Computational Vision

Primary visual cortex

- Color opponency
- Coding perspective
- Next week:
    - Learning invariances

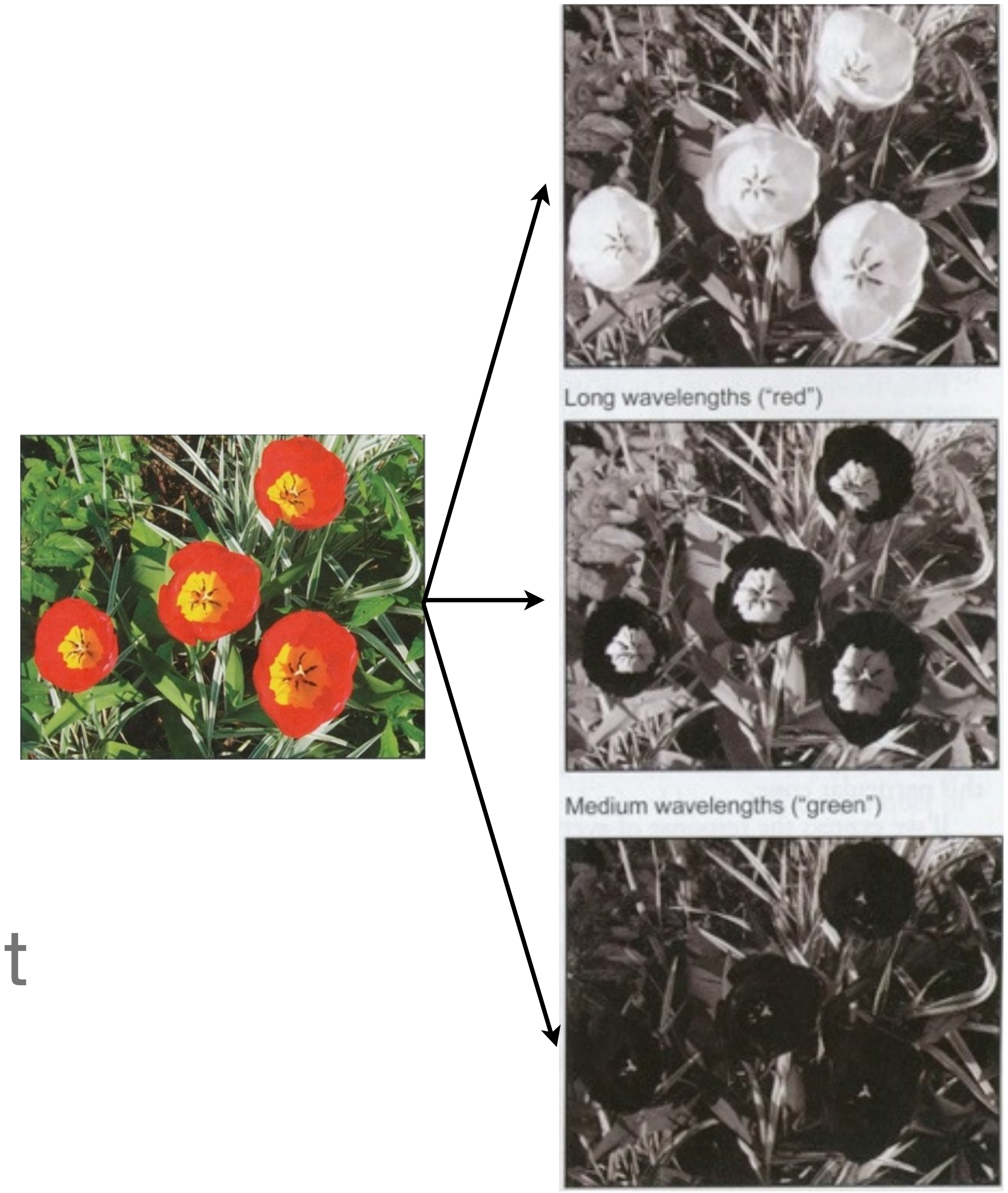
Computational Vision

Primary visual cortex

- Color opponency
- Coding perspective

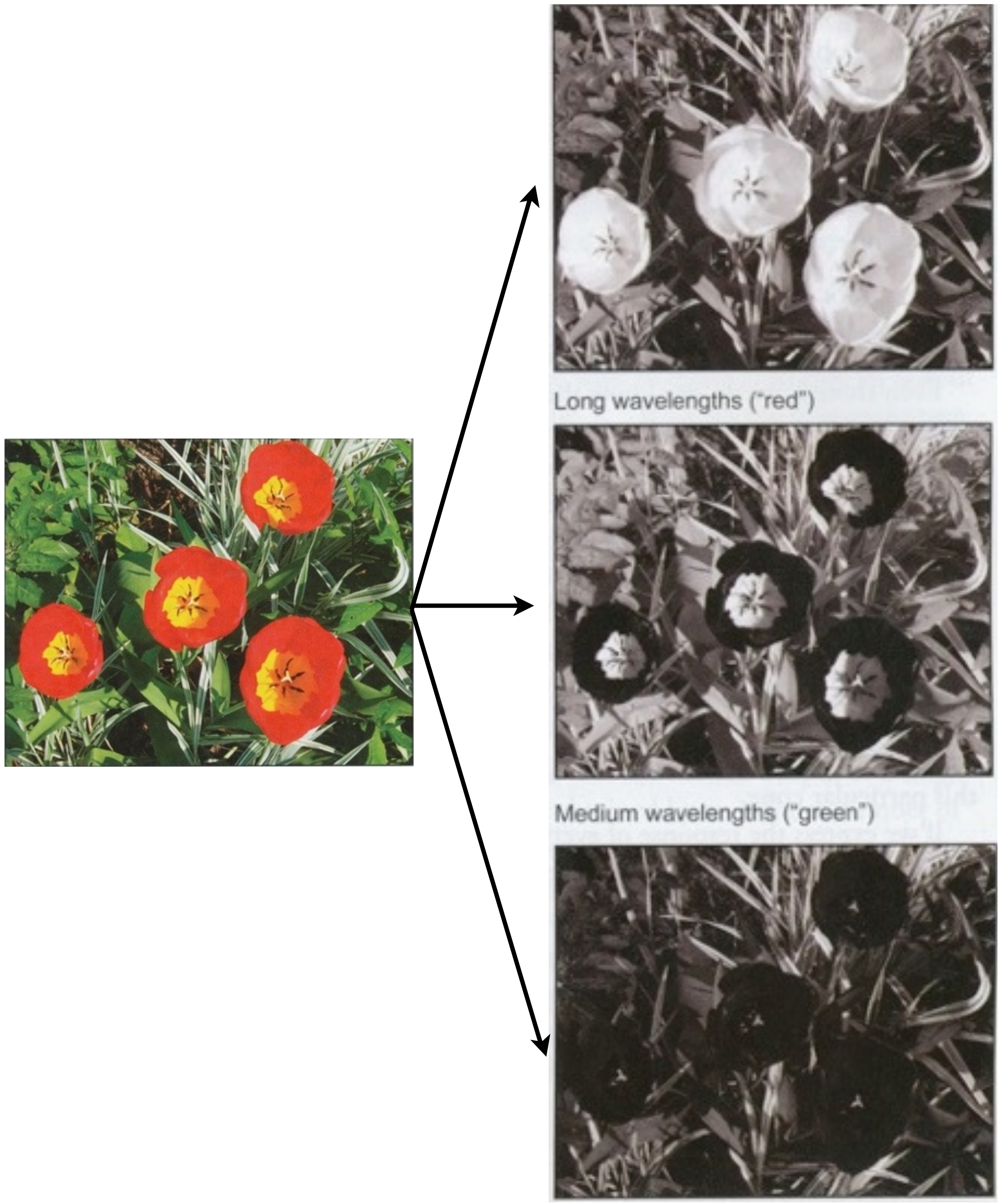
Gray-world assumption

- Given image with sufficient color variations, average of RGB components should be close to common gray value
- True for variations in color that are random and independent
- Given a large enough amount of samples, the average should tend to converge to the mean value (which is gray)



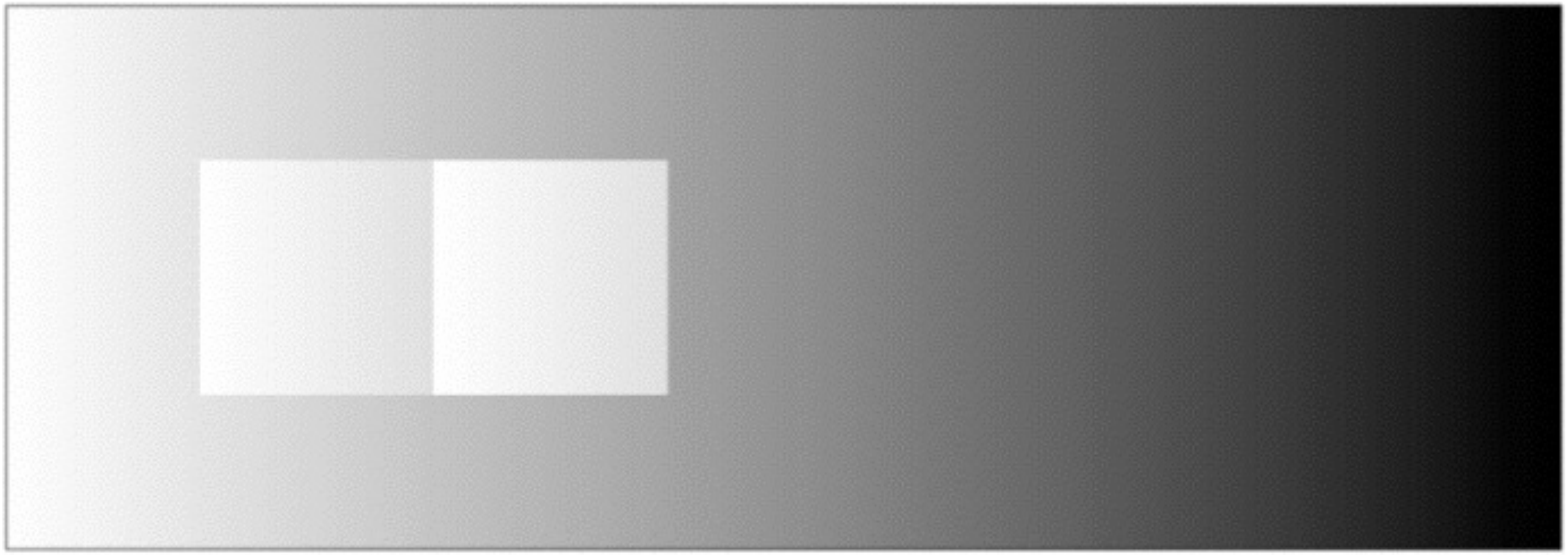
White-world assumption

- Brightest patch is white

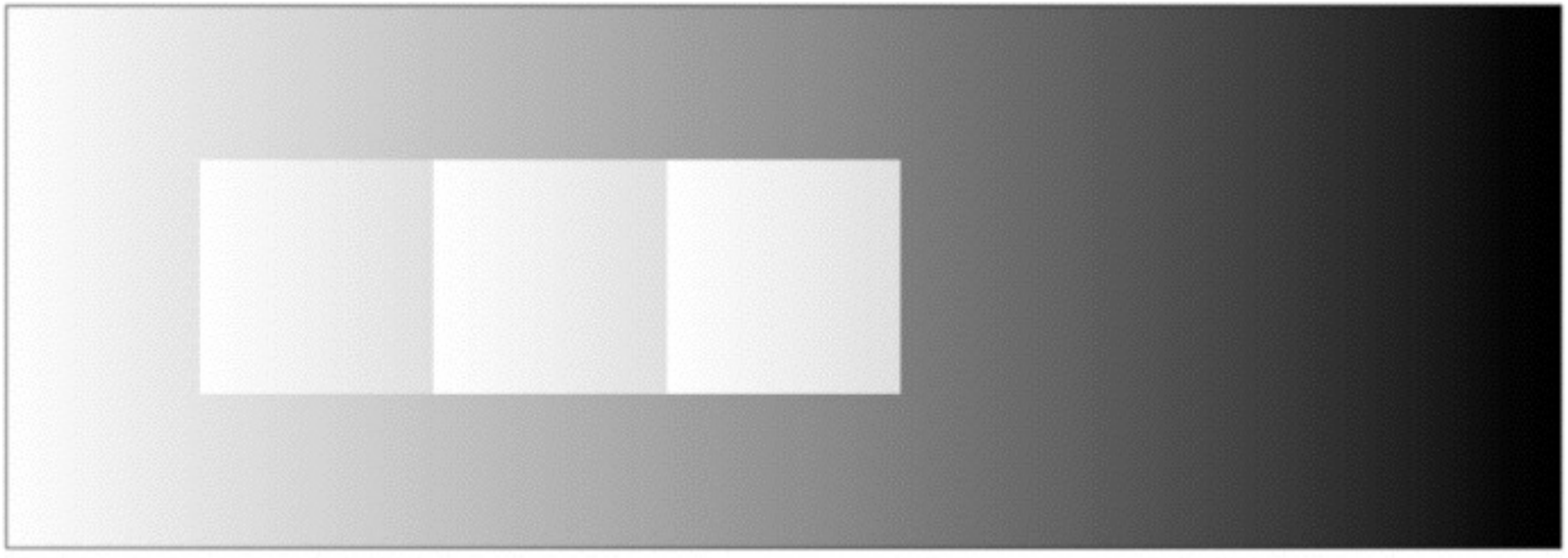


Gelb / Gilchrist demo

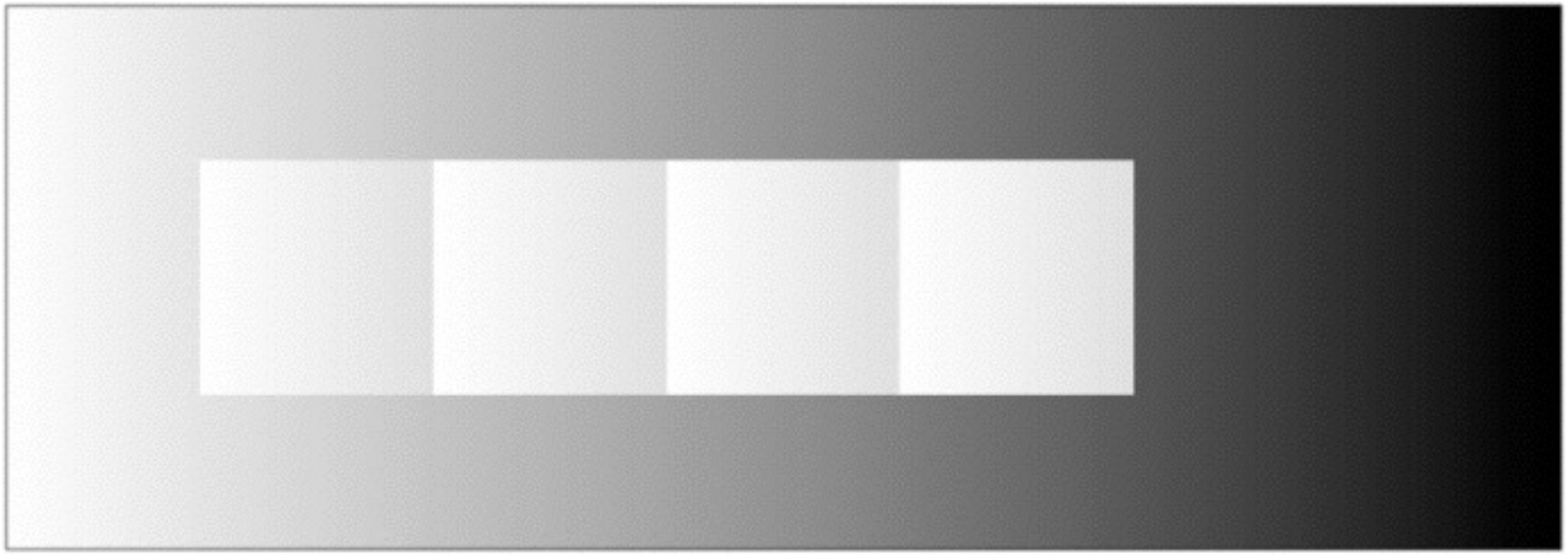
Gelb / Gilchrist demo



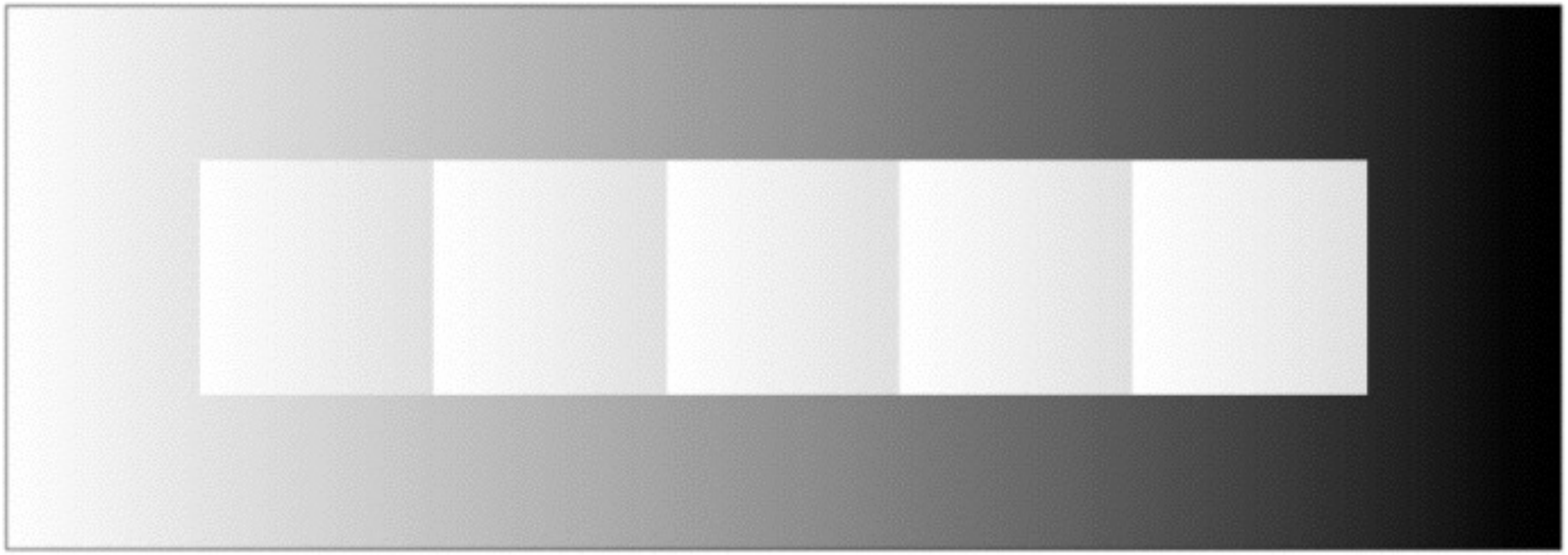
Gelb / Gilchrist demo



Gelb / Gilchrist demo

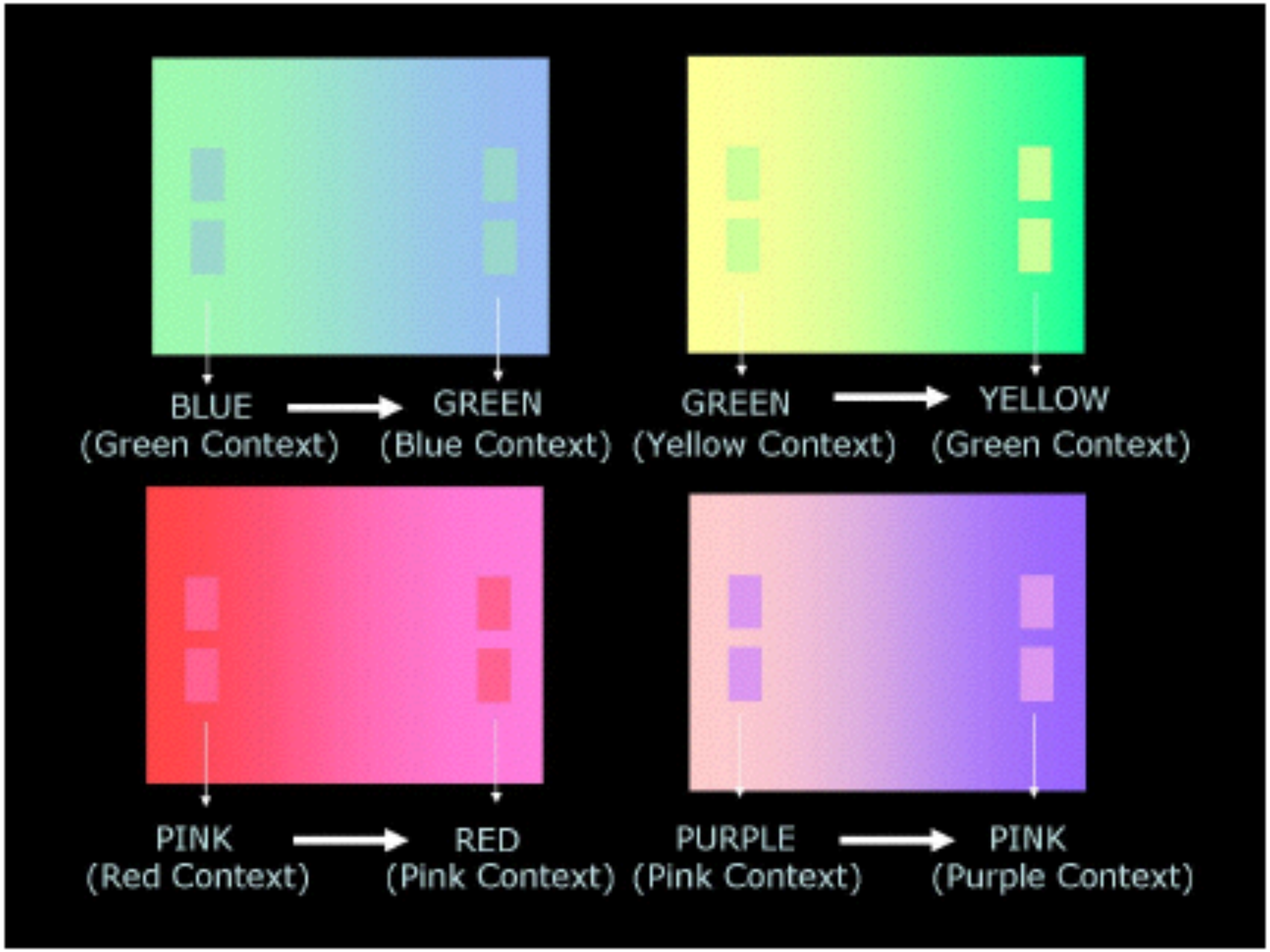


Gelb / Gilchrist demo

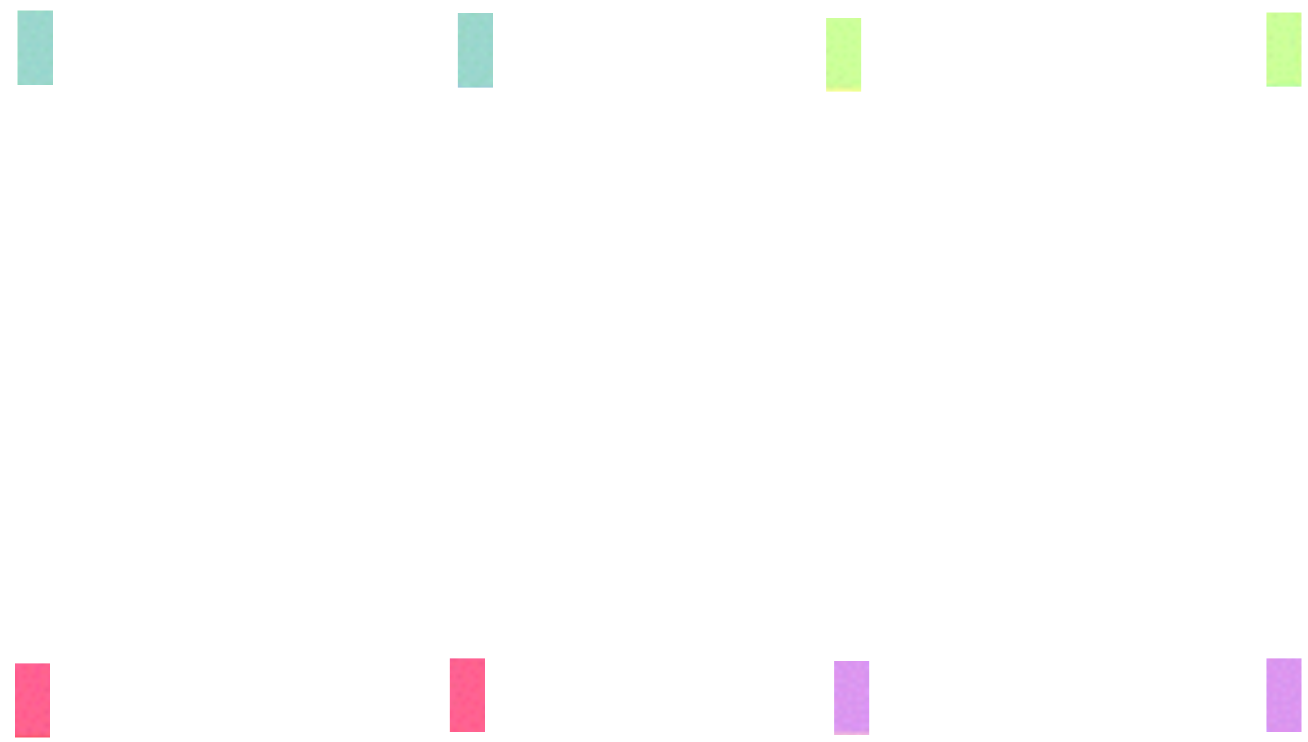


Gelb / Gilchrist demo

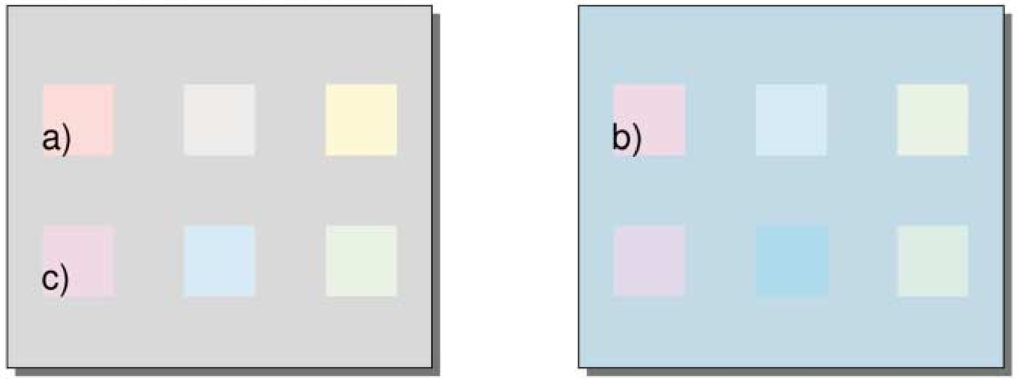
Color induction

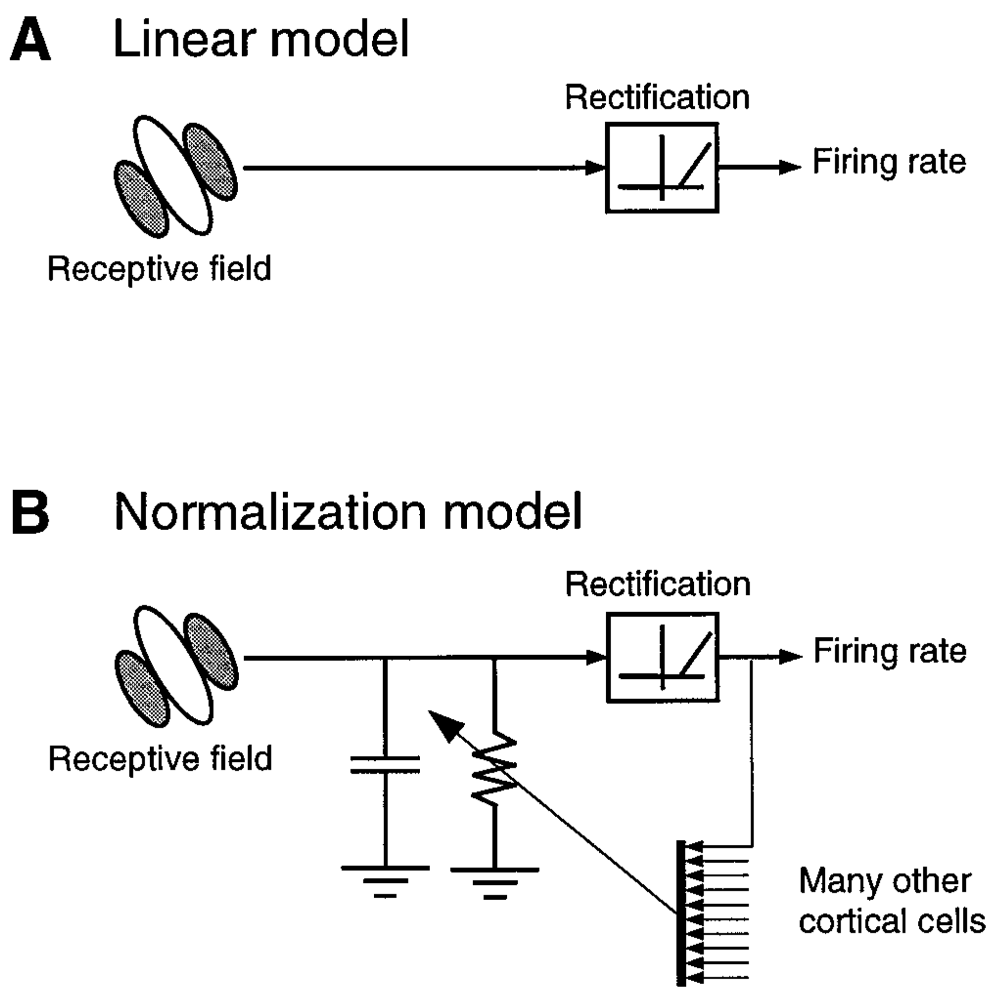


Color induction

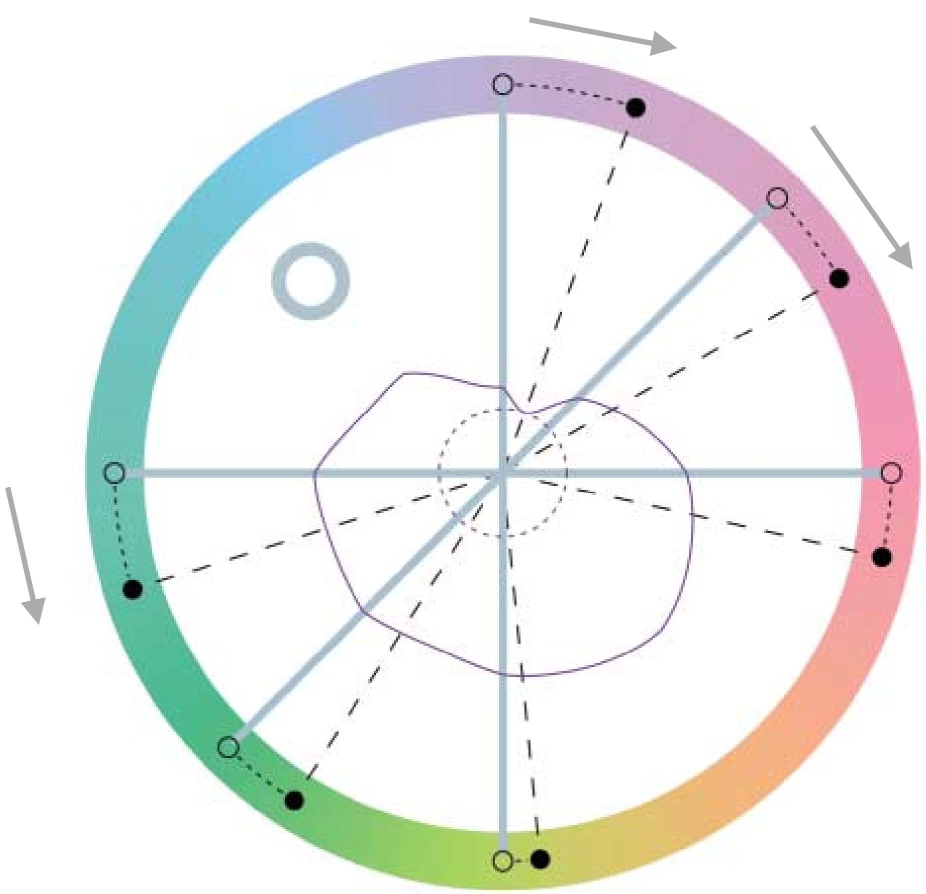


Color induction





Other color channels



Wachtler et al 2003

Zhang & Serre in prep

Computational Vision

Primary visual cortex

- Color opponency
- Coding perspective

What is coding?

- Let $\mathbf{x} = [3, 3, 4, 4, 5, 5]^T$ be a vector in $\mathbb{R}^6$
- Can be represented as linear combination in the standard basis as

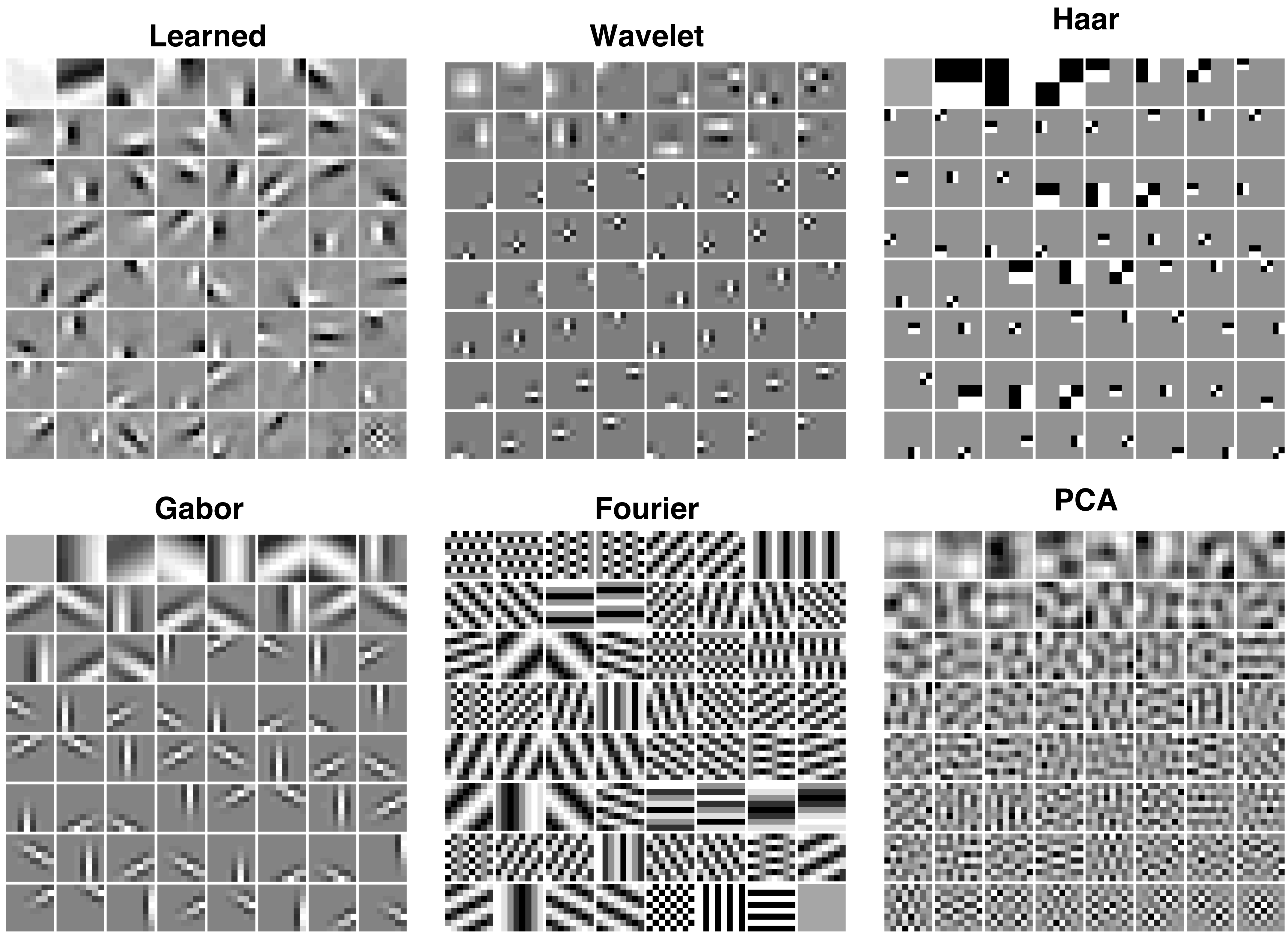
$$\mathbf{x} = 3 \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + 3 \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + 4 \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + 4 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + 5 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} + 5 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Linear code

- where $\mathbf{I} \in \mathbb{R}^{6\times 6}$ is the identity matrix, also called standard basis

Many possible basis for images

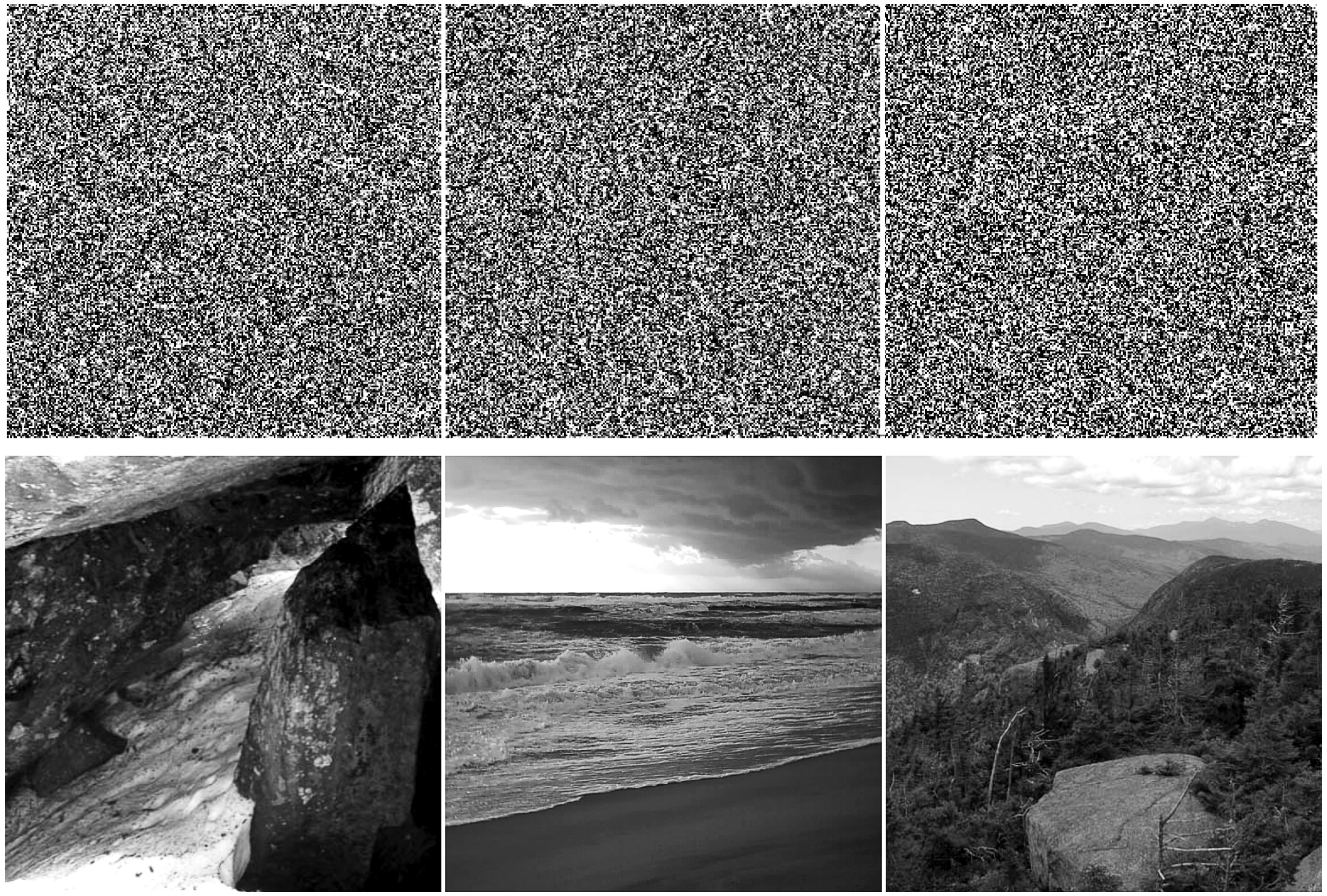
Olshausen & Lewicki 1999



Coding and image statistics

- Why center-surround in LGN or Gabor functions in V1?
- JJ Gibson on the need to understand the visual environment in order to understand visual processing

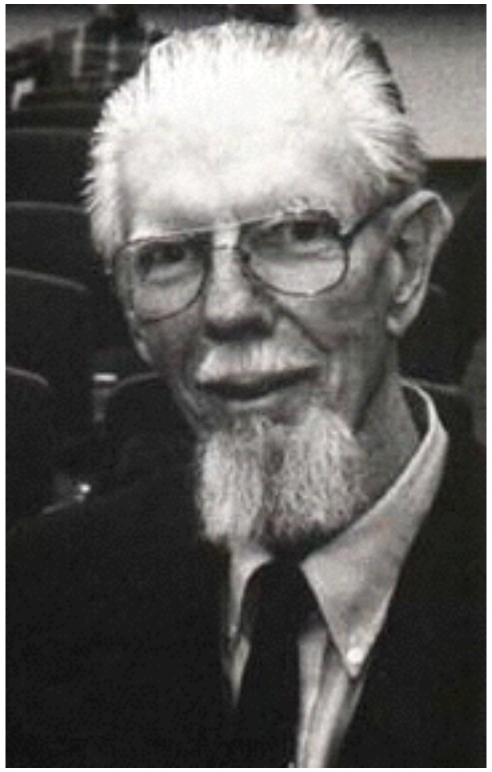
JJ Gibson (1904–1979)



Coding and image statistics

- Efficient coding:
  - Represent most relevant visual information with the fewest physical and metabolic resources
- Redundancy Reduction:
  - Attneave (1954): Some Informational Aspects of Visual Perception
  - Barlow (1961) Possible Principles Underlying the Transformations of Sensory Messages
    - nervous system should reduce redundancy
    - makes more efficient use of neural resources

F Attneave (1919–1991)

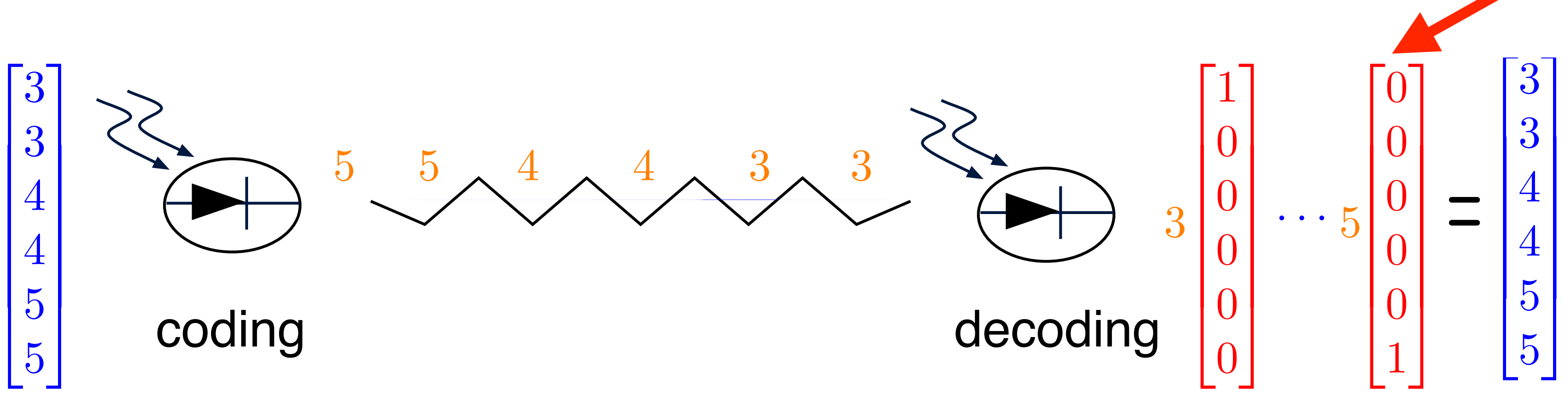


H Barlow (1921–

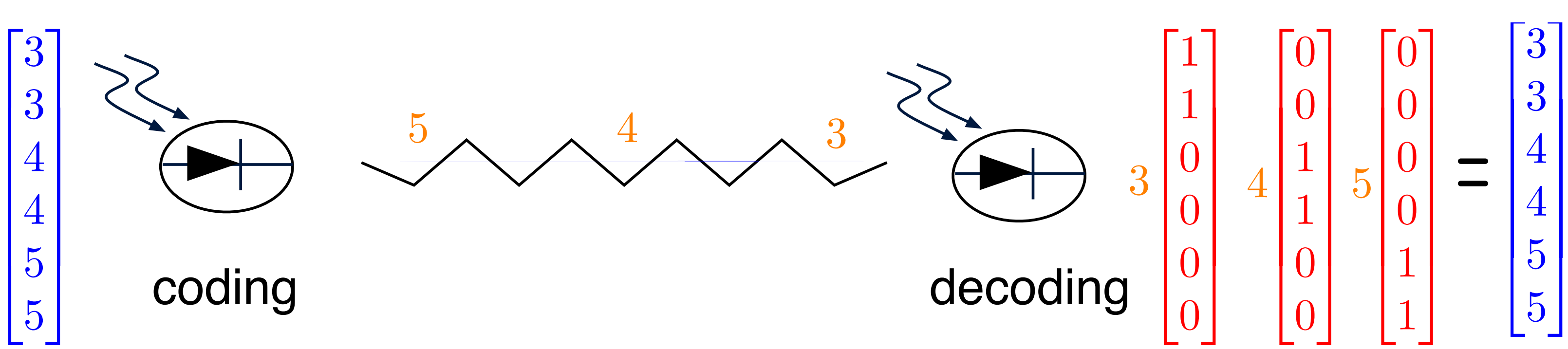
What is coding?

Neuron response

- Not so efficient code…



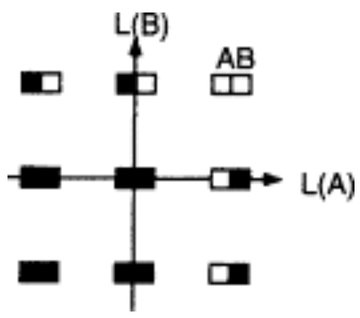
- A more efficient code…



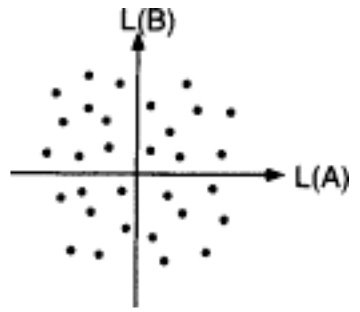
Coding and image statistics

- Natural images are not random
  - Exhibit specific properties that deviate from random processes

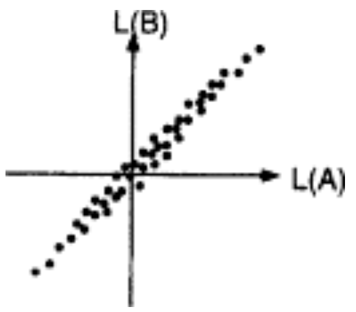
State-space of representable images in a two-pixel array



Random process



Structured distribution



Coding and image statistics

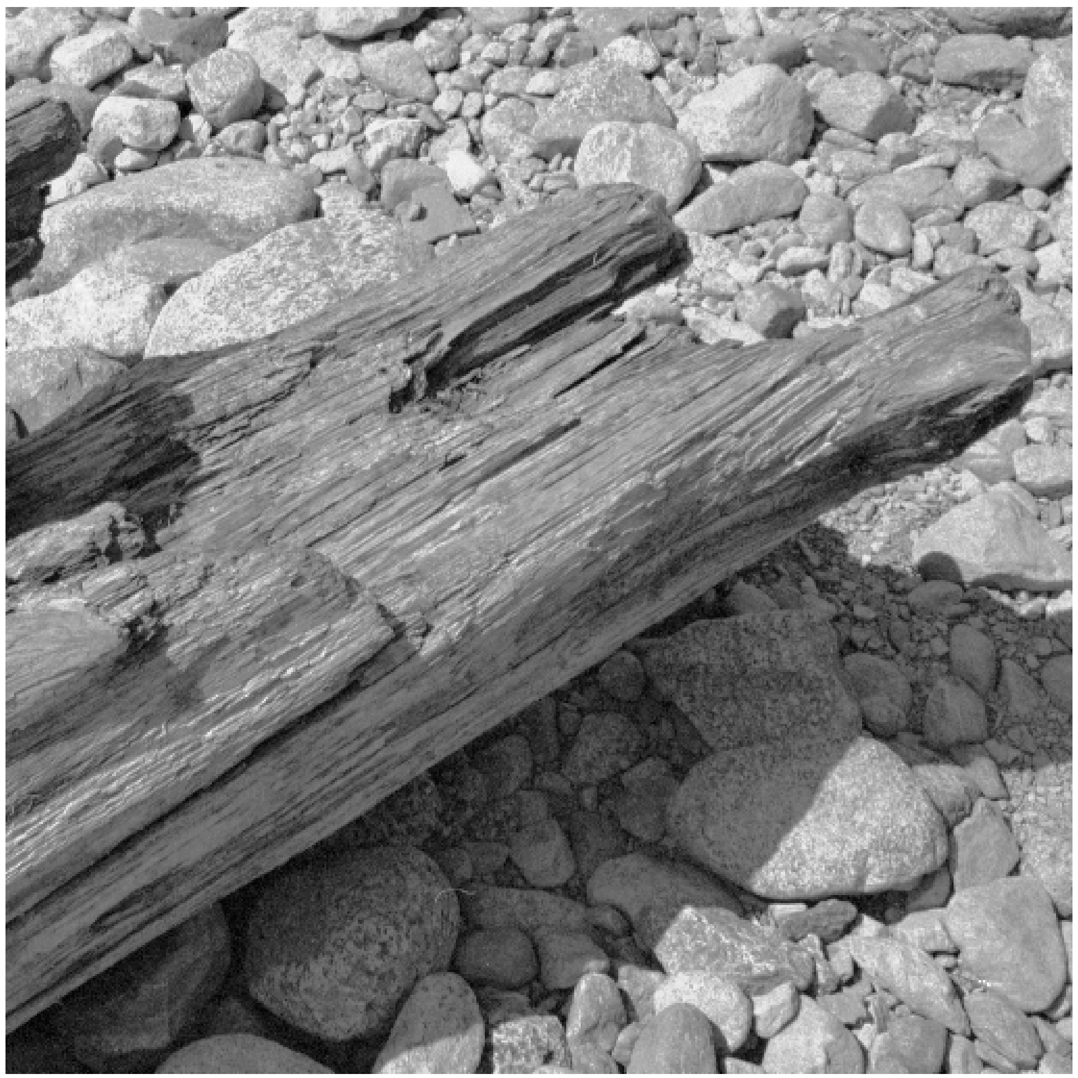
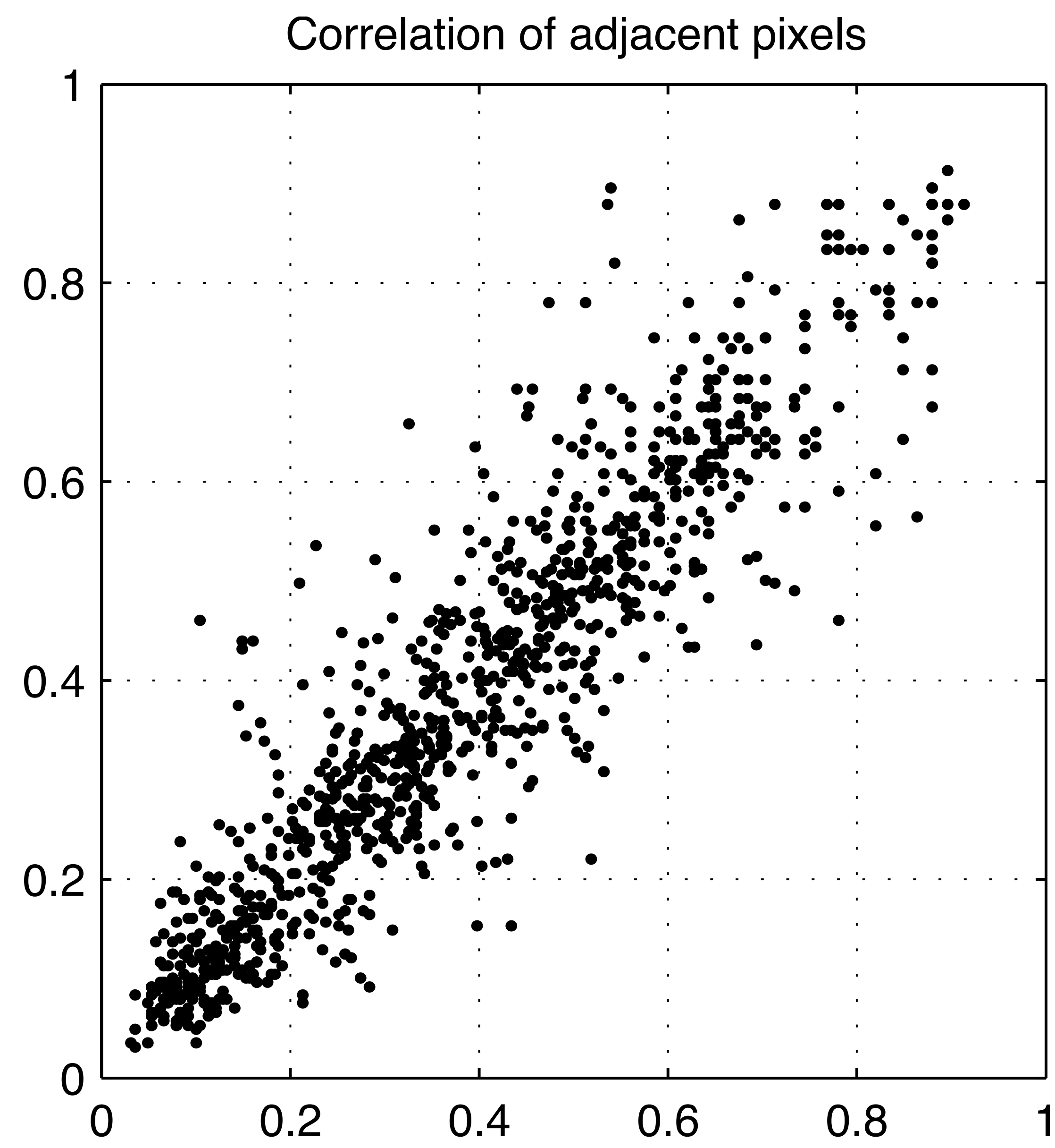


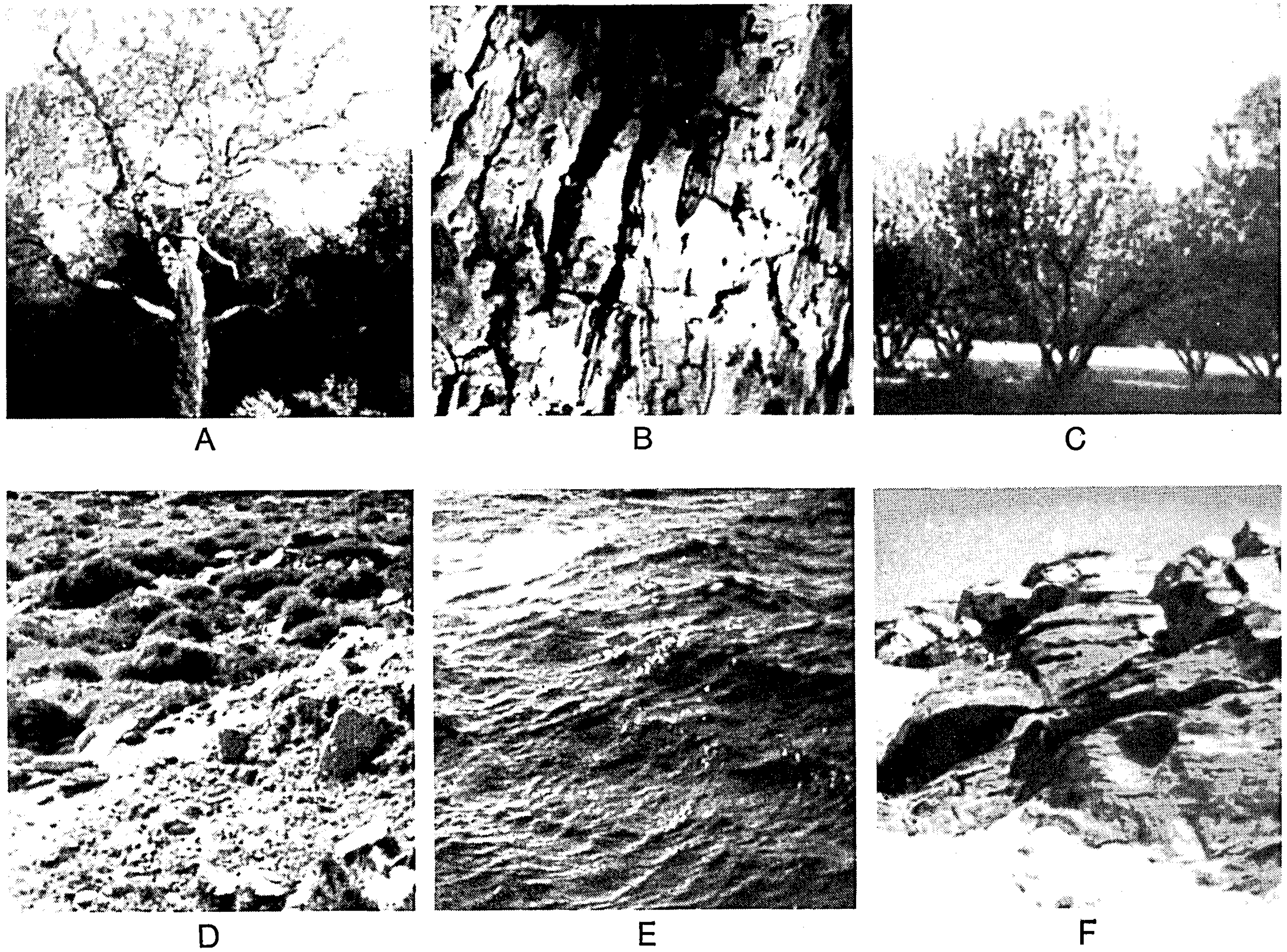
image from Field (1994)



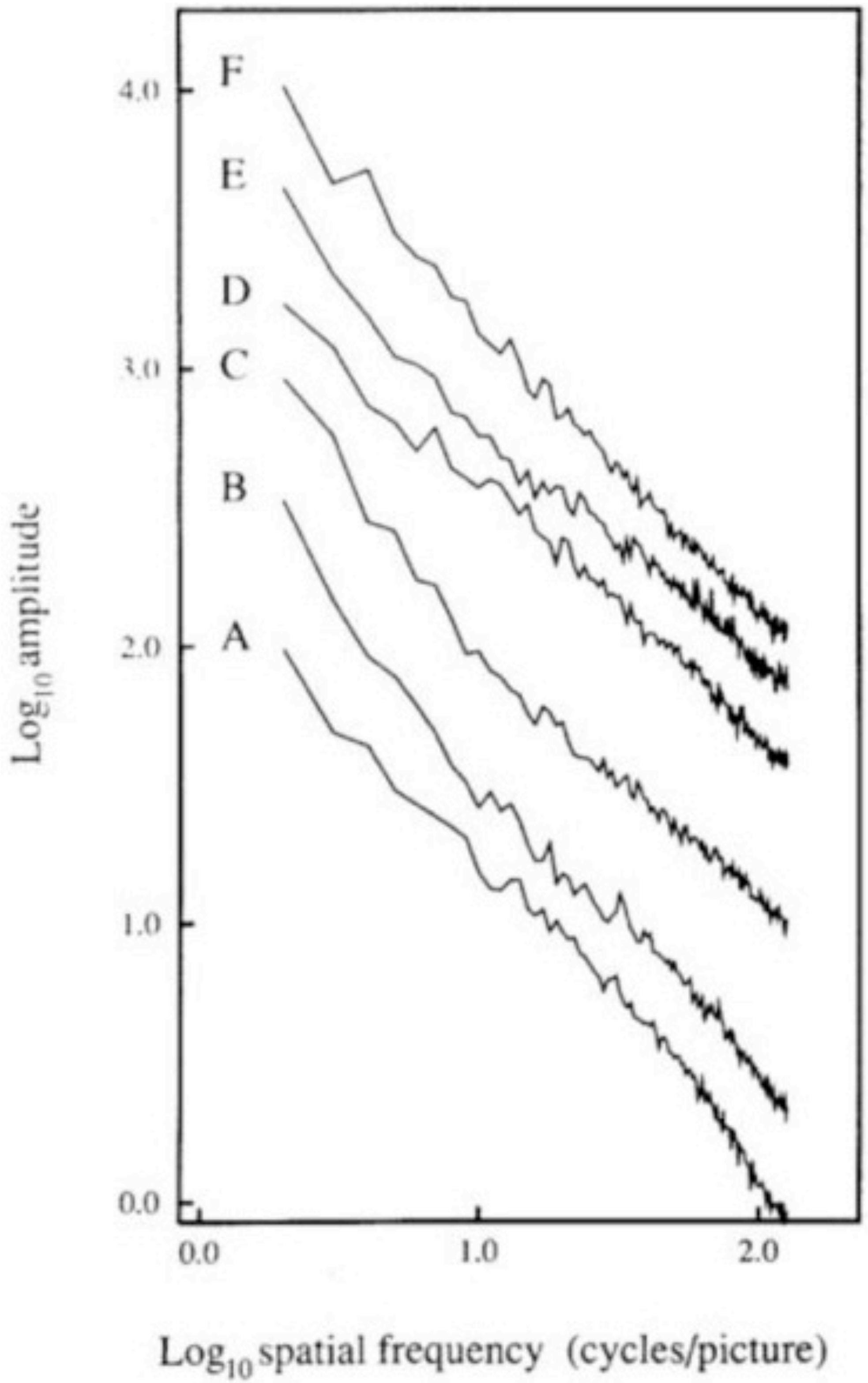
source: Lewicki

Coding and image statistics

source: Olshausen

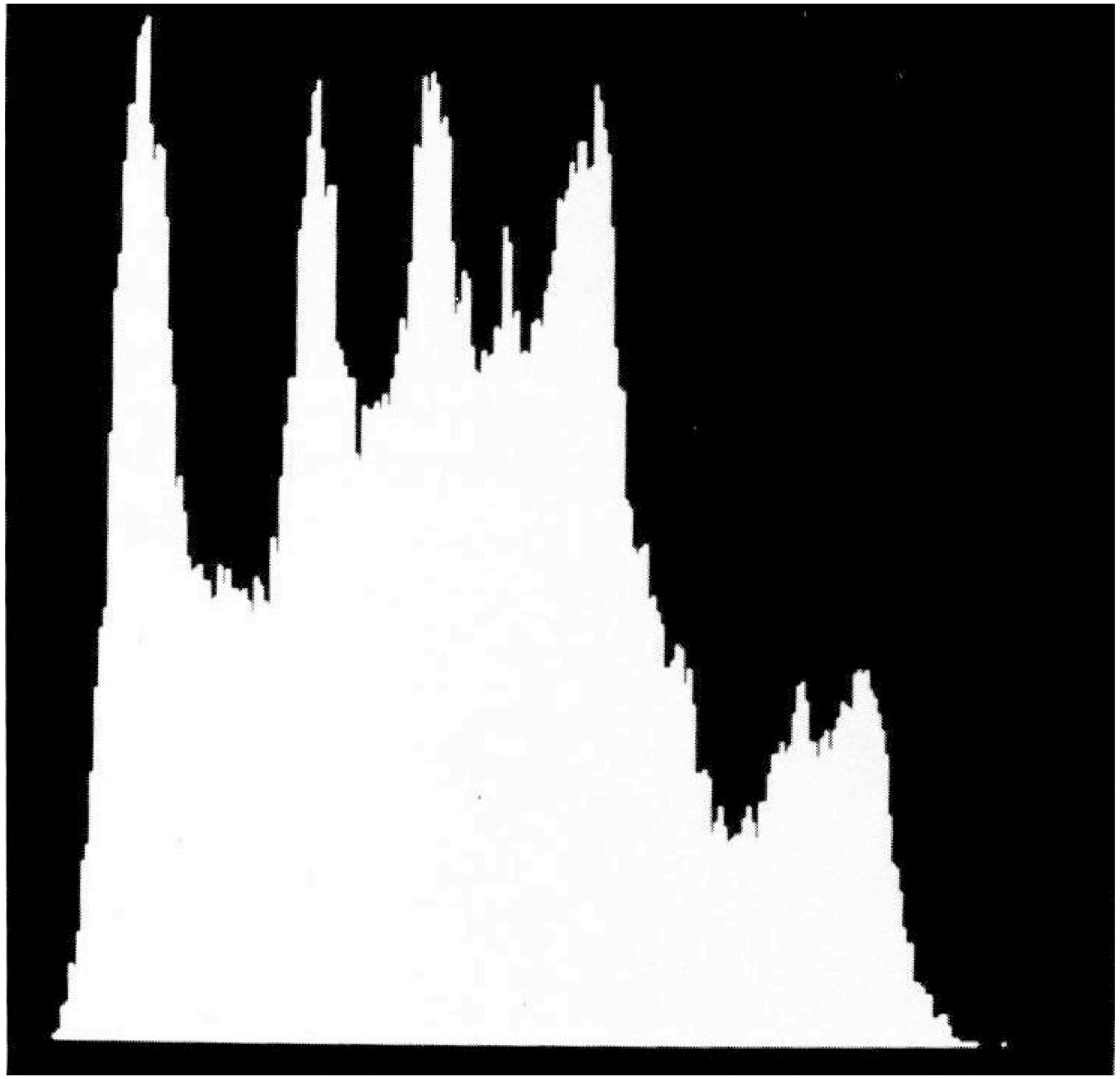


(Field 1987)



Coding and image statistics

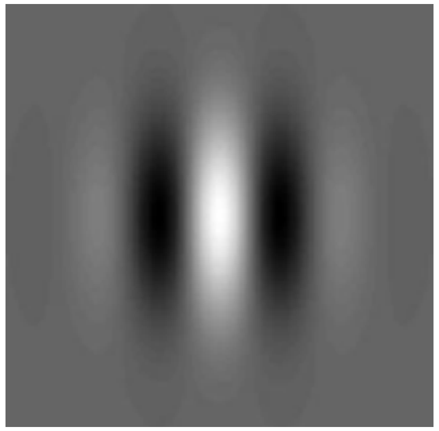
Lena: a standard 8 bit 256x256 gray scale image

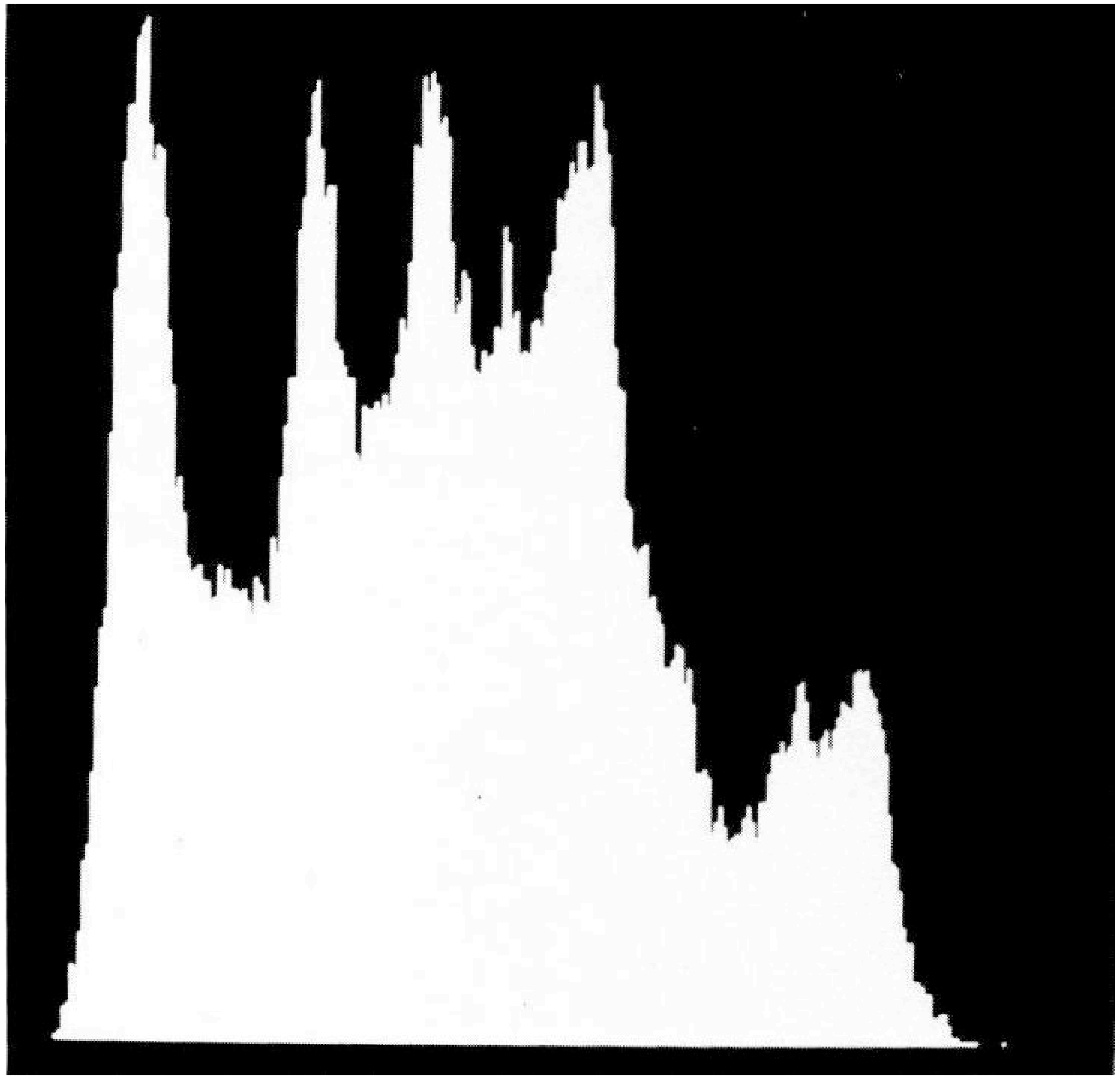


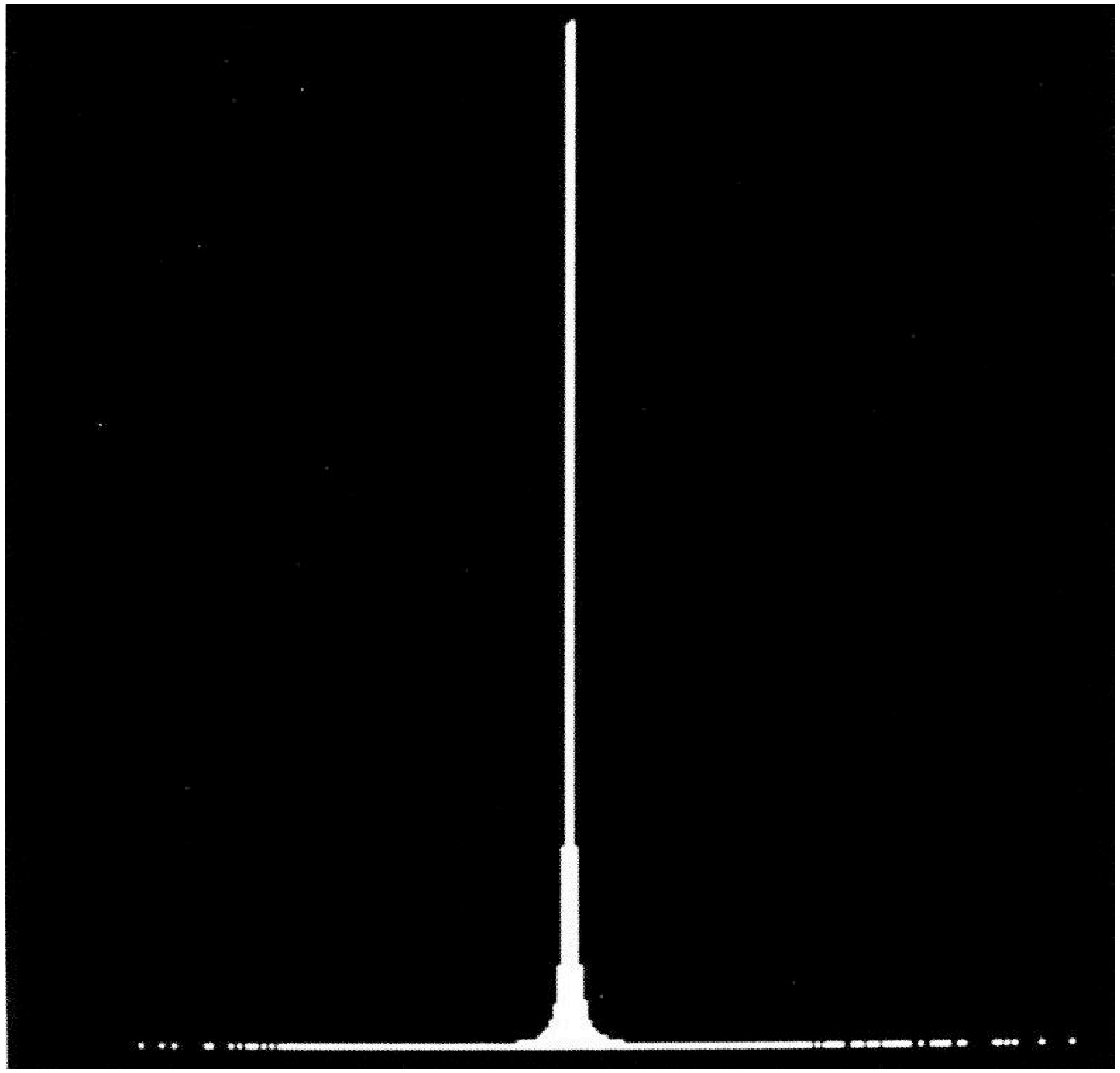
histogram of pixel values
Entropy = 7.57 bits

source: Lewicki

Coding and image statistics







Pixel entropy = 7.57 bits

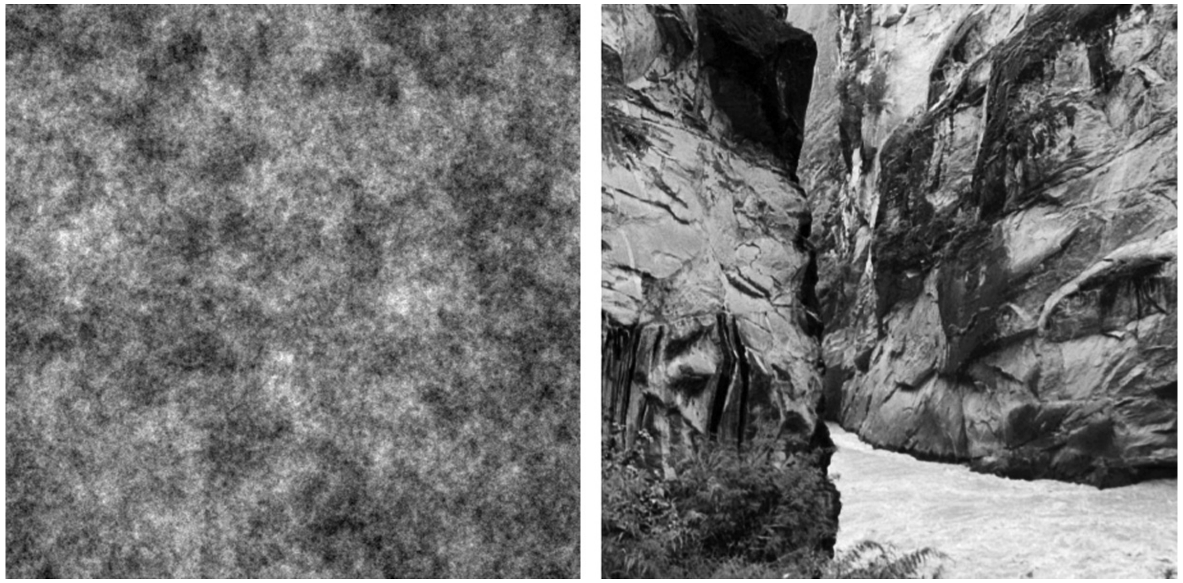
Recoding with 2D Gabor functions
Coefficient entropy = 2.55 bits

source: Lewicki

Beyond efficient coding

- RR is appropriate when there is a bottleneck. But V1 expands dimensionality - many more neurons than inputs
- The real goal of sensory representation is to model the redundancy in images, not necessarily to reduce it (Barlow 2001)

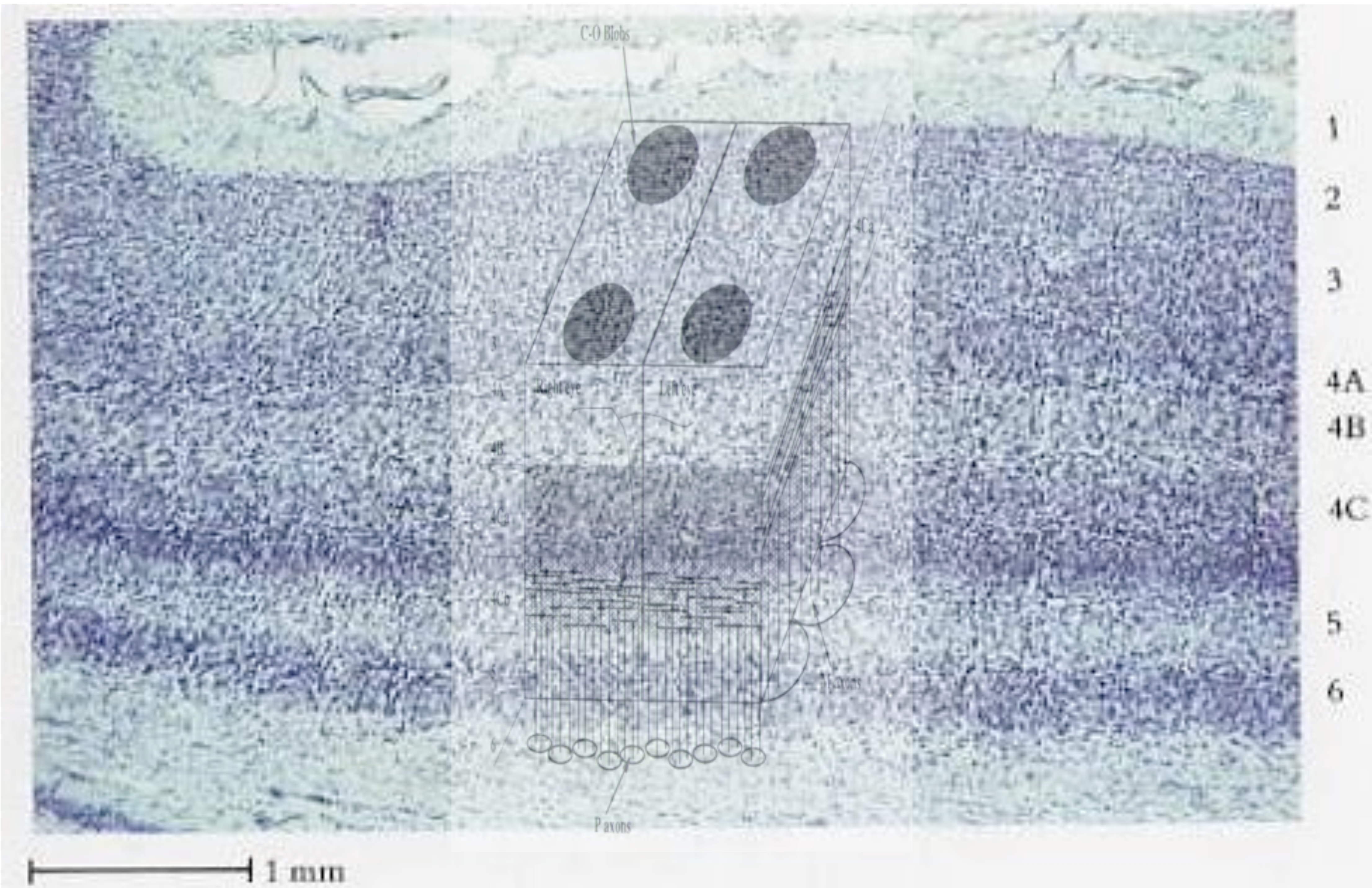
Same pairwise correlations but noise image lacks other statistical regularities of scenes…



Graham & Field 2009

Beyond efficient coding

1 mm^2 of cortex analyzes ca. 14 x 14 array of retinal sample nodes and contains 100,000 neurons!

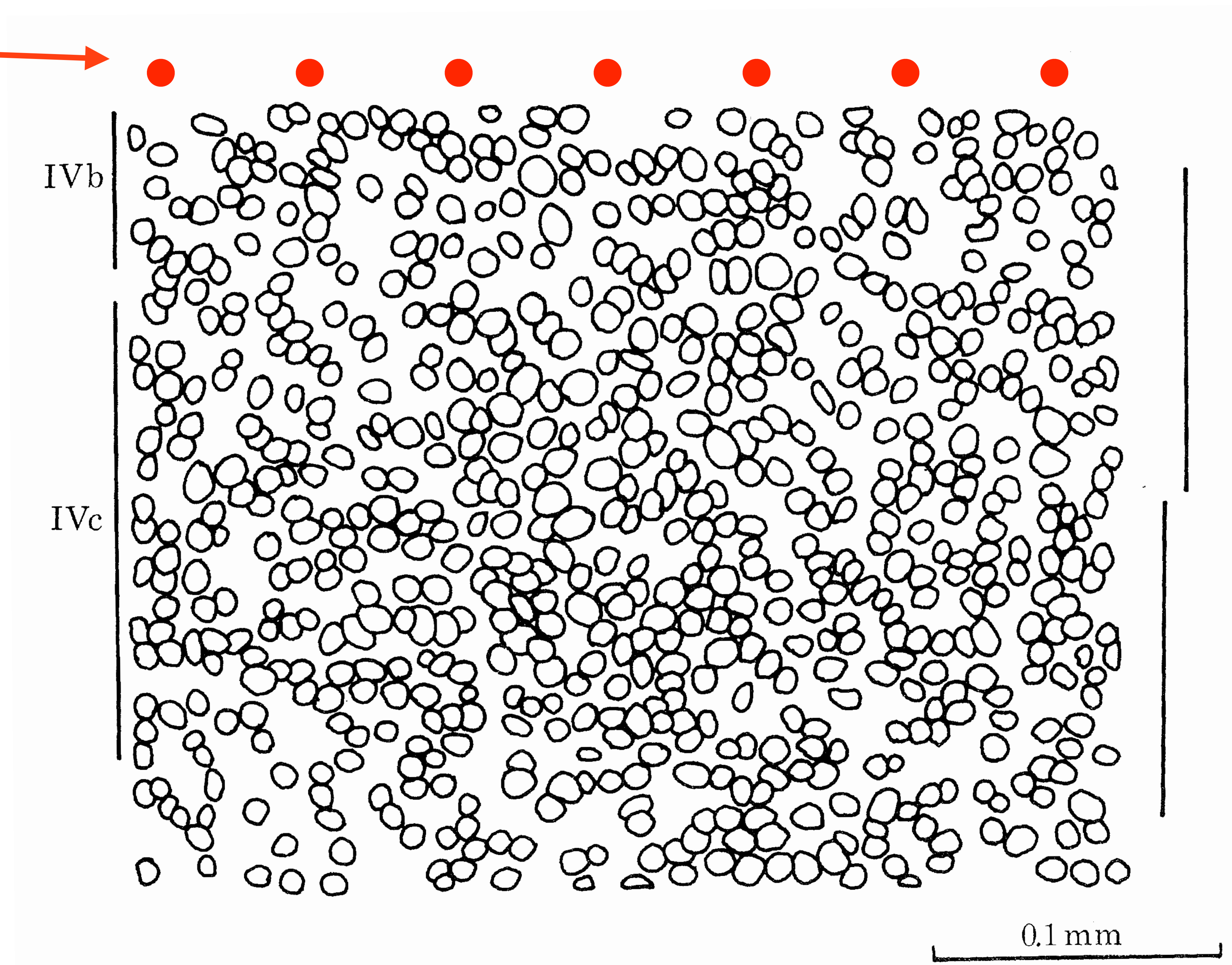


source: Olshausen

Beyond efficient coding

LGN
afferents

layer 4
cortex



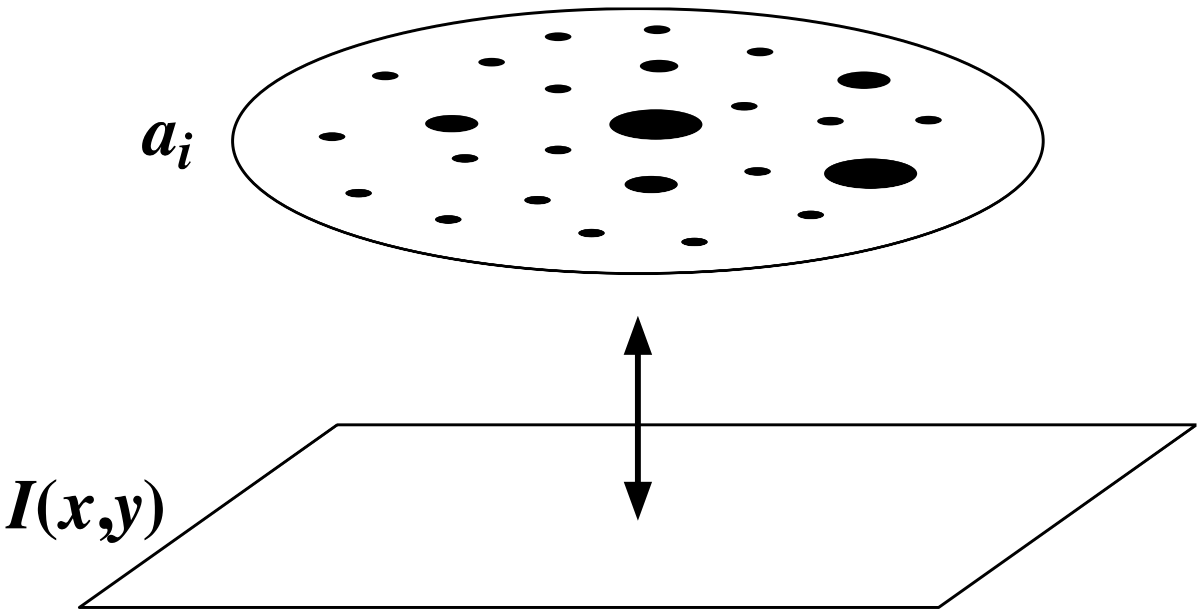
source: Olshausen

Sparse codes

$$3\begin{bmatrix}1\\1\\0\\0\\0\\0\end{bmatrix}\;4\begin{bmatrix}0\\0\\1\\1\\0\\0\end{bmatrix}\;5\begin{bmatrix}0\\0\\0\\0\\1\\1\end{bmatrix} = \begin{bmatrix}3\\3\\4\\4\\5\\5\end{bmatrix} \;\Longrightarrow\; \mathbf{0}\begin{bmatrix}?\\?\\?\\?\\?\\?\end{bmatrix}\;3\begin{bmatrix}1\\1\\0\\0\\0\\0\end{bmatrix}\;\mathbf{0}\begin{bmatrix}?\\?\\?\\?\\?\\?\end{bmatrix}\;4\begin{bmatrix}0\\0\\1\\1\\0\\0\end{bmatrix}\quad 4\begin{bmatrix}0\\0\\1\\1\\0\\0\end{bmatrix}\;\mathbf{0}\begin{bmatrix}?\\?\\?\\?\\?\\?\end{bmatrix} = \begin{bmatrix}3\\3\\4\\4\\5\\5\end{bmatrix}$$

Sparse overcomplete codes

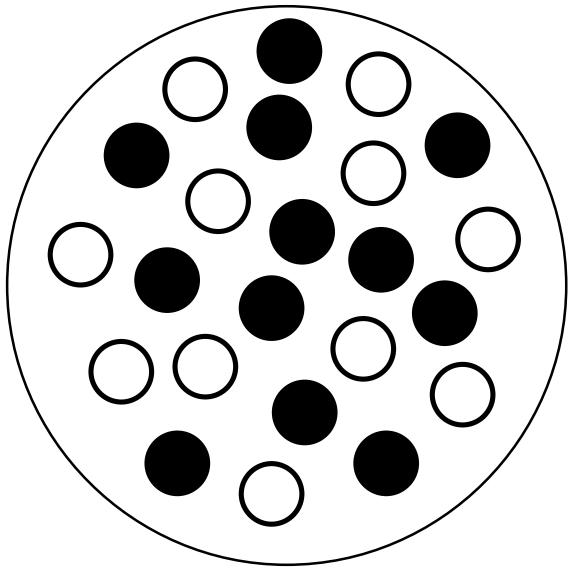
- Provides a way to group things together so that the world can be described in terms of a small number of events at any given moment
- Converts higher-order redundancy in images into a simple form of redundancy



source: Olshausen

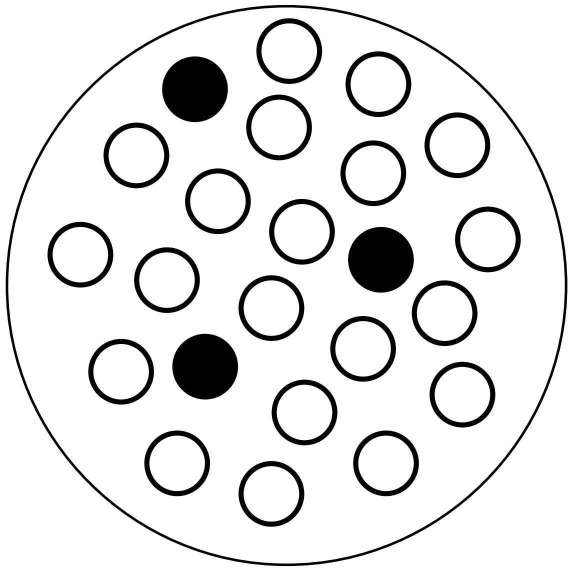
Sparse vs. dense vs. ‘grand-mother cells’ code

Dense codes
(ascii)



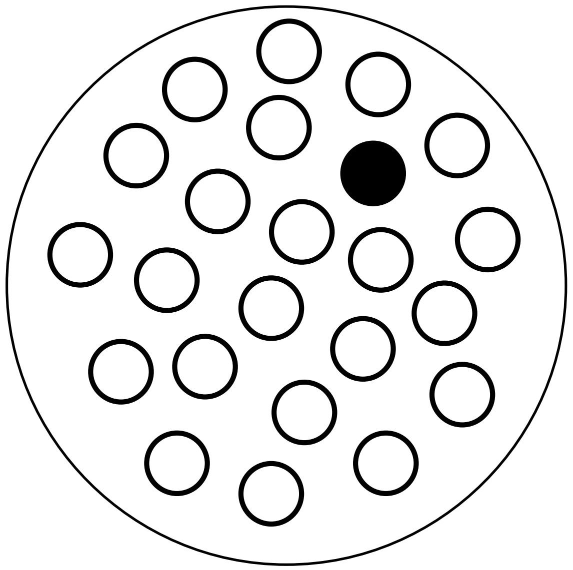
+ High combinatorial capacity (2^N)
- Difficult to read out

Sparse, distributed codes



+ Decent combinatorial capacity ($\sim N^K$)
+ Still easy to read out

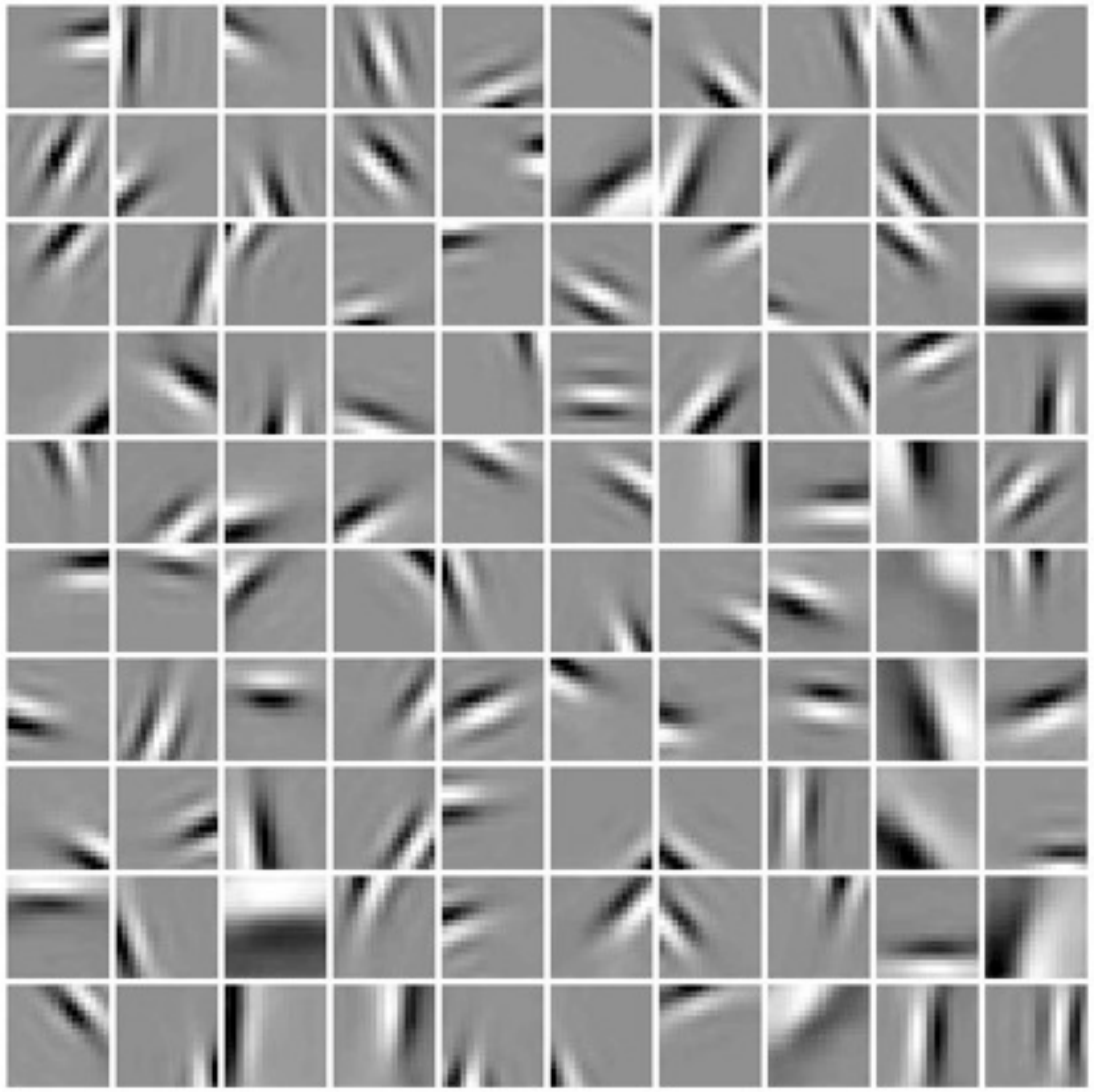
Local codes
(grandmother cells)

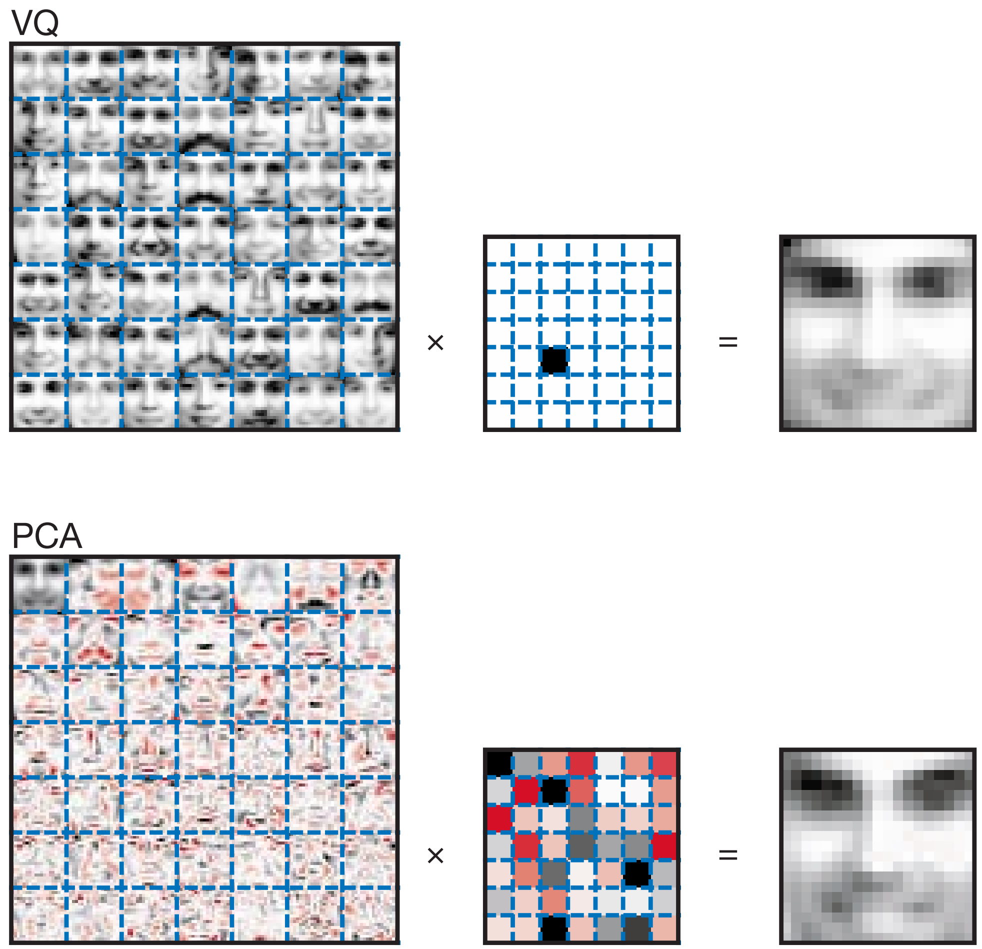


- Low combinatorial capacity (N)
+ Easy to read out

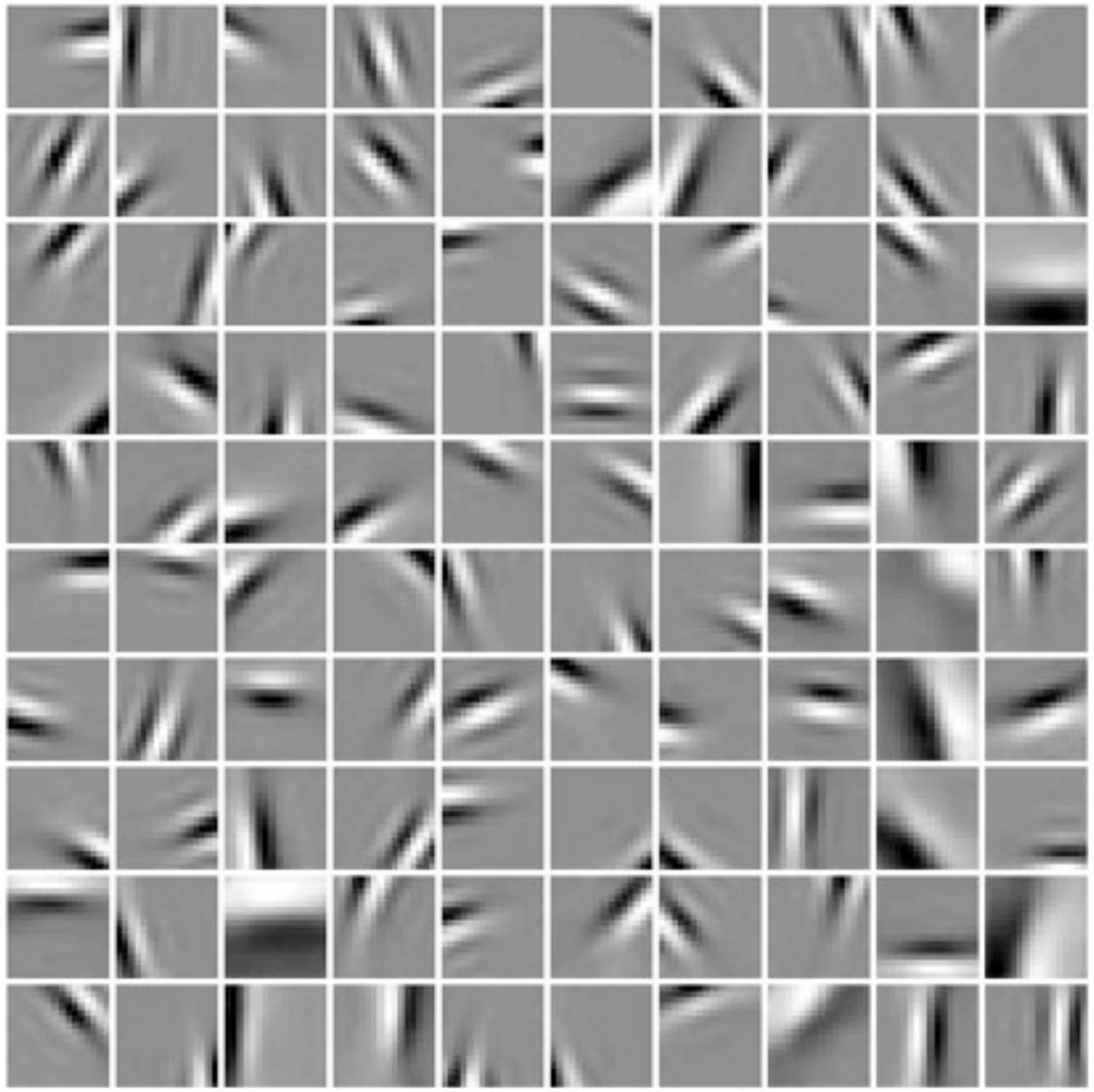
source: Olshausen

Sparse code for natural images





Sparse coding



$$\underbrace{I(x,y)}_{\text{image}} = \sum_i \underbrace{a_i}_{\substack{\text{neural}\\\text{activities}\\\text{(sparse)}}} \underbrace{\phi_i(x,y)}_{\text{features}} + \underbrace{\epsilon(x,y)}_{\substack{\text{other}\\\text{stuff}}}$$

source: Olshausen

Sparse coding

- Usually, dictionary $\mathbf{\Phi}$ is overcomplete: Linear system has many solutions…
- To get reasonable solution, additional constraints on coefficients $\mathbf{a}$ are needed

$$I(x,y) = \sum_i a_i \phi_i(x,y) + \epsilon(x,y)$$

image

neural activities (sparse)

features

other stuff

Sparse coding

- With sparsity constraint, sparse solution can be obtained:

$$\mathbf{a}^* = \arg\min_{\mathbf{a}} \underbrace{\|\mathbf{x} - \Phi\mathbf{a}\|_2}_{\text{reconstruction error}} + \underbrace{\lambda\|\mathbf{a}\|_1}_{\text{sparsity}}$$

- This is a convex optimization problem and has many solvers

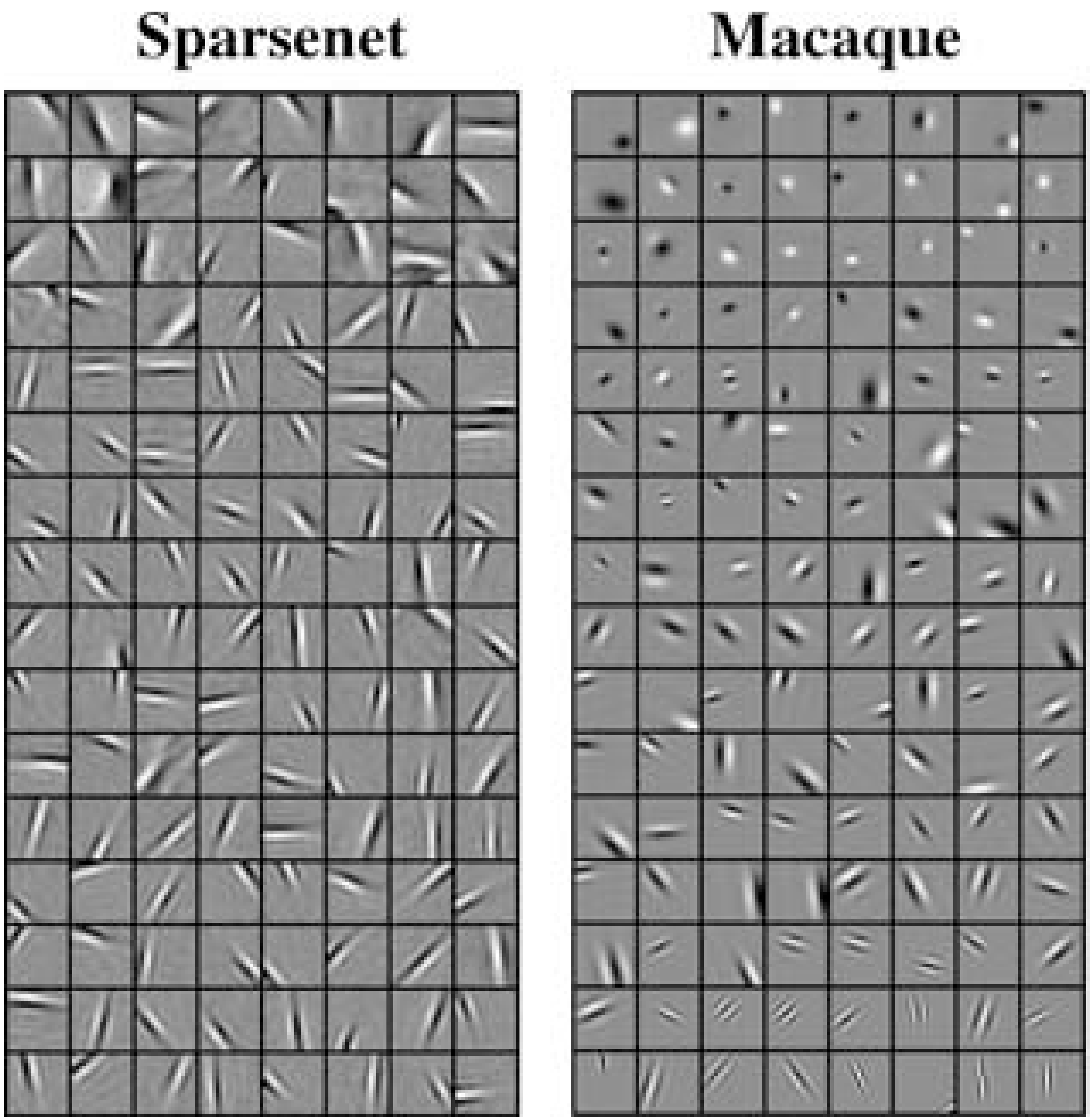
Sparse coding

- Given samples, $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, how to learn a set of basis functions that are capable of sparsely coding all samples

$$< \mathbf{a}^*, \mathbf{\Phi}^* >= \arg\min_{\mathbf{a},\mathbf{\Phi}} \sum_{i=1}^{N} \{\|\mathbf{x}_i - \mathbf{\Phi}\mathbf{a}_i\|_2 + \lambda\|\mathbf{a}_i\|_1\}$$

- A natural approach to solving this problem is to alternate between, $\mathbf{a}$ and $\mathbf{\Phi}$, minimizing over one while keeping the other one fixed

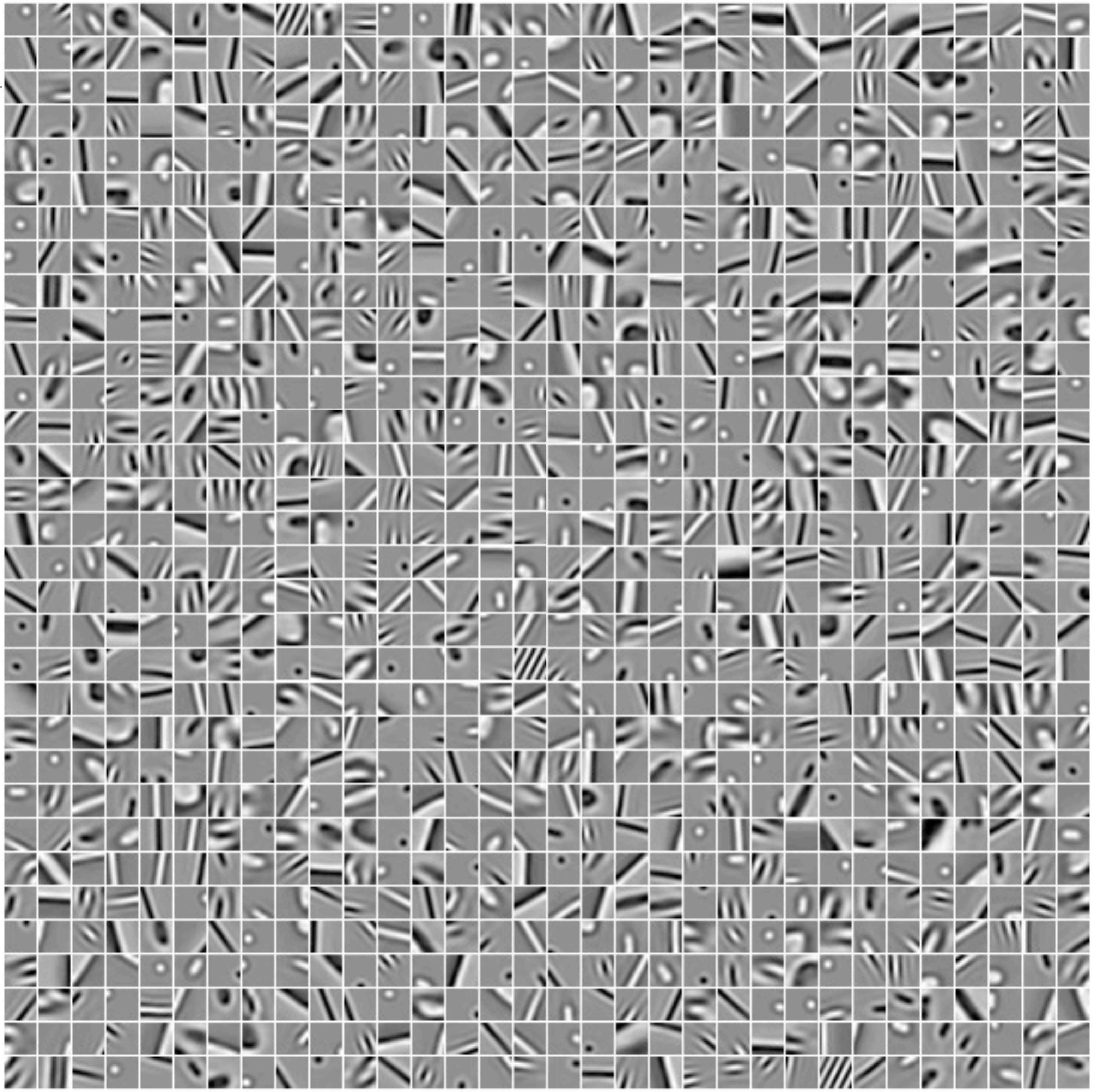
Sparse code for natural images



Rehn and Sommer 2006

Sparse code for natural images

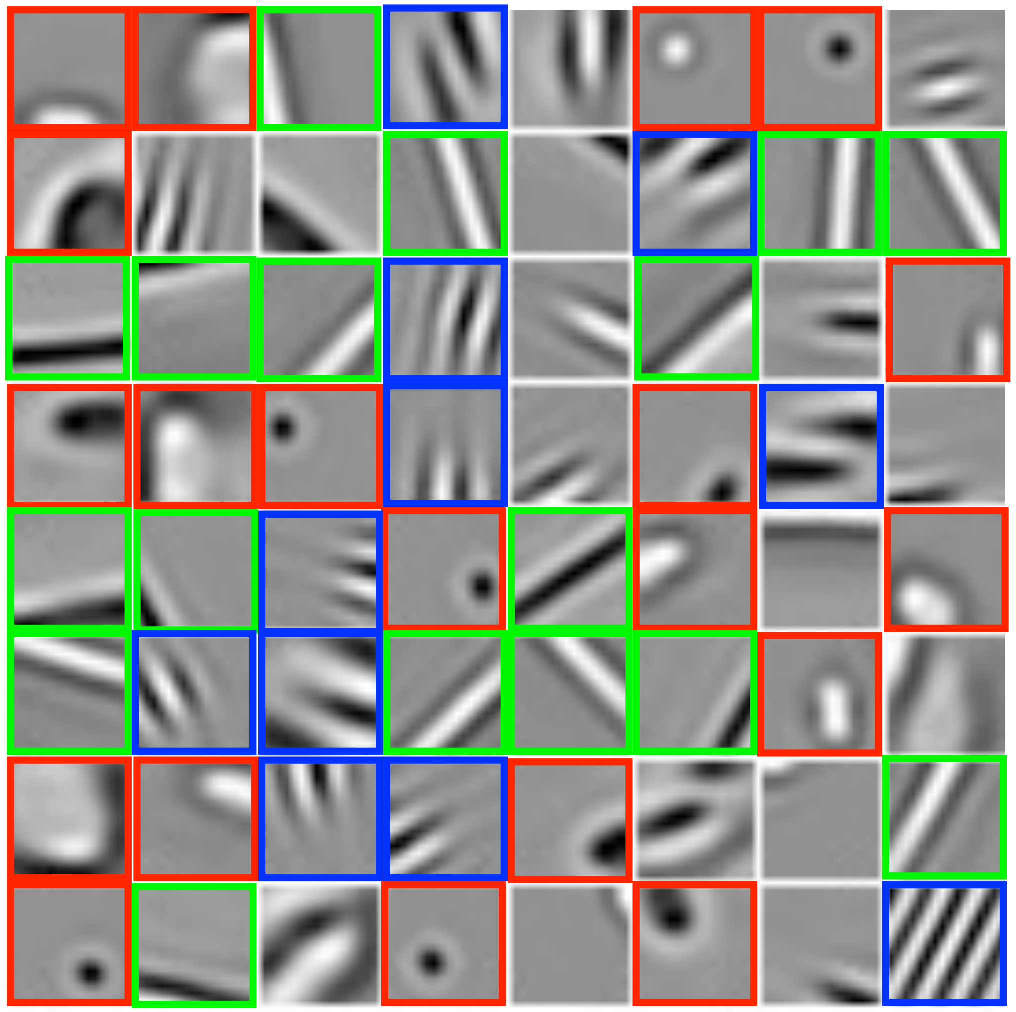
- w| increased overcompleteness and sparsity



source: Olshausen

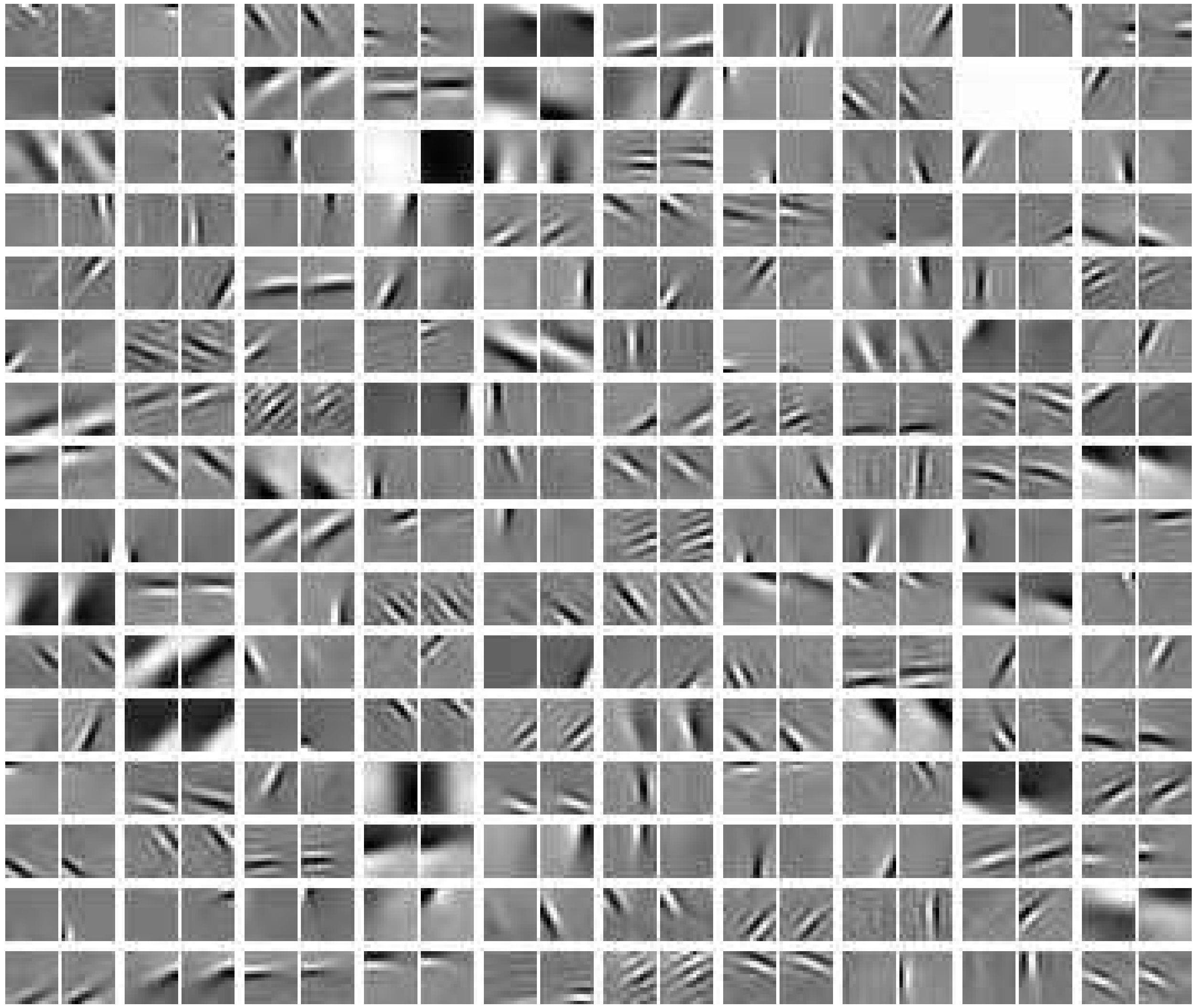
Sparse code for natural images

- w| increased overcompleteness and sparsity



source: Olshausen

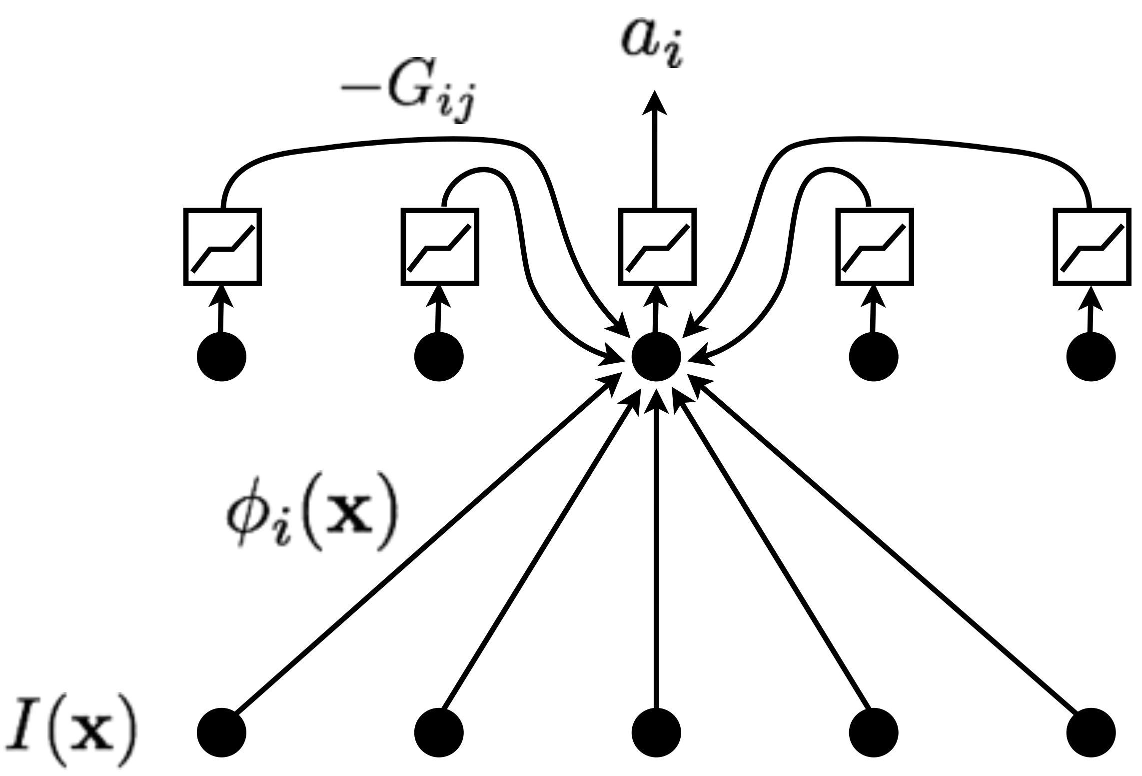
Extensions to color and disparity



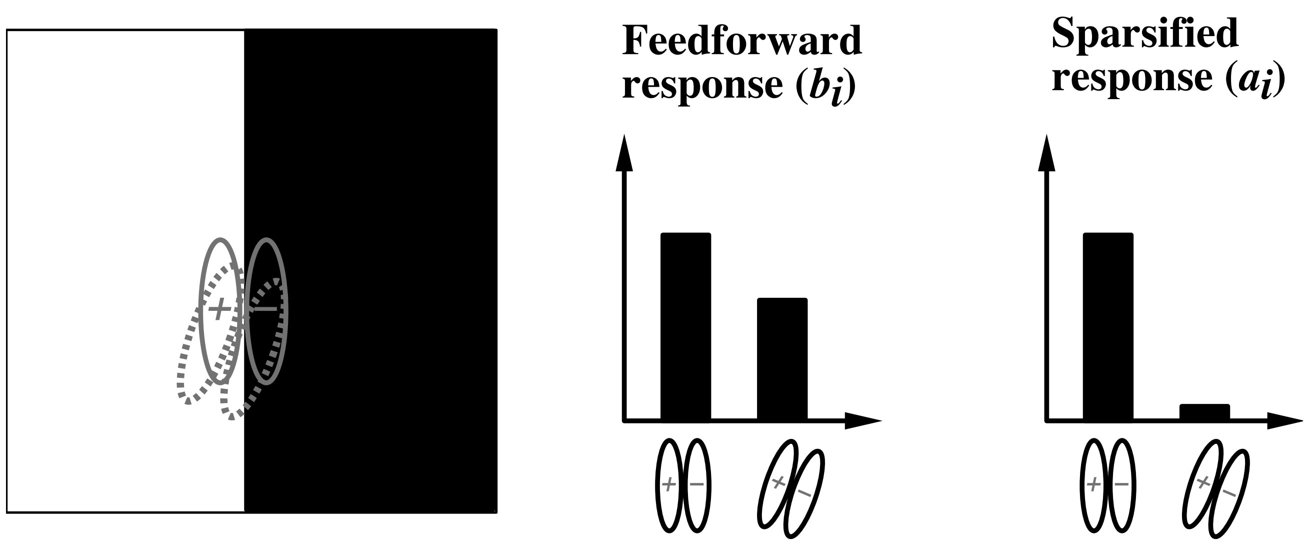
Wachtler, Lee and Sejnowski (2001), Hoyer & Hyvarinen (2000)

Non-linear encoding

Solutions may be computed by a network of leaky integrators and threshold units (Rozell et al. 2008)



‘Explaining away’



source: Olshausen