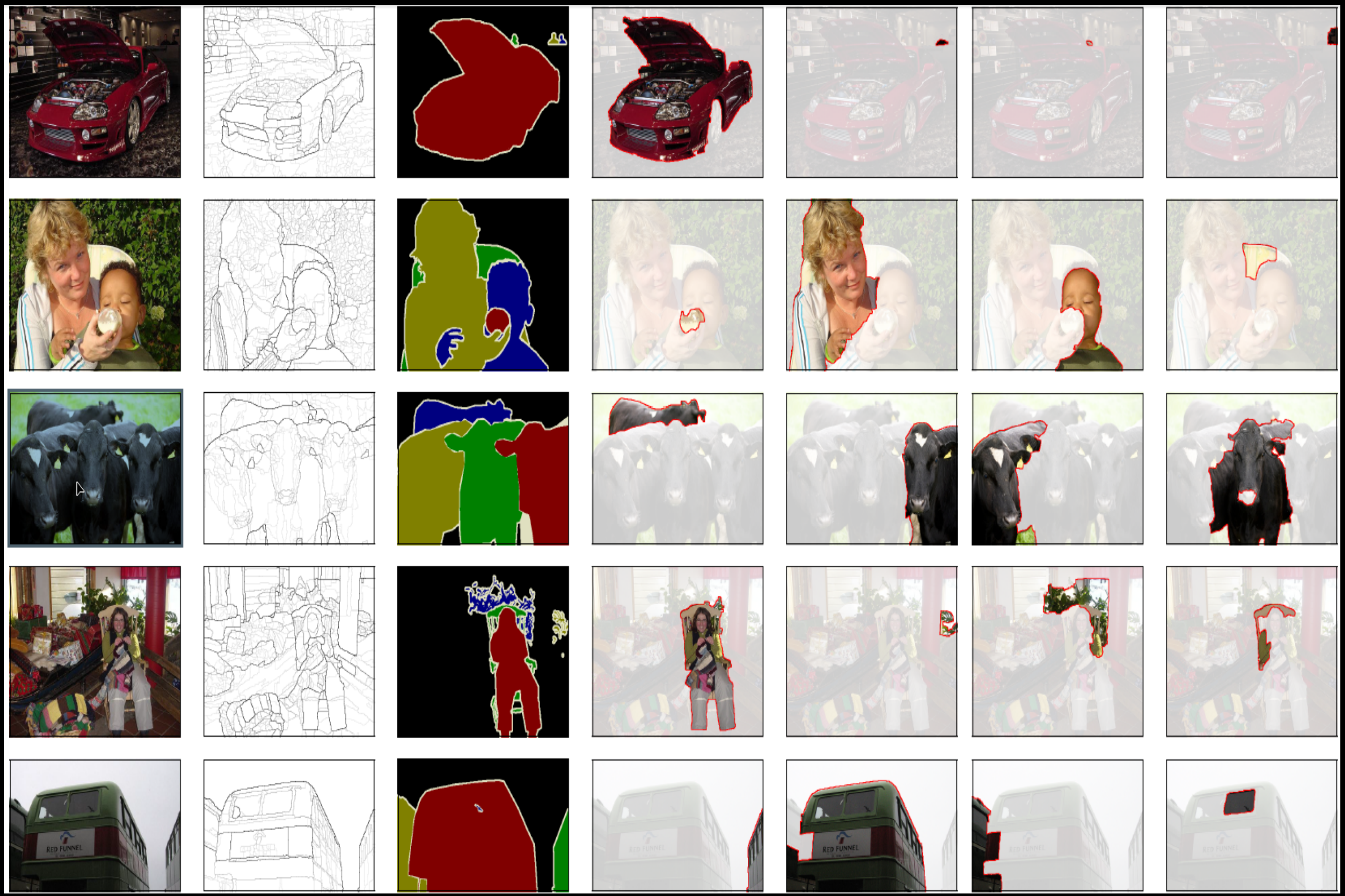


Fig. 5. **Object segmentation as combinatorial optimization**: Examples of objects (b), (c), formed by selecting regions from a hierarchy (a).

Examples



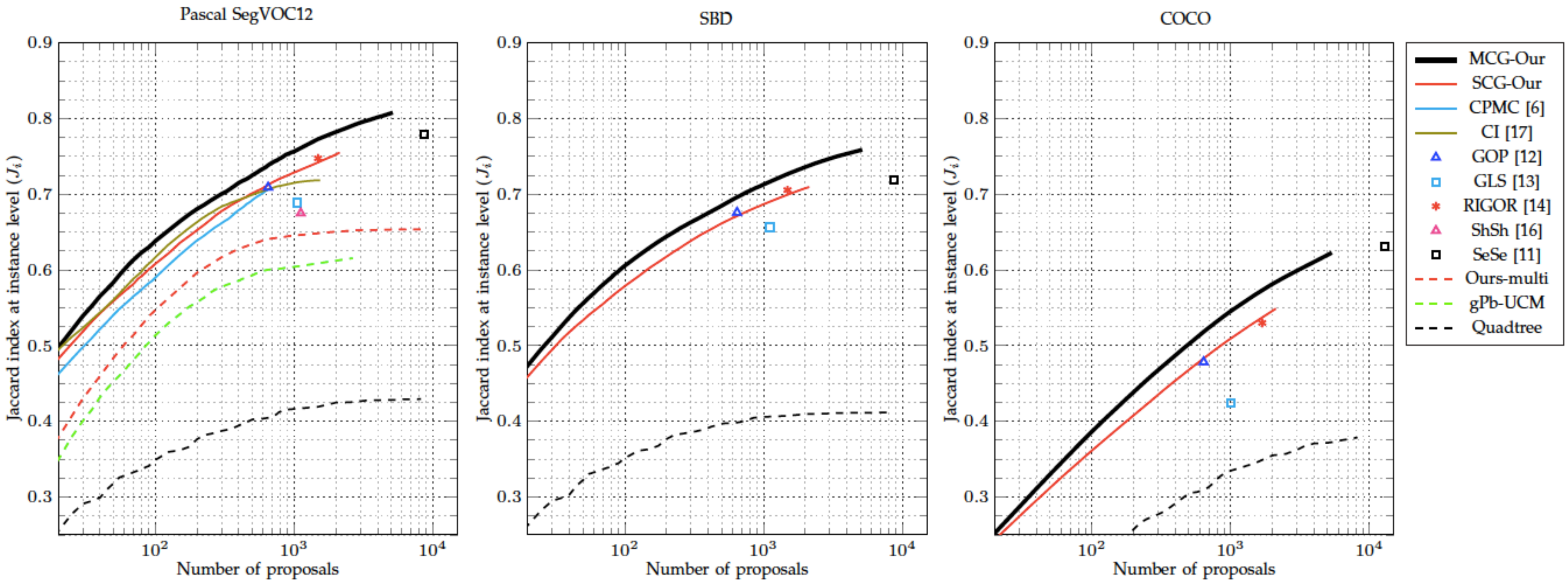


Fig. 8. **Object Proposals: Jaccard index at instance level.** Results on SegVOC12, SBD, and COCO.

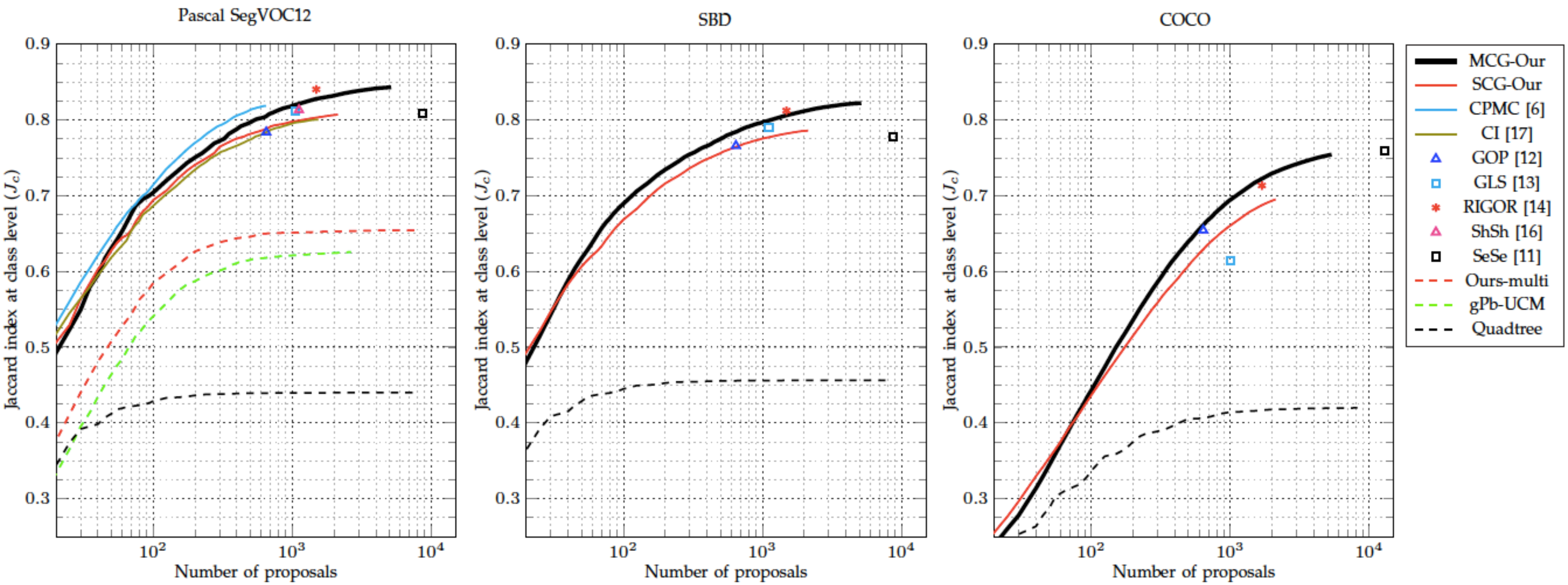


Fig. 9. **Object Proposals: Jaccard index at class level.** Results on SegVOC12, SBD, and COCO.