Sparse Coding

Barlow (1972)

Perception, 1972, volume 1, pages 371–394

Single units and sensation: A neuron doctrine for perceptual psychology?

H B Barlow
Department of Physiology–Anatomy, University of California, Berkeley, California 94720
Received 6 December 1972

Abstract. The problem discussed is the relationship between the firing of single neurons in sensory pathways and subjectively experienced sensations. The conclusions are formulated as the following five dogmas:
1. To understand nervous function one needs to look at interactions at a cellular level, rather than either a more macroscopic or microscopic level, because behaviour depends upon the organized

2. The sensory system is organized to achieve as complete a representation of the sensory stimulus as possible with the minimum number of active neurons.

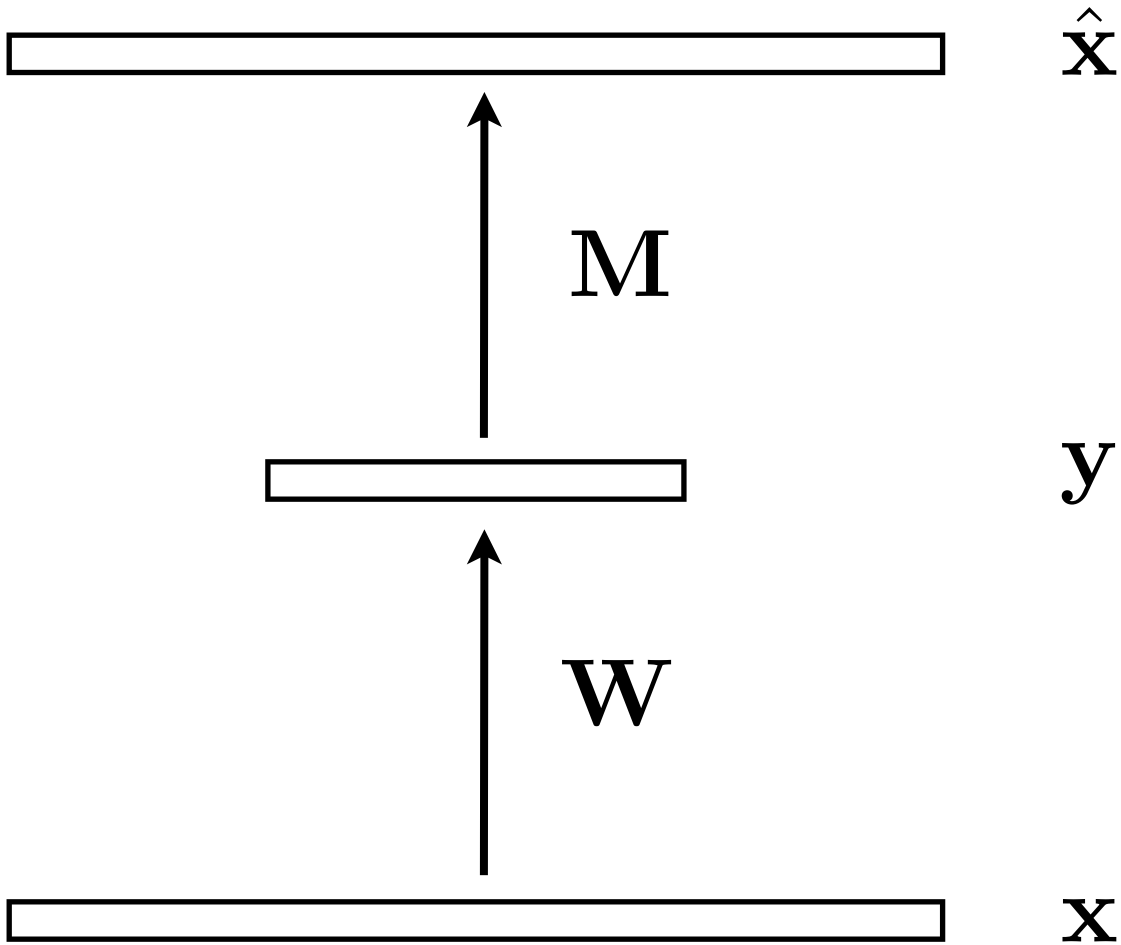
neurons, each of which corresponds to a pattern of external events of the order of complexity of the events symbolized by a word.
5. High impulse frequency in such neurons corresponds to high certainty that the trigger feature is present.
The development of the concepts leading up to these speculative dogmas, their experimental basis, and some of their limitations are discussed.

Barlow (1972)

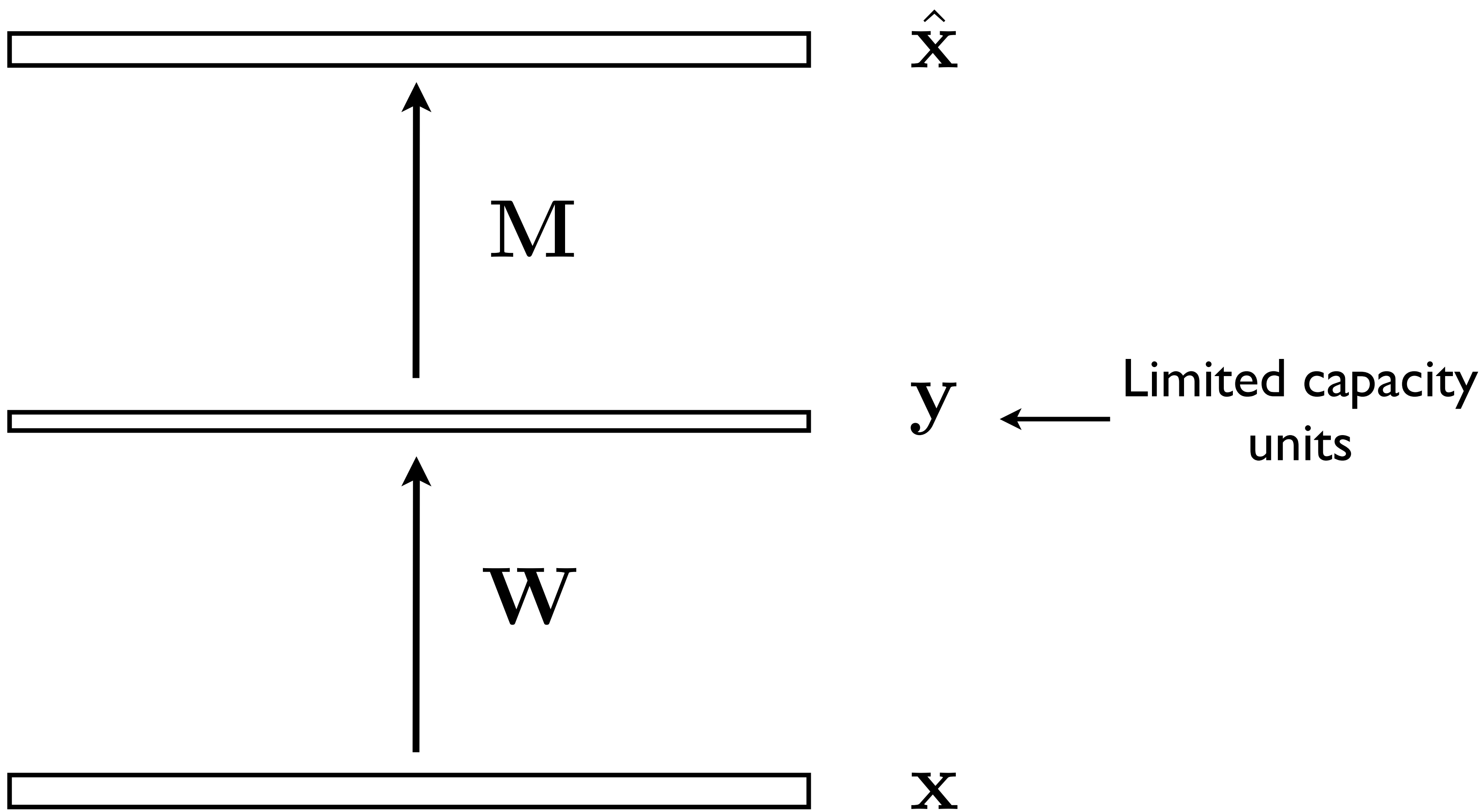
The second dogma goes beyond the evidence, but it attempts to make sense out of it. It asserts that the overall direction or aim of information processing in higher sensory centres is to represent the input as completely as possible by activity in as few neurons as possible (Barlow, 1961, 1969b). In other words, not only the proportion but also the actual number of active neurons, K, is reduced, while as much information as possible about the input is preserved.

Autoencoder networks

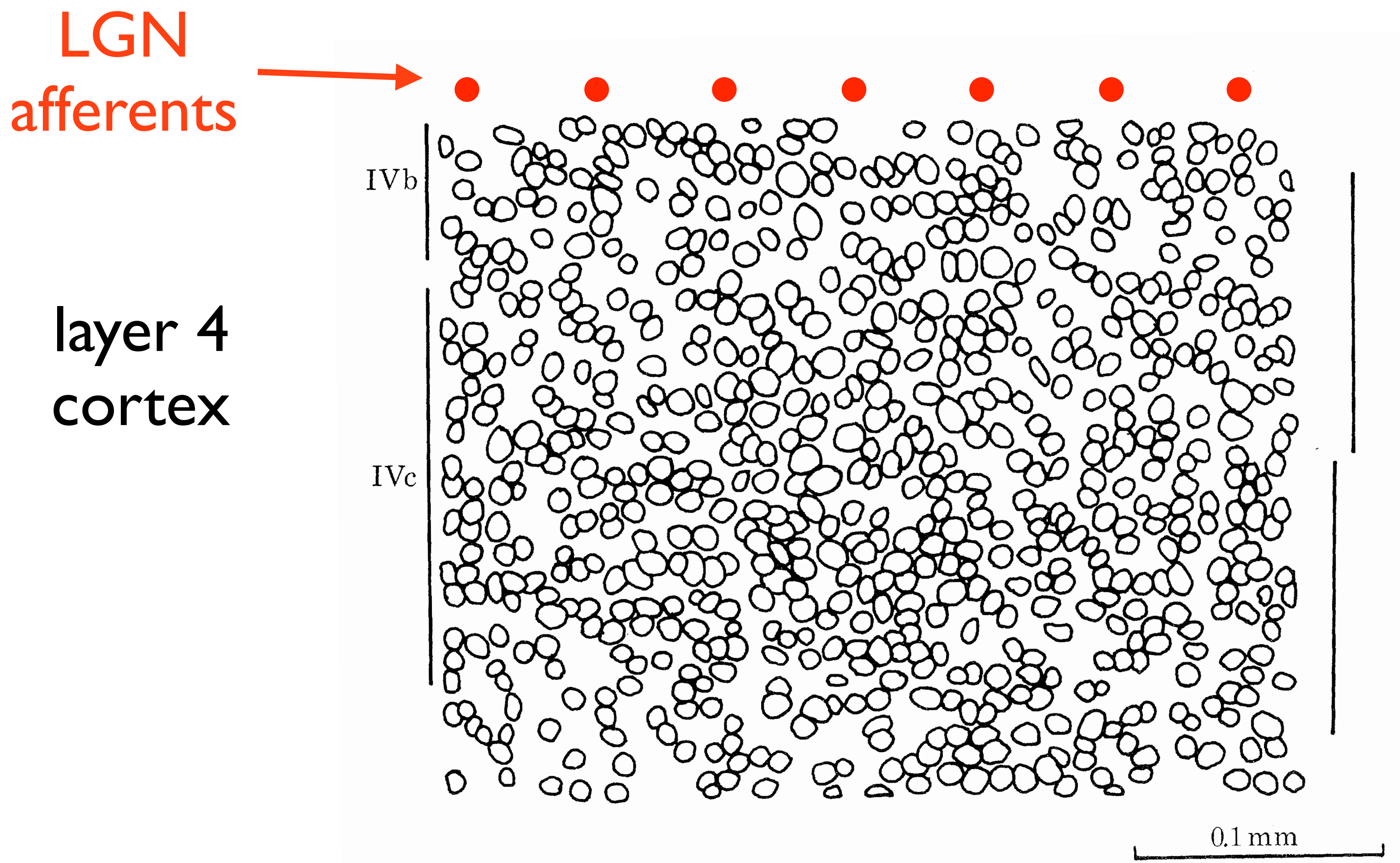
$$\min_{\mathbf{W},\mathbf{M}} |\mathbf{x} - \hat{\mathbf{x}}|^2$$



Bottleneck may also be in the form of limited capacity units.
Optimal strategy in this case is to whiten.

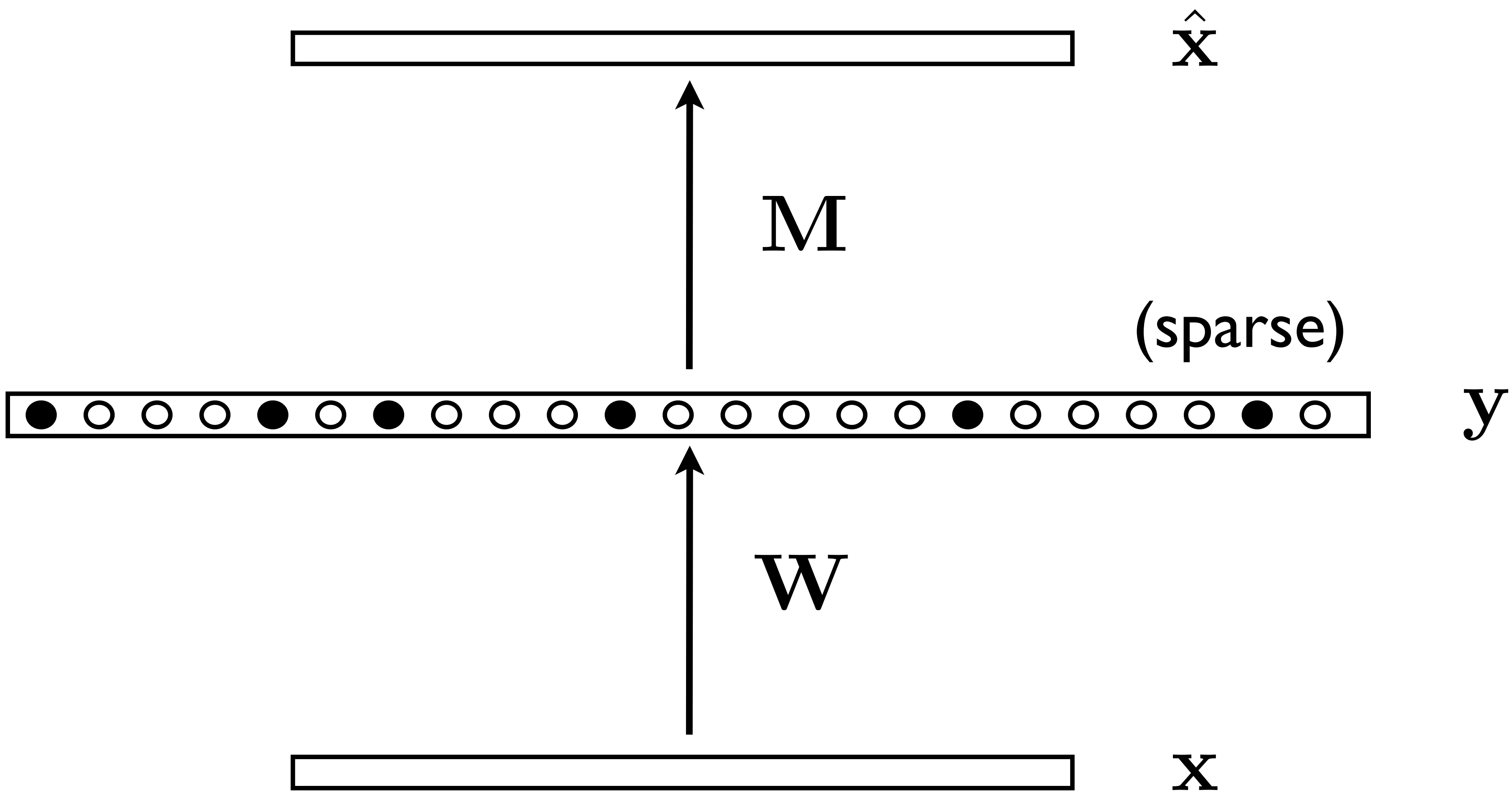


V1 is highly overcomplete



Barlow (1981)

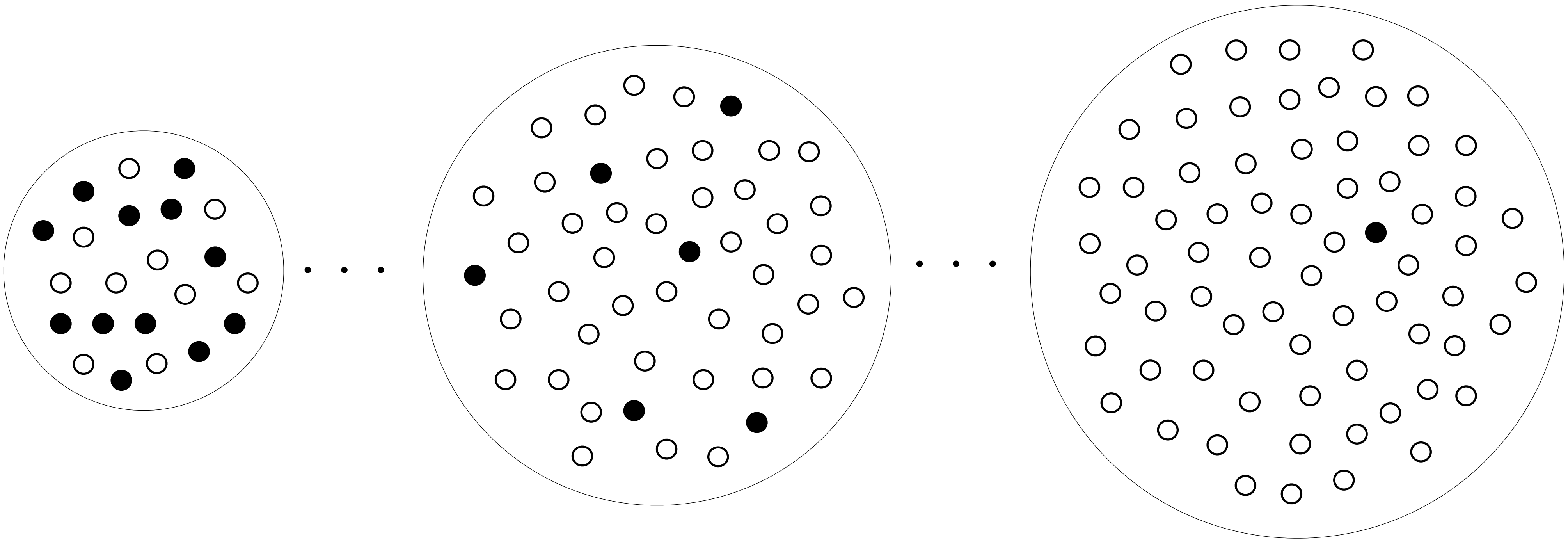
Sparse codes impose a different type of bottleneck by limiting the number of active units



Dense codes
(e.g., ascii)

Sparse, distributed codes

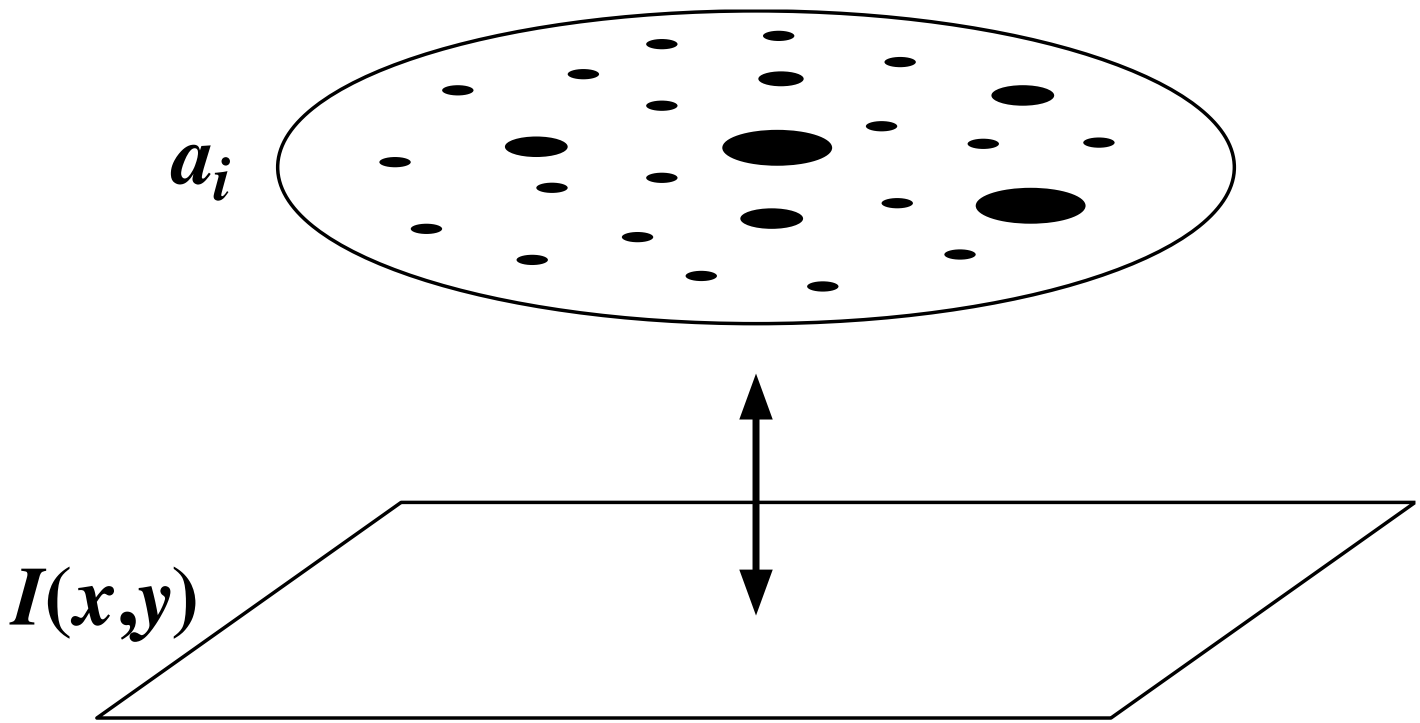
Local codes
(e.g., grandmother cells)



2^N $\binom{N}{K}$ N

From: Foldiak & Young (1995)

How to form a sparse code of images?



Biol. Cybern. 64, 165–170 (1990)

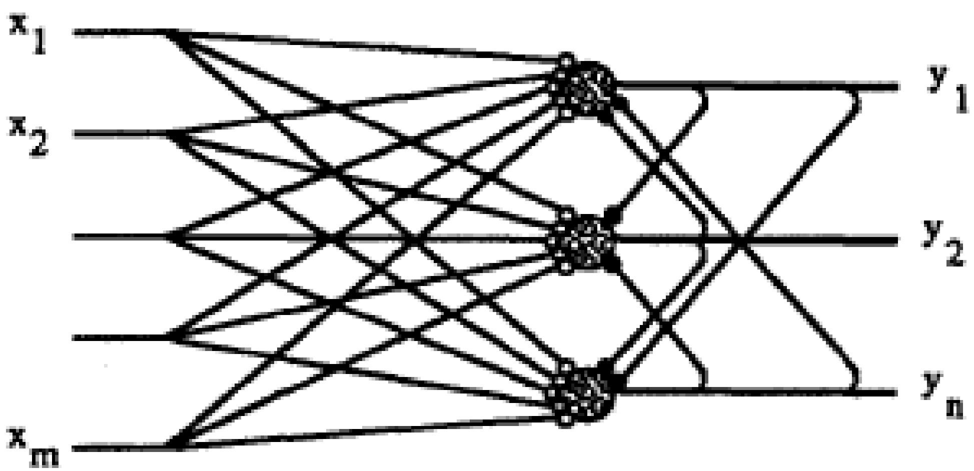
Biological Cybernetics

Forming sparse representations by local anti-Hebbian learning

P. Földiák

Physiological Laboratory, University of Cambridge, Downing Street, Cambridge CB2 3EG, United Kingdom

$$\frac{dy_i^*}{dt} = f\left(\sum_{j=1}^{m} q_{ij}x_j + \sum_{j=1}^{n} w_{ij}y_j^* - t_i\right) - y_i^*$$



anti-Hebbian rule–

$$\Delta w_{ij} = -\alpha(y_i y_j - p^2)$$

(if $i = j$ or $w_{ij} > 0$ then $w_{ij} := 0$)

Hebbian rule–

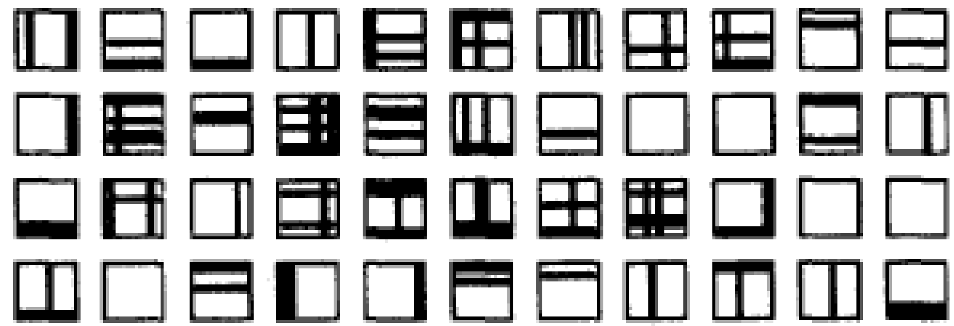
$$\Delta q_{ij} = \beta y_i (x_j - q_{ij})$$

threshold modification–

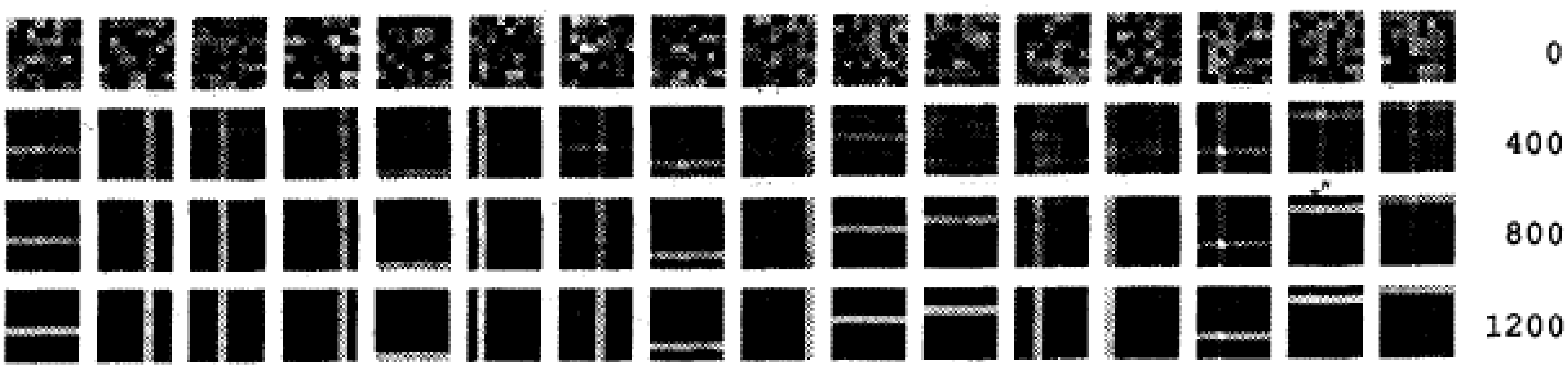
$$\Delta t_i = \gamma(y_i - p) .$$

Learning lines

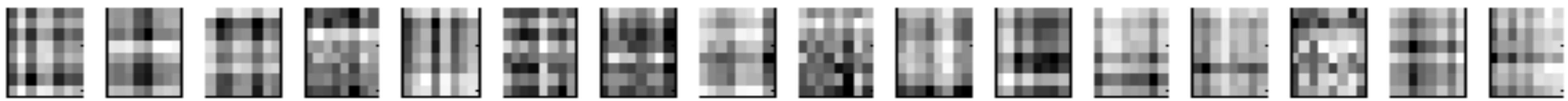
Input patterns:



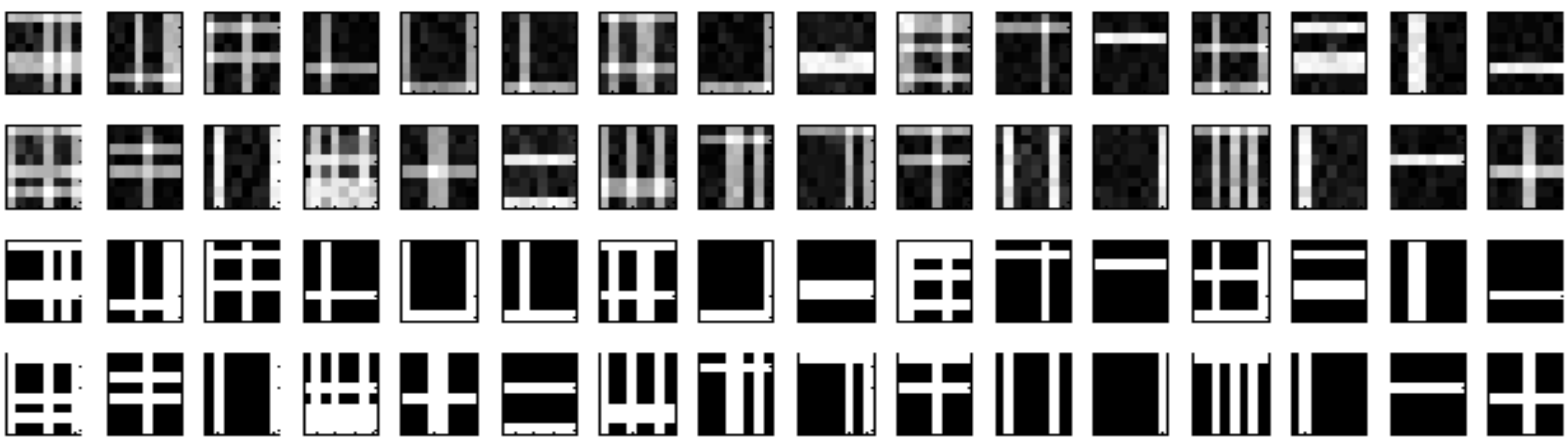
Learned weights:



PCA solution



Reconstructions



Problems

- How to deal with graded input signals? (i.e., real images)
- No objective function

Sparse coding model for graded signals

(Olshausen & Field, 1996)

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

image

neural activities (sparse)

features

other stuff

Energy function

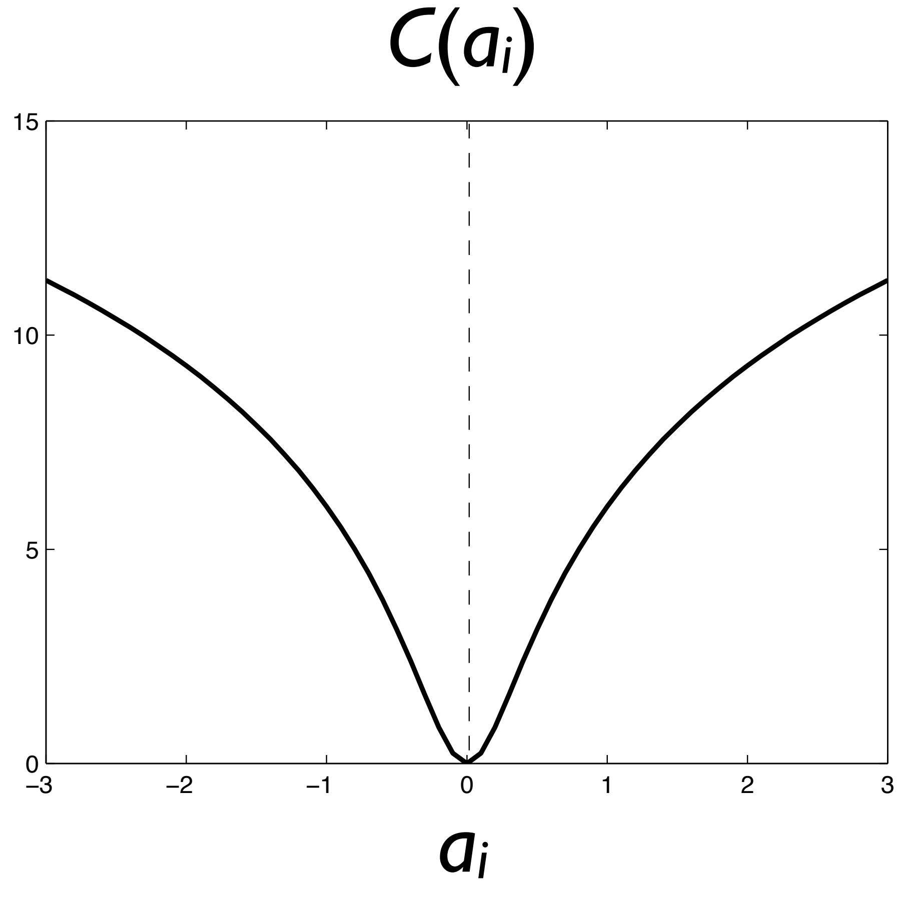
$$E = \frac{1}{2}|\mathbf{I} - \Phi\,\mathbf{a}|^2 + \lambda \sum_i C(a_i)$$

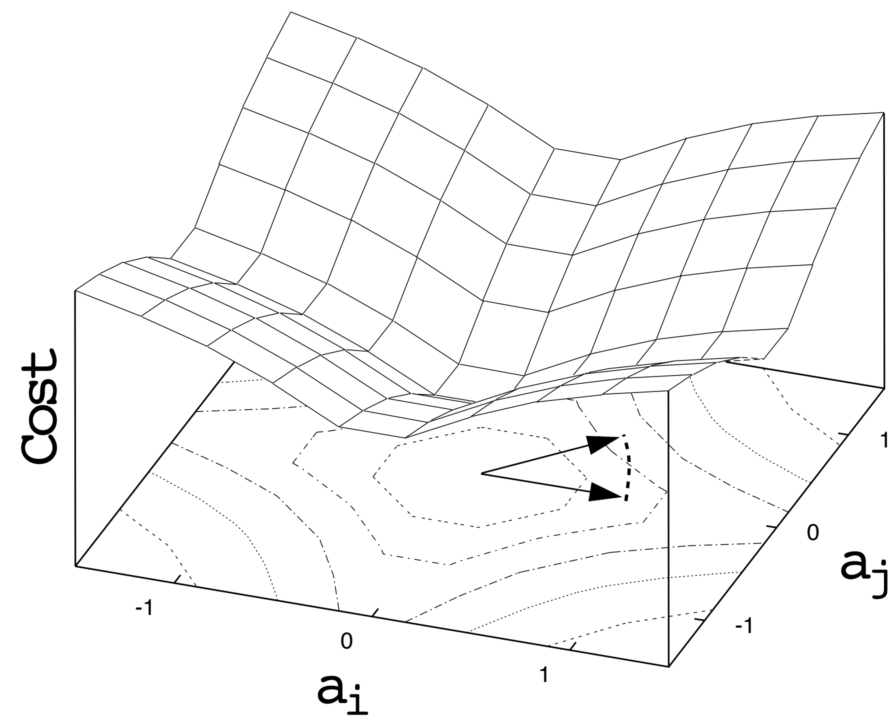
preserve information

be sparse

Cost function

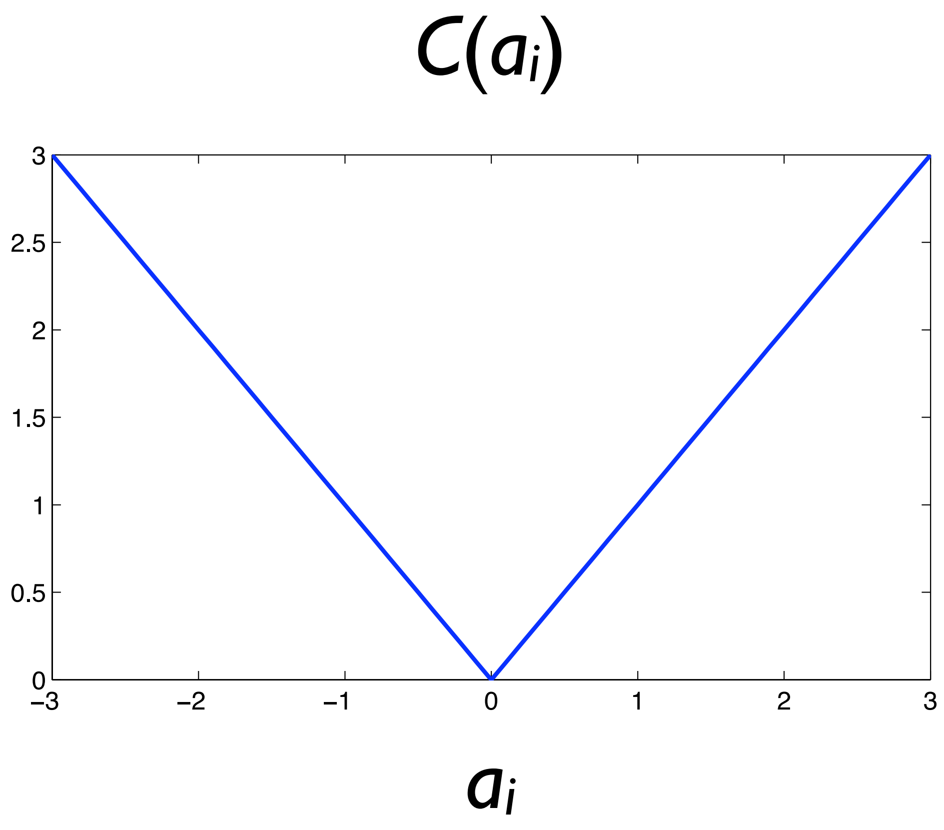
$$C(a_i) = \log(1 + a_i^2)$$

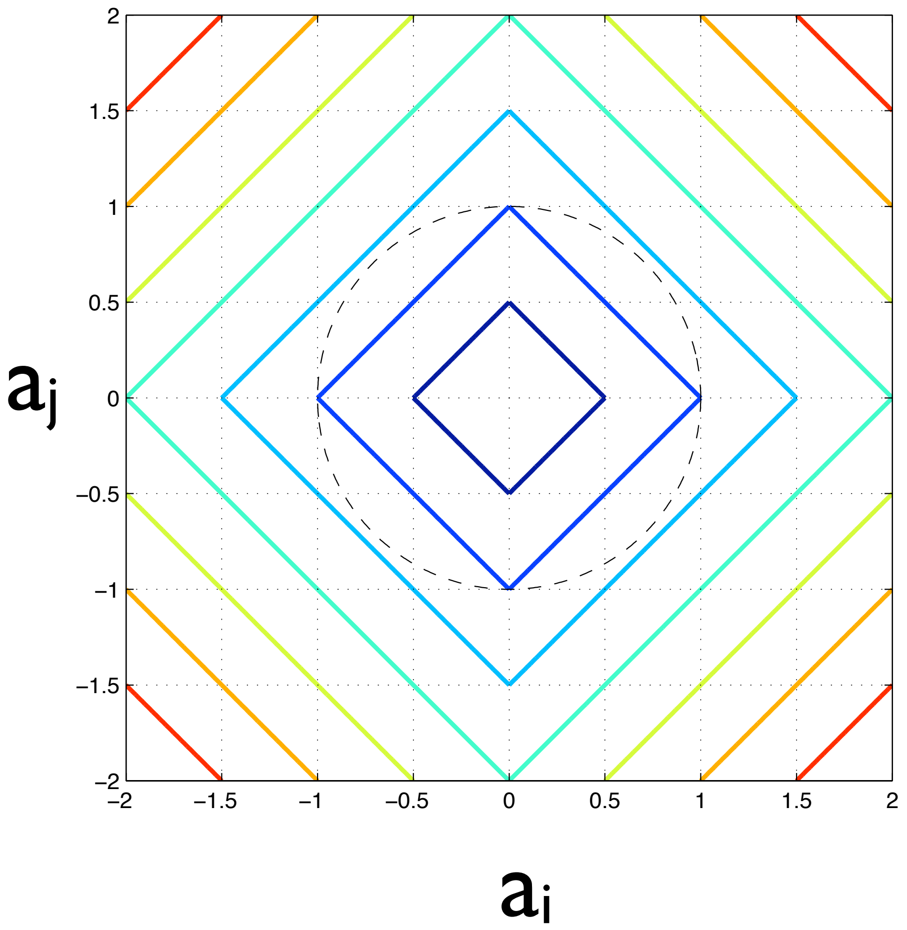




Cost function

$$C(a_i) = |a_i|$$





Compute coefficients via gradient descent

$$
\begin{aligned}
\tau\, \dot{a}_i &= -\frac{dE}{da_i} \\
&= b_i - \sum_{j \neq i} G_{ij}\, a_j - f_\lambda(a_i)
\end{aligned}
$$

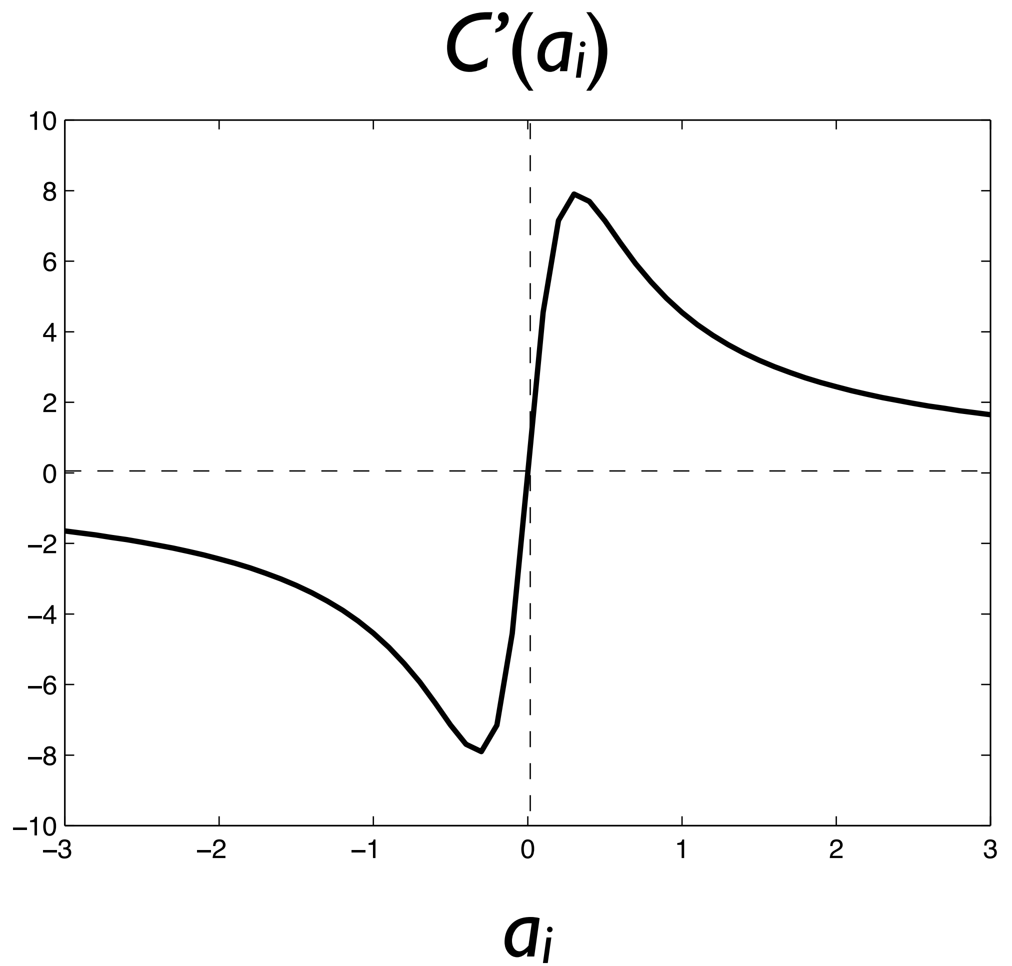
Where

$$b_i = \sum_{x,y} \phi_i(x,y)\, I(x,y)$$

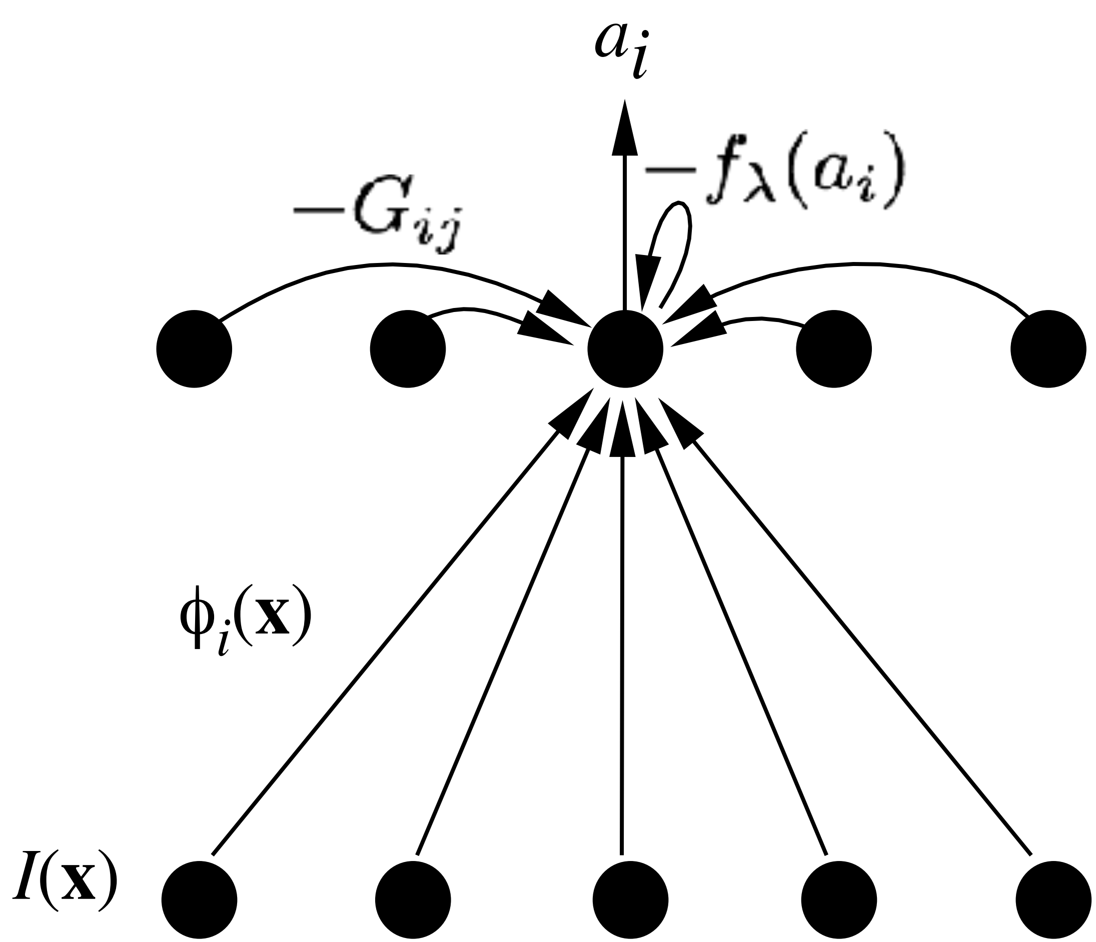
$$G_{ij} = \sum_{x,y} \phi_i(x,y)\, \phi_j(x,y)$$

$$f_\lambda(a_i) = a_i + \lambda\, C'(a_i)$$

Sparse cost derivative (*C'*)



Network implementation



Alternative formulation (the Hopfield trick)

Let

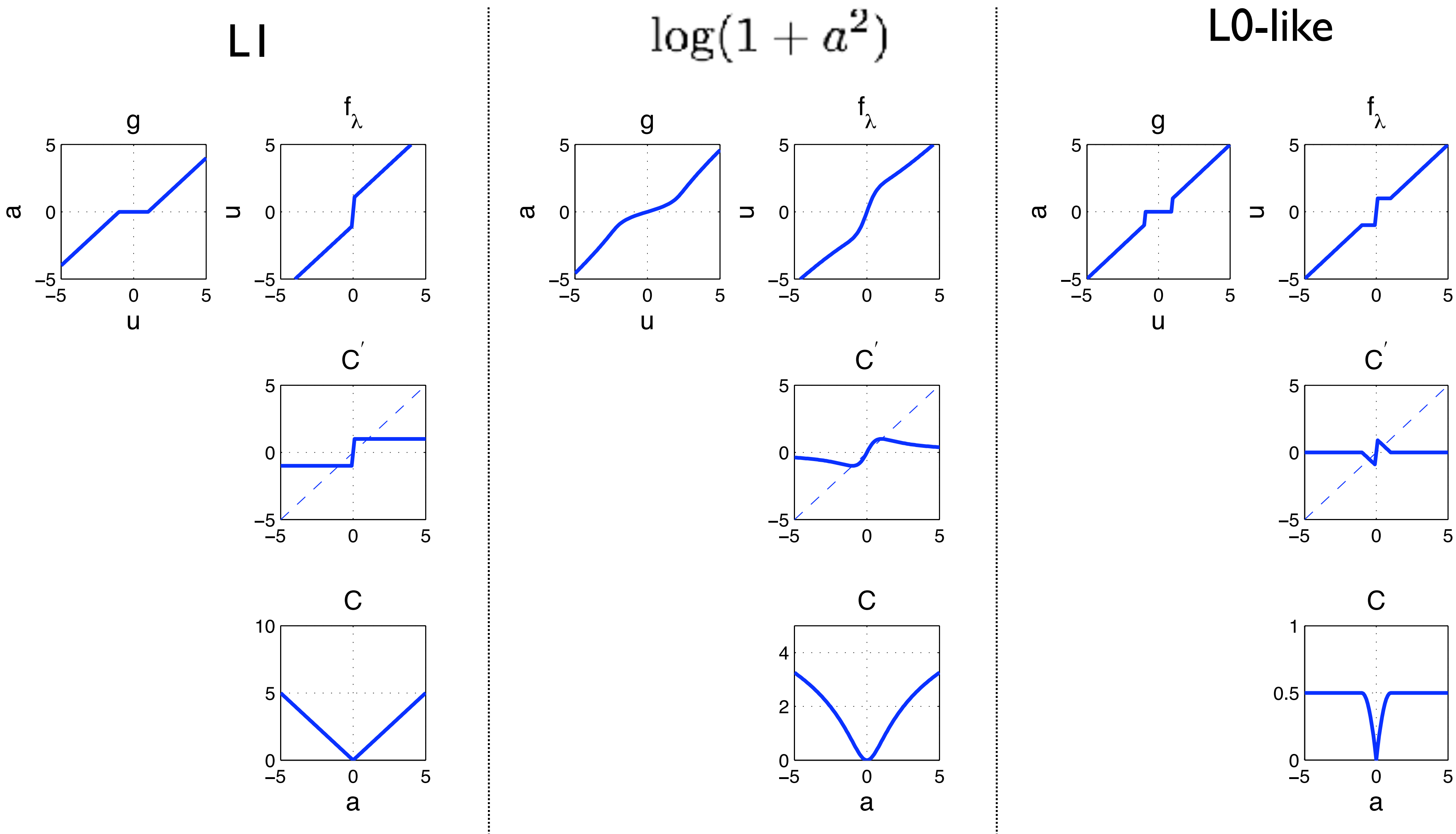
$$u_i = f_\lambda(a_i), \quad \text{or} \quad a_i = f_\lambda^{-1}(u_i) \equiv g(u_i)$$

$$\begin{aligned} \tau\, \dot{u}_i &= -\frac{dE}{da_i} \\ &= b_i - \sum_{j \neq i} G_{ij}\, a_j - u_i \end{aligned}$$

Thus

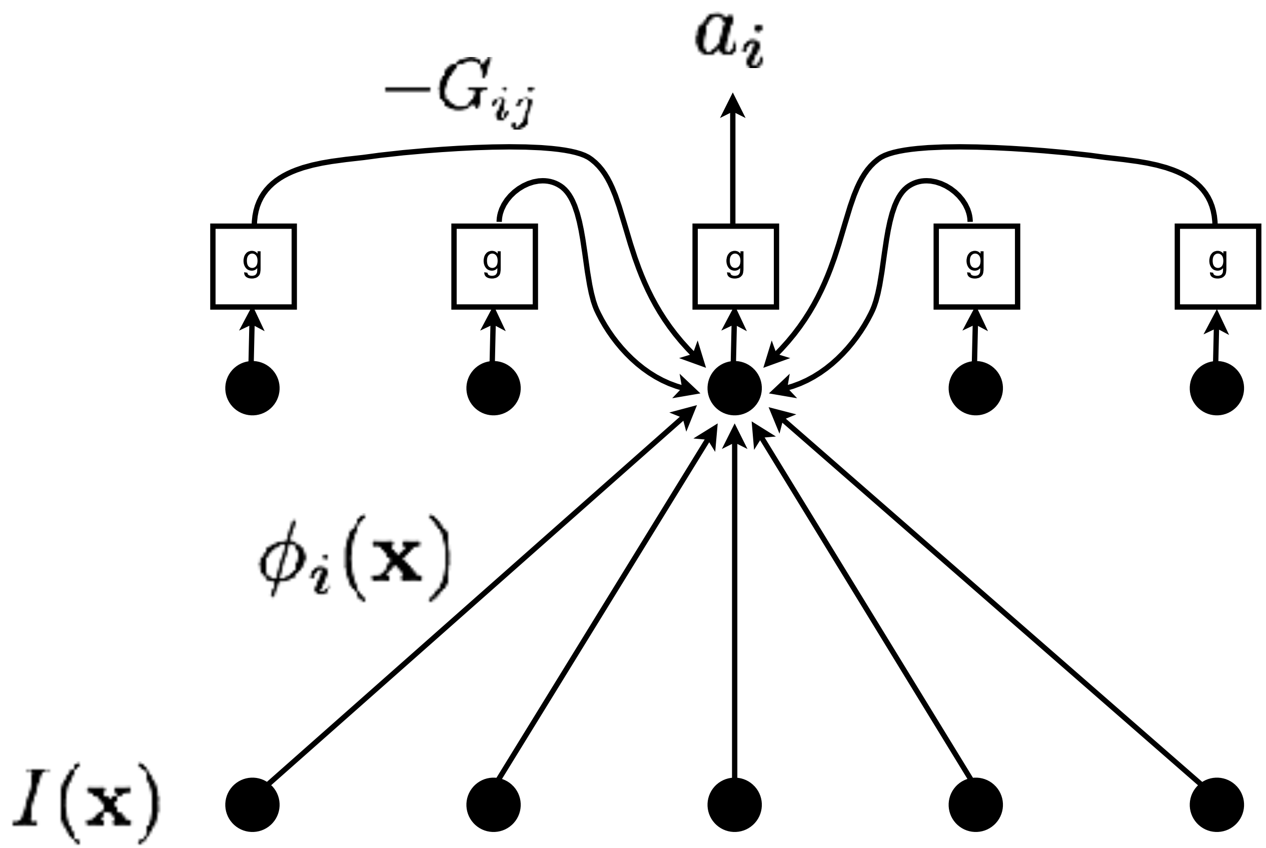
$$\boxed{\begin{aligned} \tau\, \dot{u}_i + u_i &= b_i - \sum_{j \neq i} G_{ij}\, a_j \\ a_i &= g(u_i) \end{aligned}}$$

Relation between the thresholding function *g* and cost function *C*



Coefficients may be computed simply via thresholding and lateral inhibition

(Rozell, Johnson, Baraniuk & Olshausen, 2008)

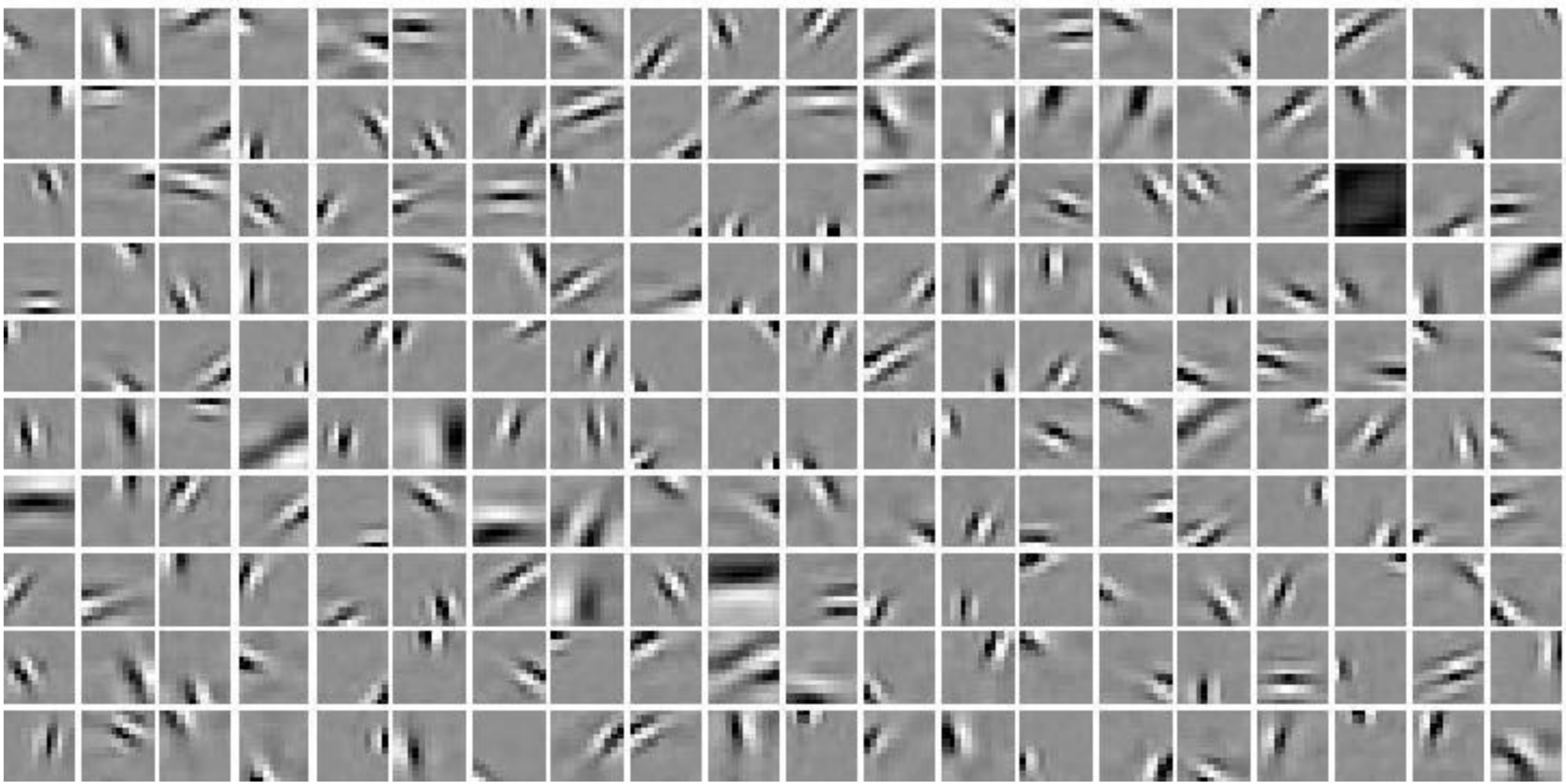


Learning rule

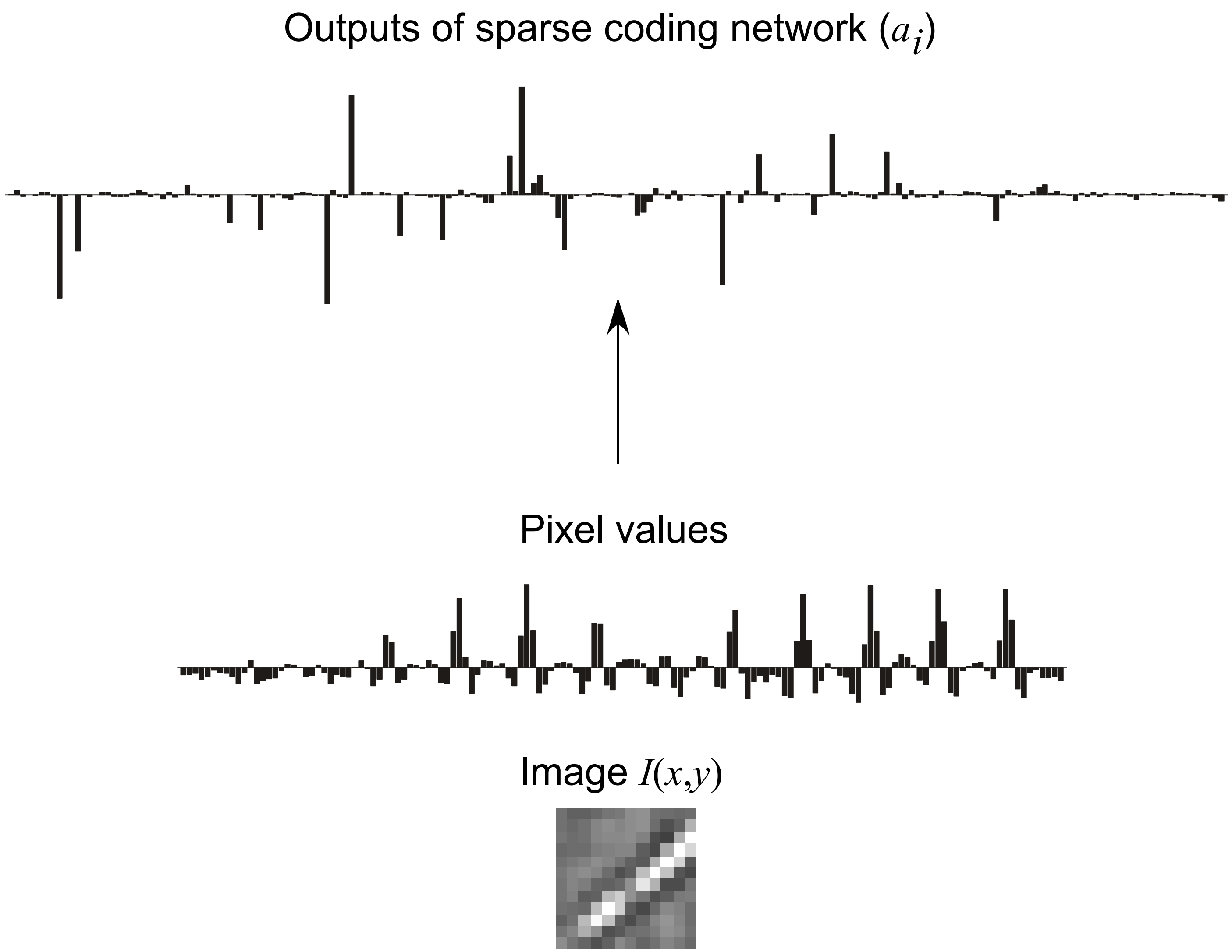
$$\begin{aligned} \Delta\phi_i &= -\eta \frac{\partial E}{\partial \phi_i} \\ &= [\mathbf{I} - \Phi\,\hat{\mathbf{a}}]\,\hat{a}_i \end{aligned}$$

Features Φ_i learned from natural images

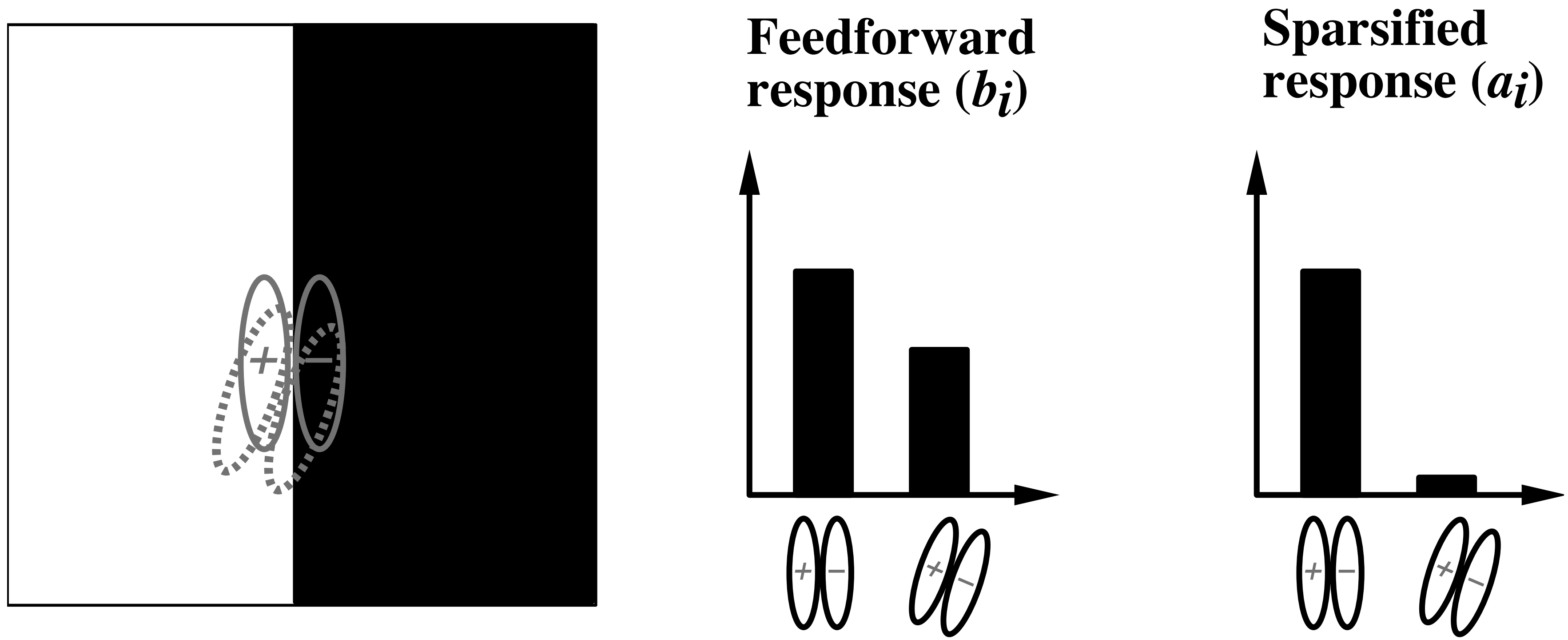
(200, 12x12 pixels)



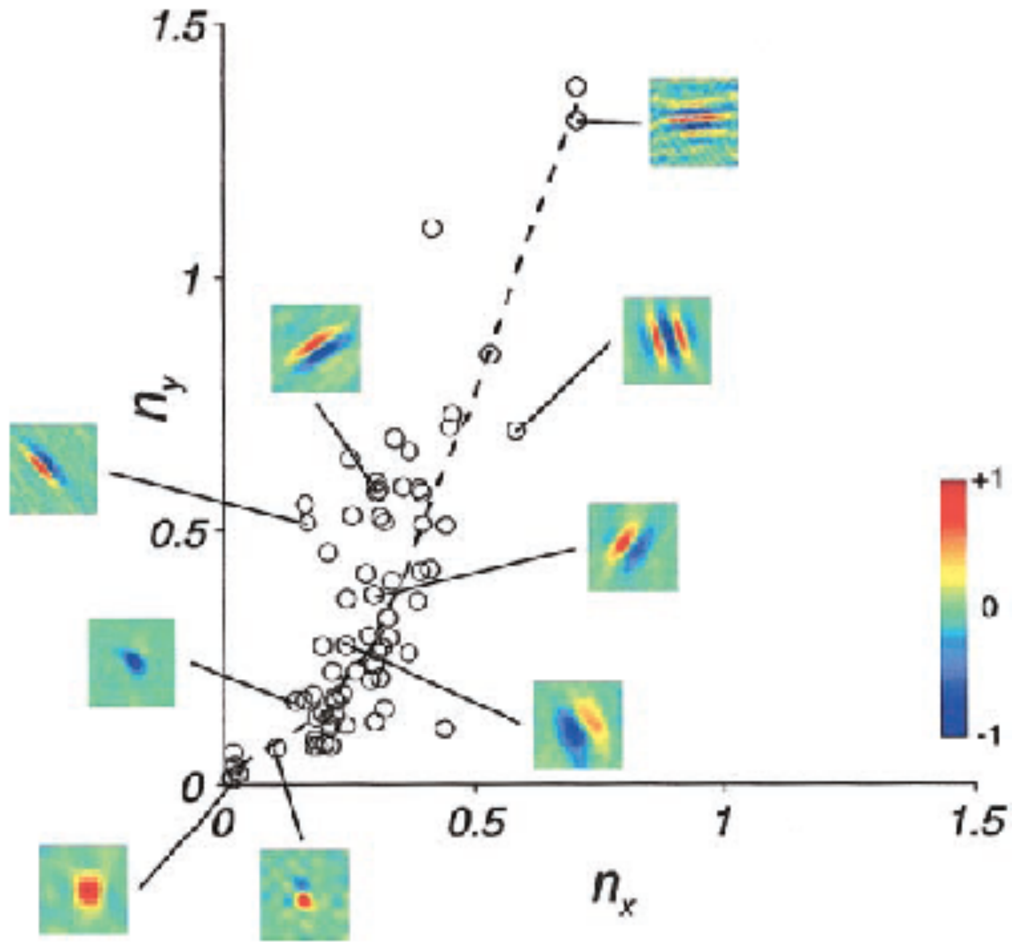
Sparsification

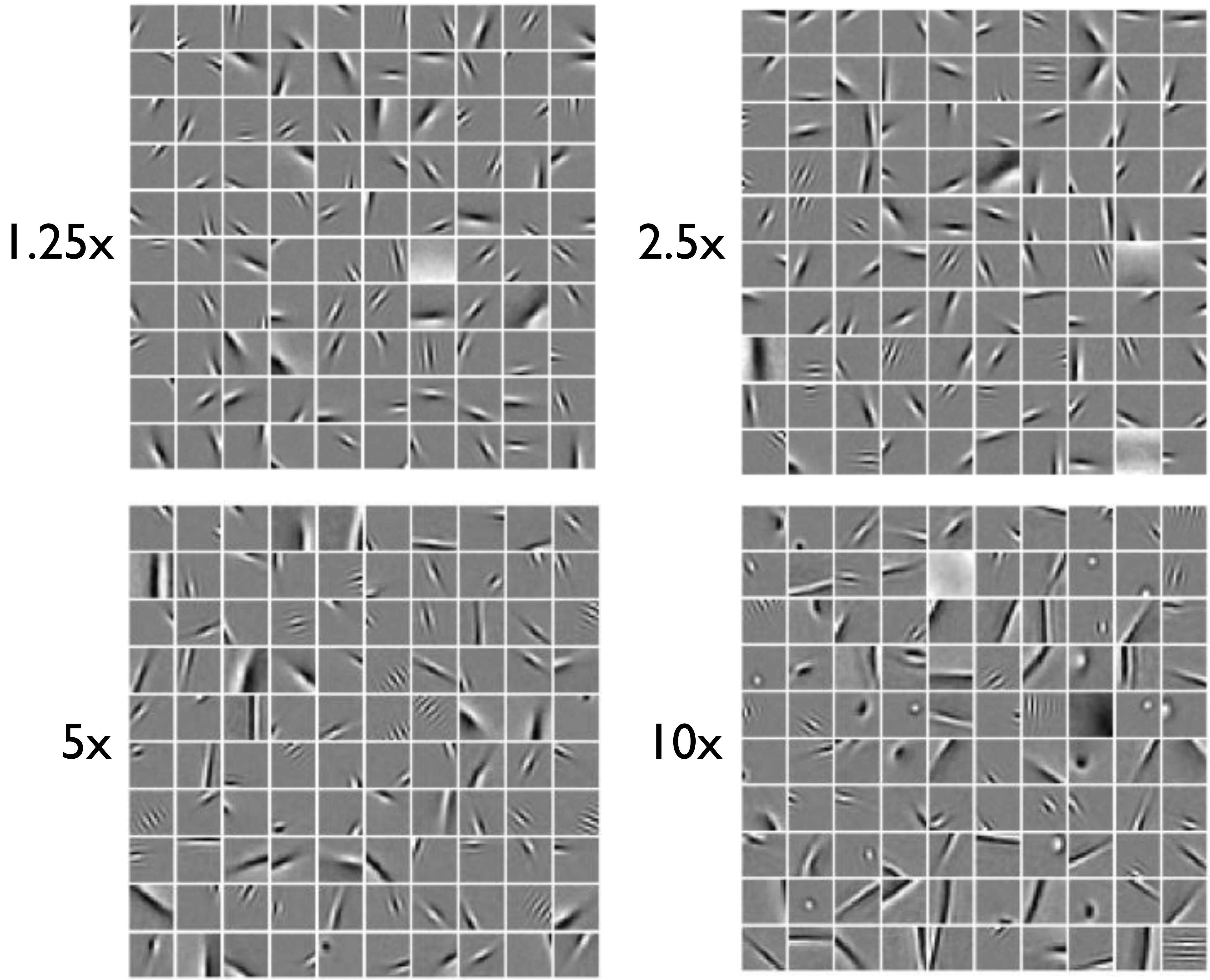


‘Explaining away’

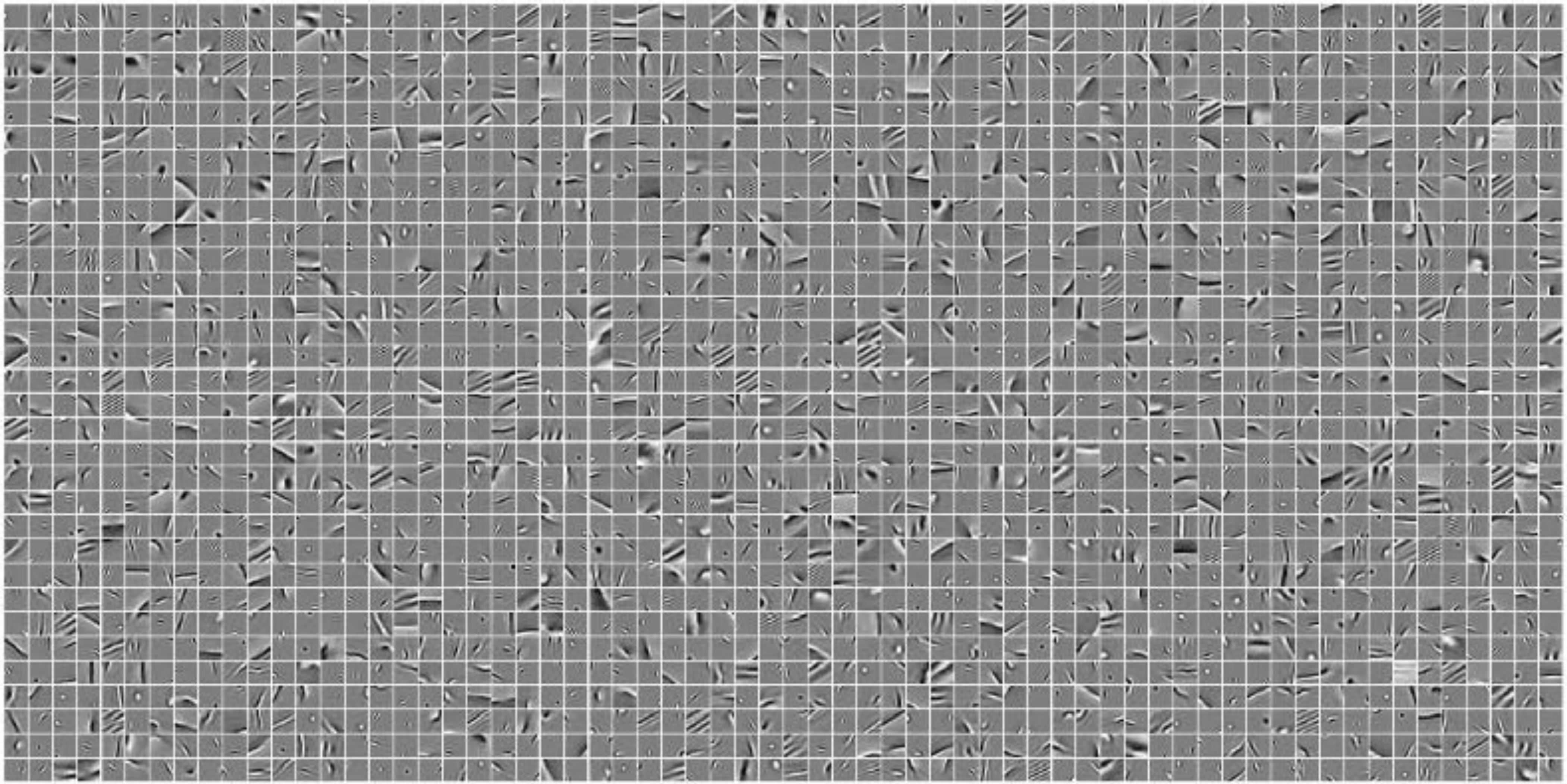


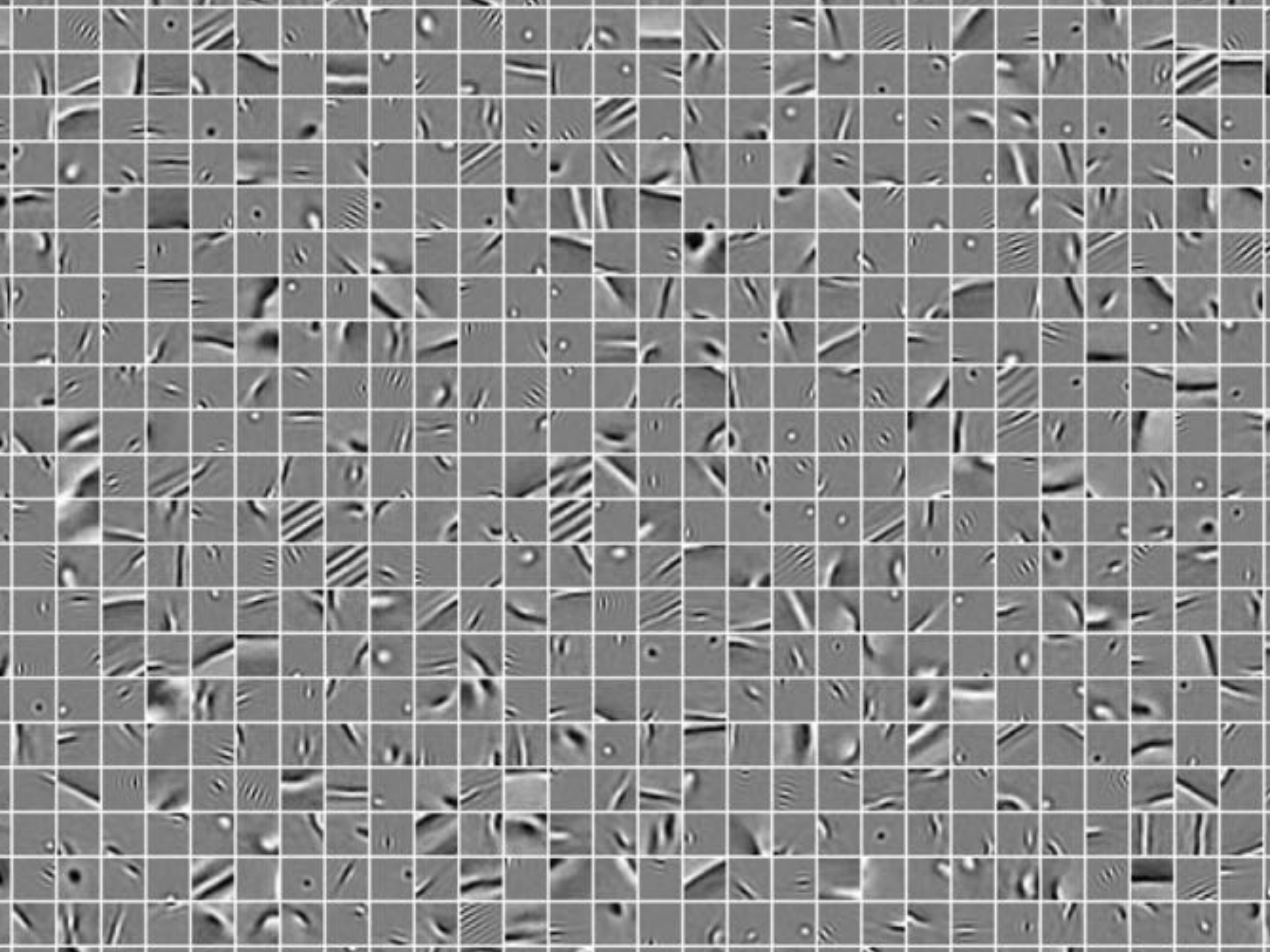
Diversity of simple-cell receptive fields in macaque V1 (Ringach 2002)



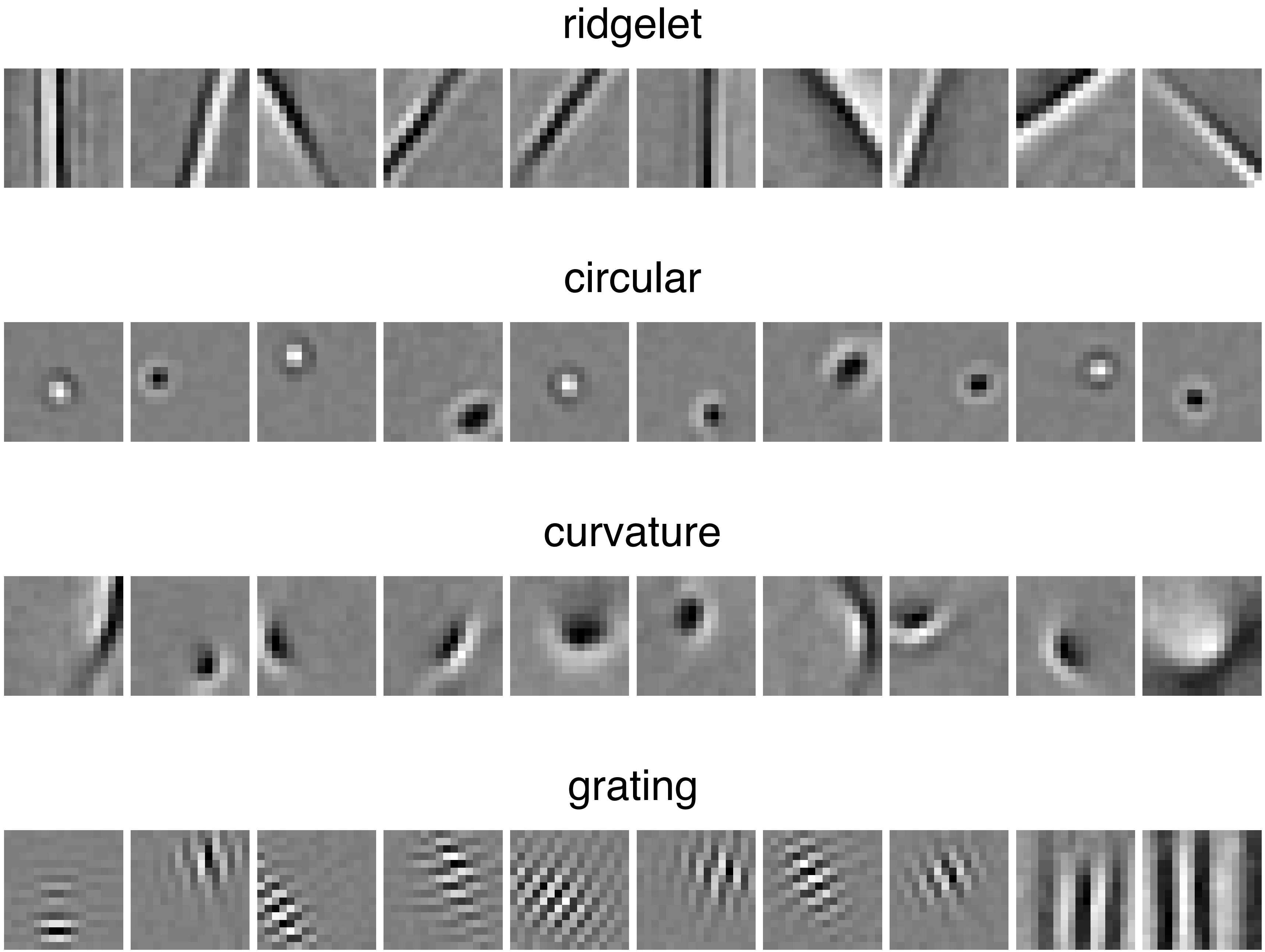

1.25x
2.5x
5x
10x

Full 10x dictionary



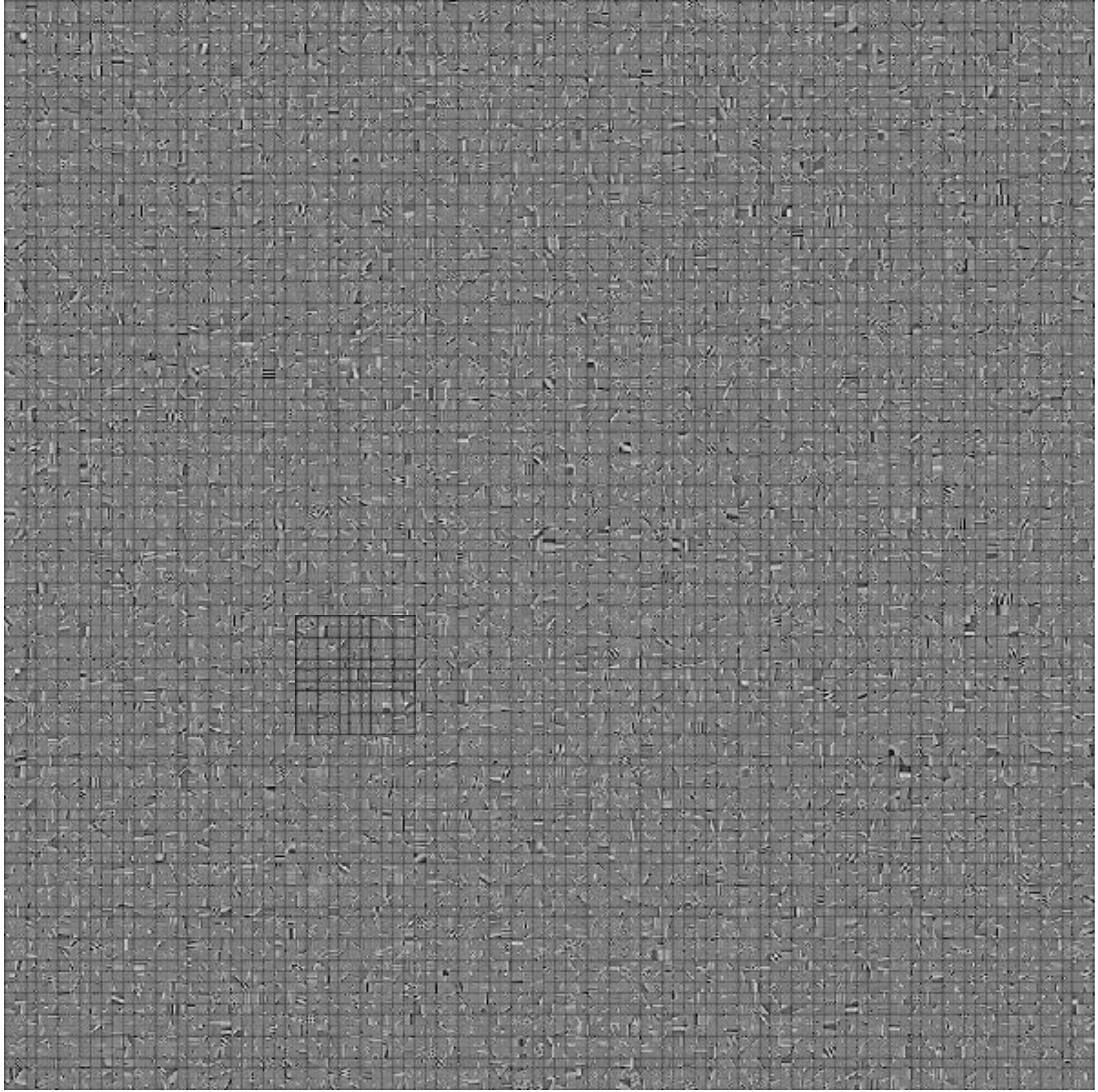


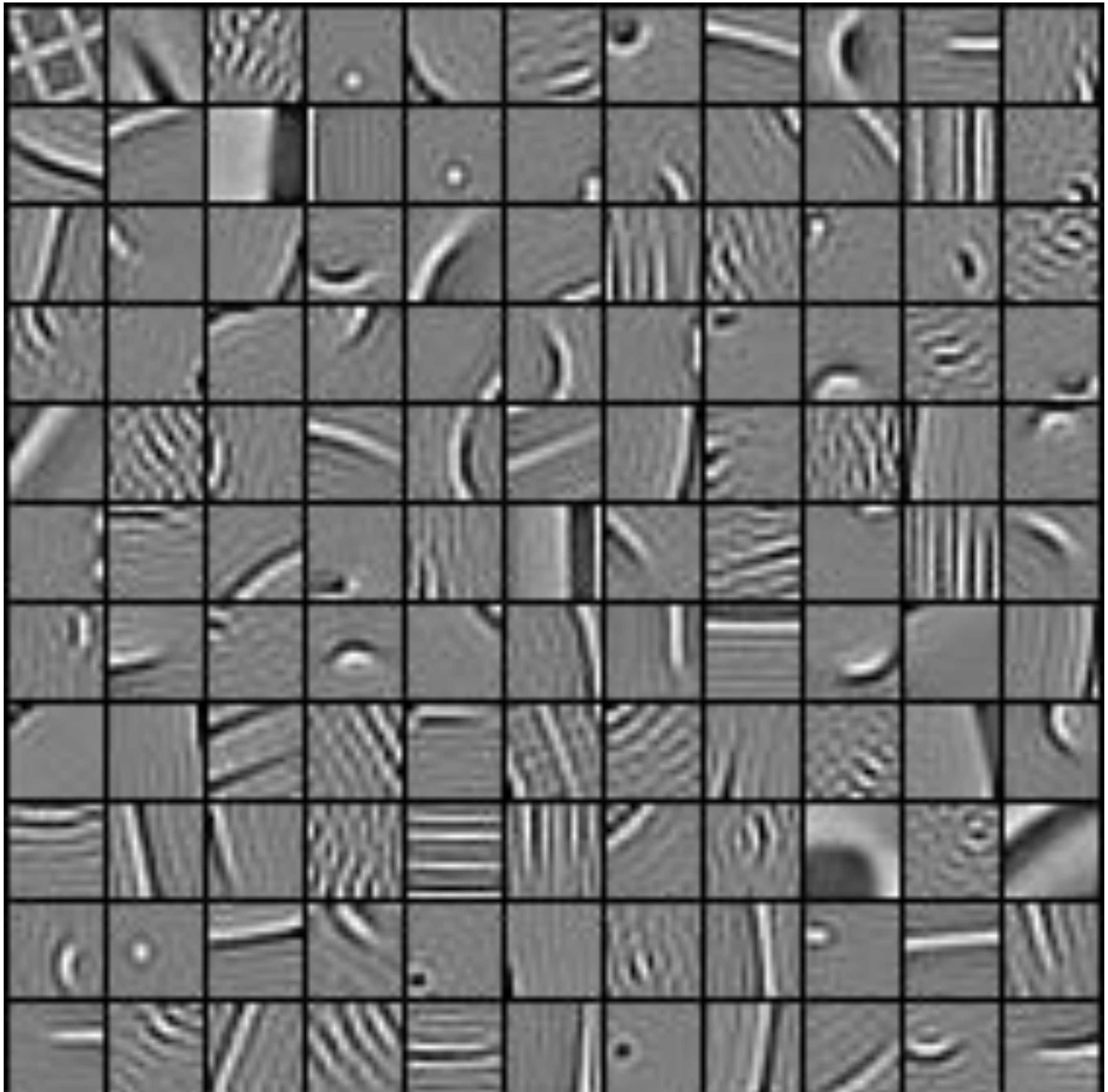
Examples from 10x dictionary
(Olshausen, 2013)



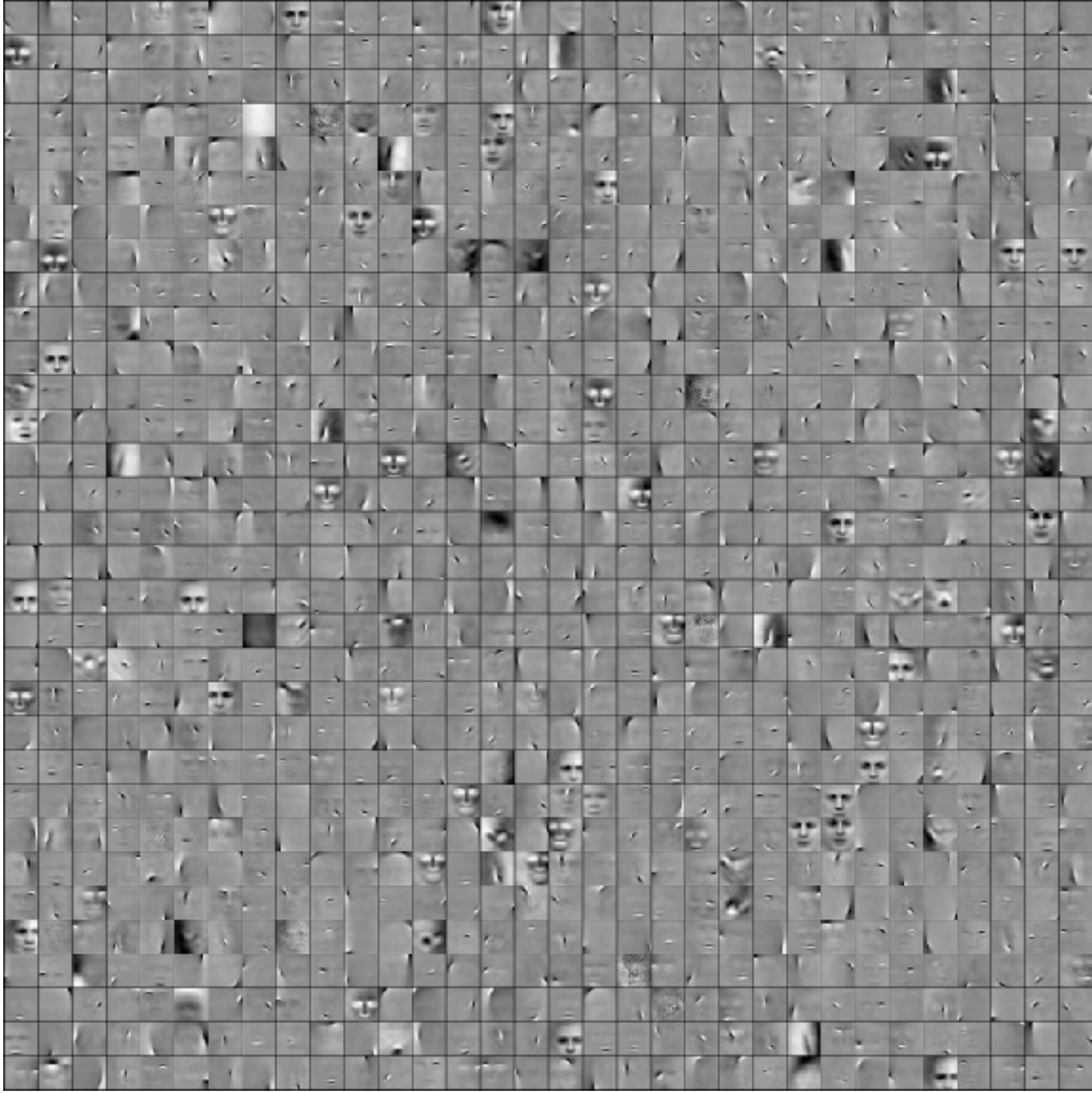
100x overcomplete learned dictionary

(obtained by Charles Cadieu after running for 8 hours on16 GPU's)



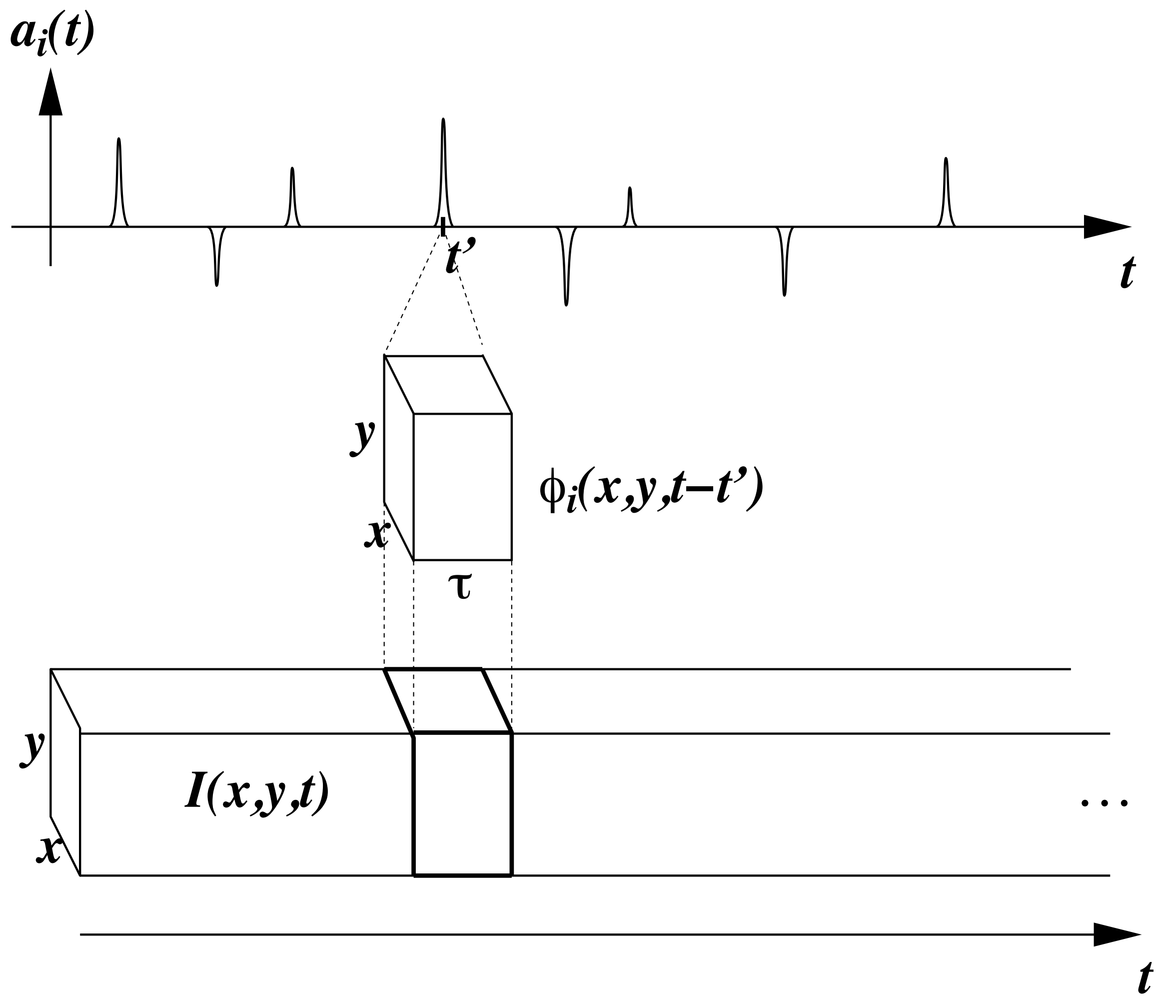


Faces
(charles
cadieu)



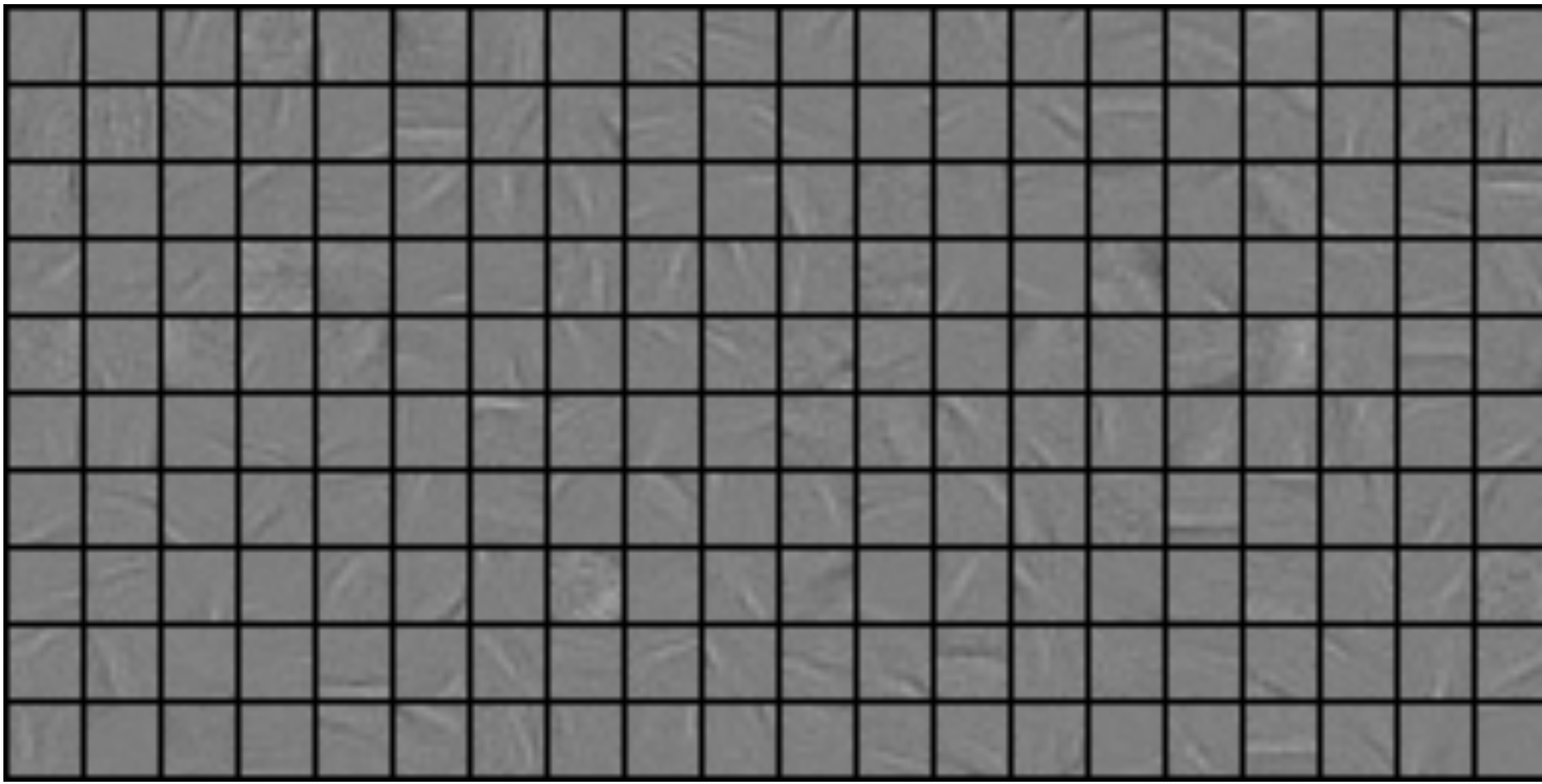
Sparse coding of time-varying images

$$I(x,y,t) = \sum_i a_i(t) * \phi_i(x,y,t) + \nu(x,y,t)$$

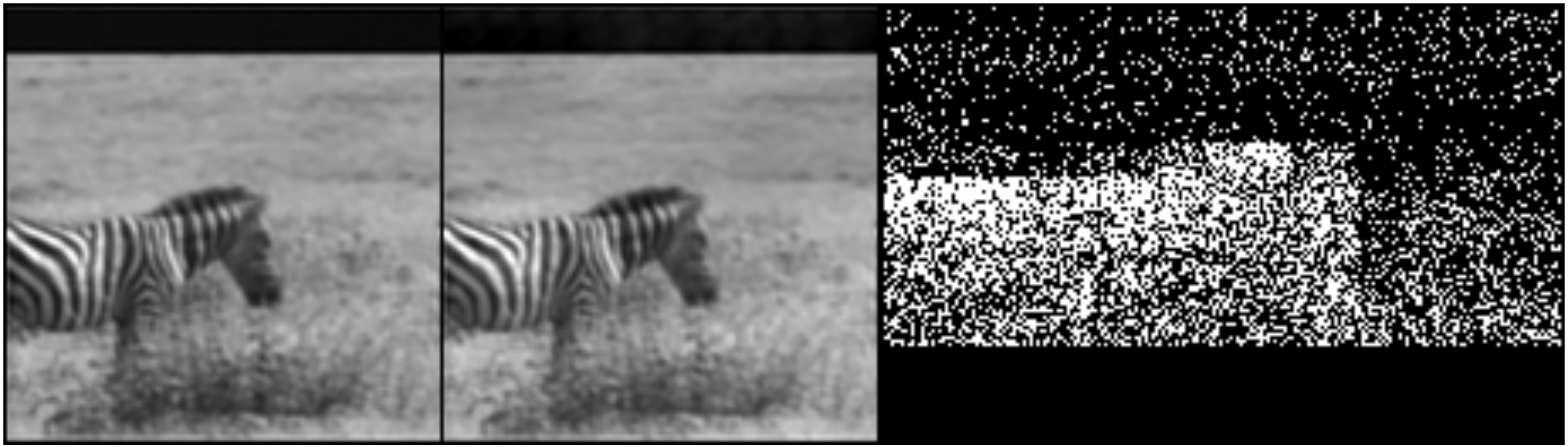


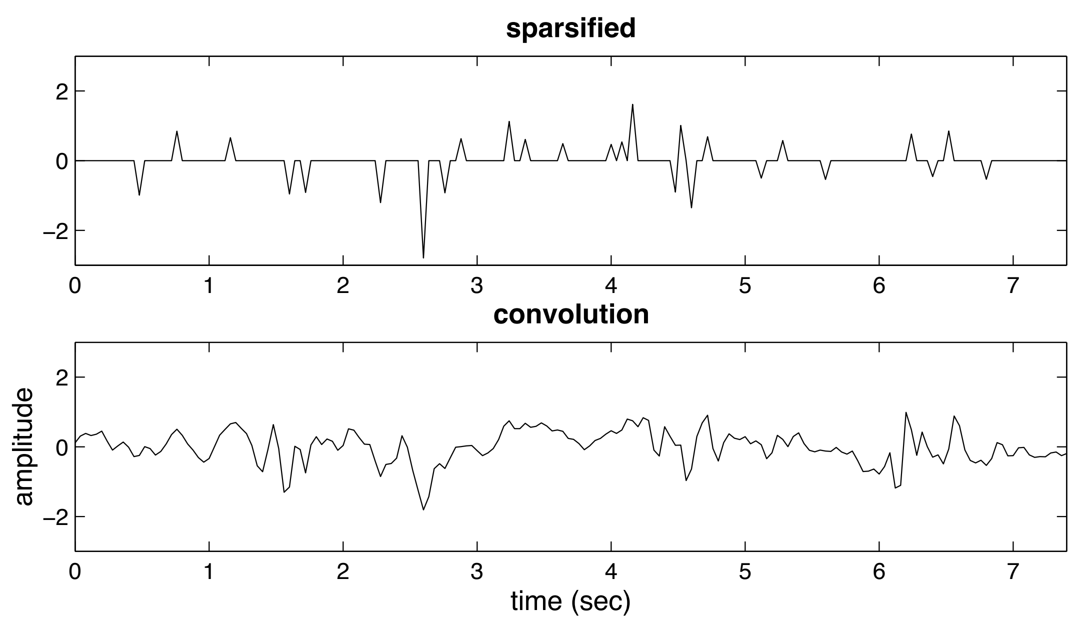
Learned basis space-time basis functions

(200 bfs, 12 x12 x 7)



Sparse coding and reconstruction





Do brains really work this way?

Evidence for sparse coding

Mushroom body, locust (Laurent)

HVC, zebra finch (Fee)

Auditory cortex, mouse (DeWeese & Zador)

Hippocampus, rat/primate (Thompson & Best; Skaggs)

Motor cortex, rabbit (Swadlow)

Barrel cortex, rat (Brecht)

Visual cortex, monkey/cat (Vinje & Gallant)

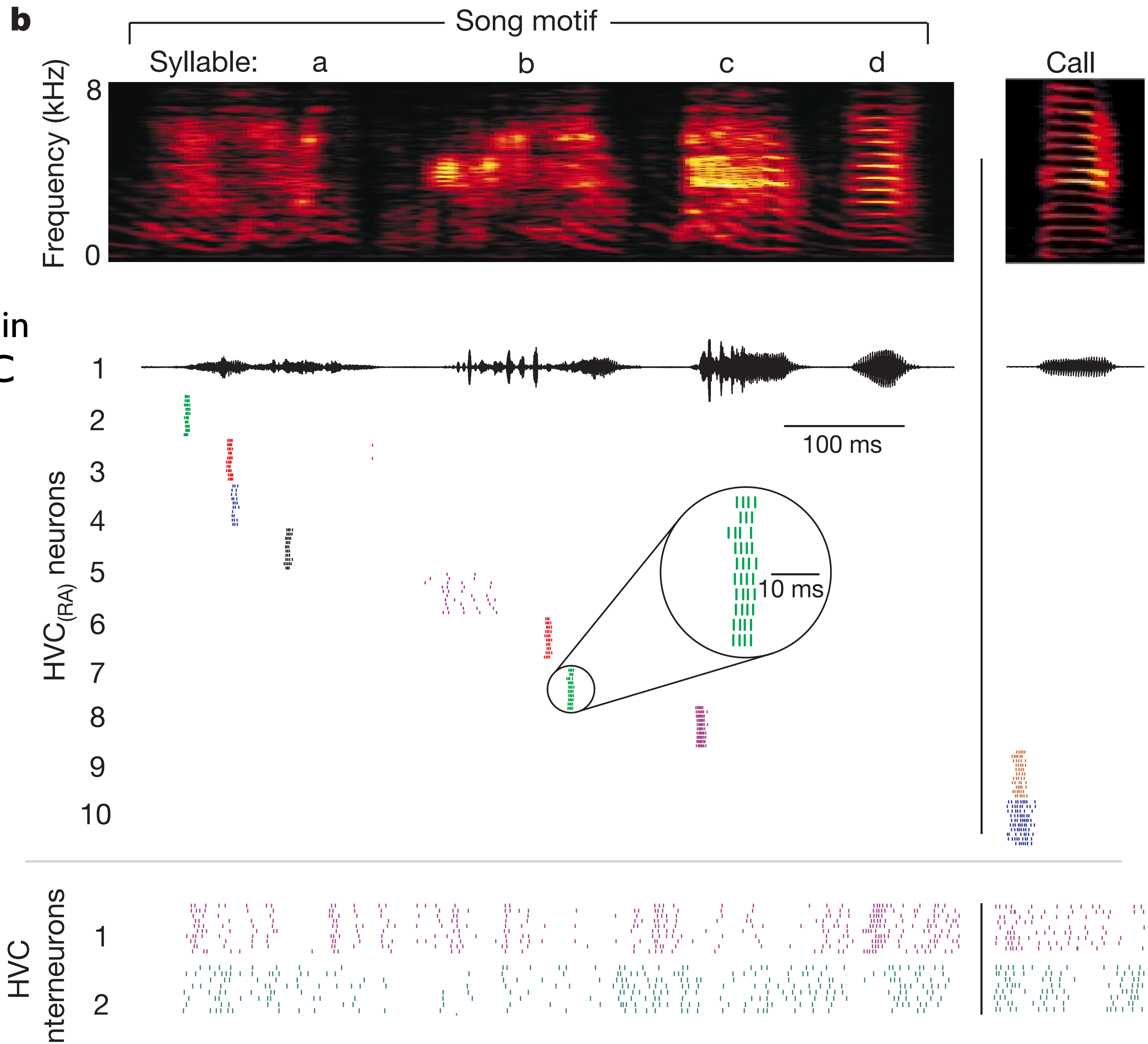
Visual cortex, cat (Gray; McCormick)

Inferotemporal cortex, human (Fried & Koch)

Olshausen BA, Field DJ (2004) Sparse coding of sensory inputs. *Current Opinion in Neurobiology, 14*, 481-487.

Sparse coding in songbird HVC

Hahnloser, Kozhevnikov & Fee (2002)



Open questions

- How to implement with spiking neurons?
  (See Zylberberg, Murphy & DeWeese, 2011)

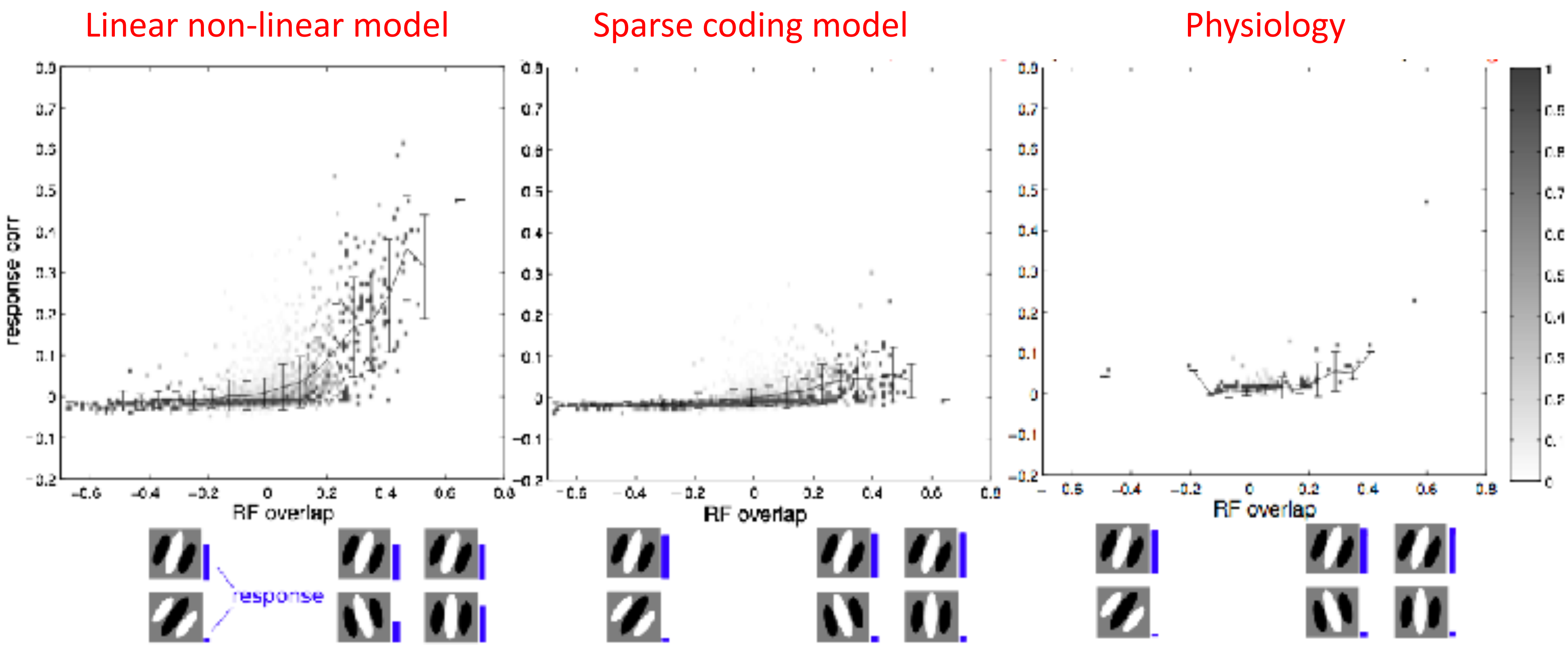
- How to implement with inhibitory interneurons?
  (Dale's law - see Zhu & Rozell, 2014)

- Are neural interactions consistent with sparse coding?

- How overcomplete?

- Time

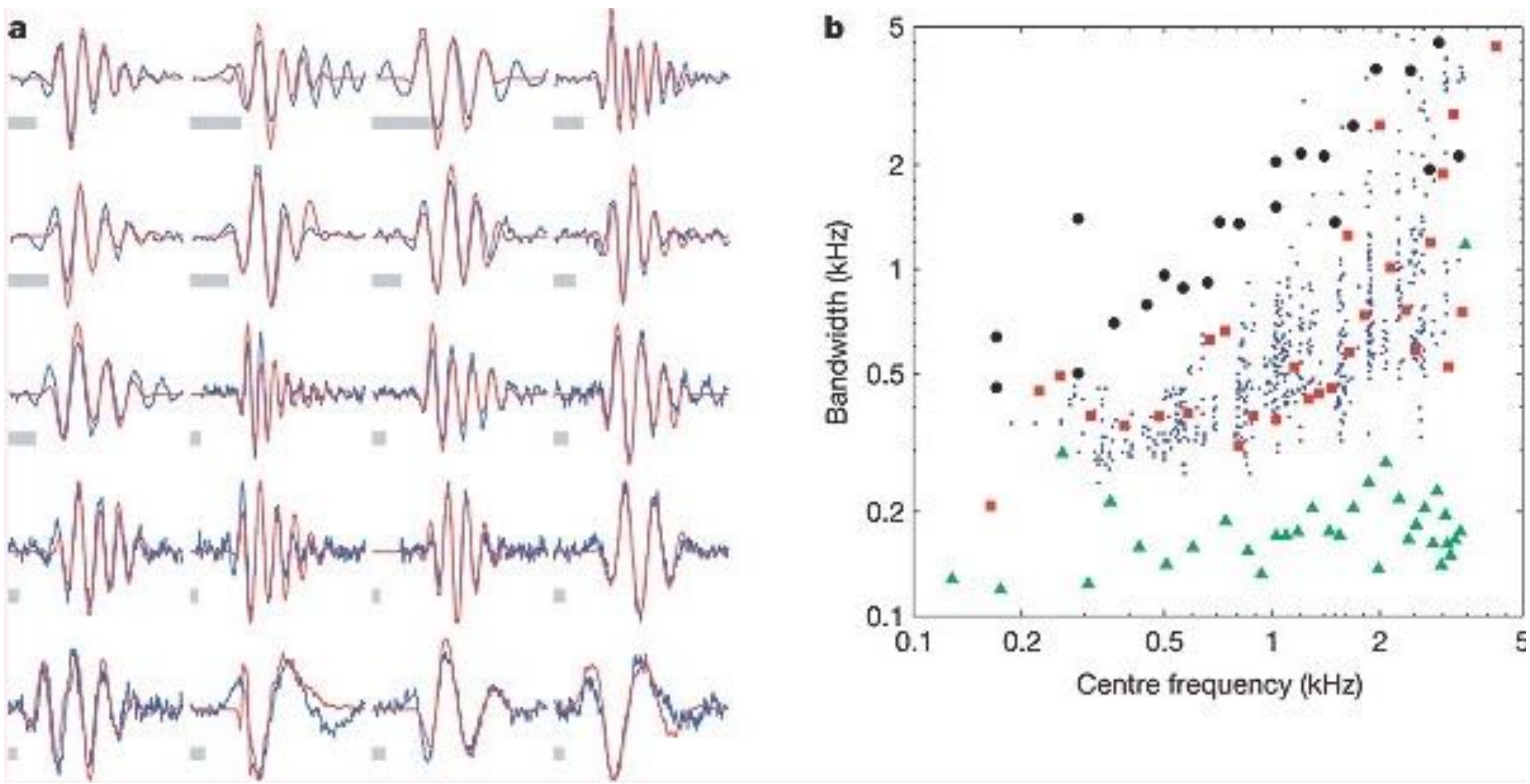
Active decorrelation



Sparse coding of natural sounds
(Smith & Lewicki 2006)

$$s(t) = \sum_i a_i(t) * \phi_i(t) + \nu(t)$$

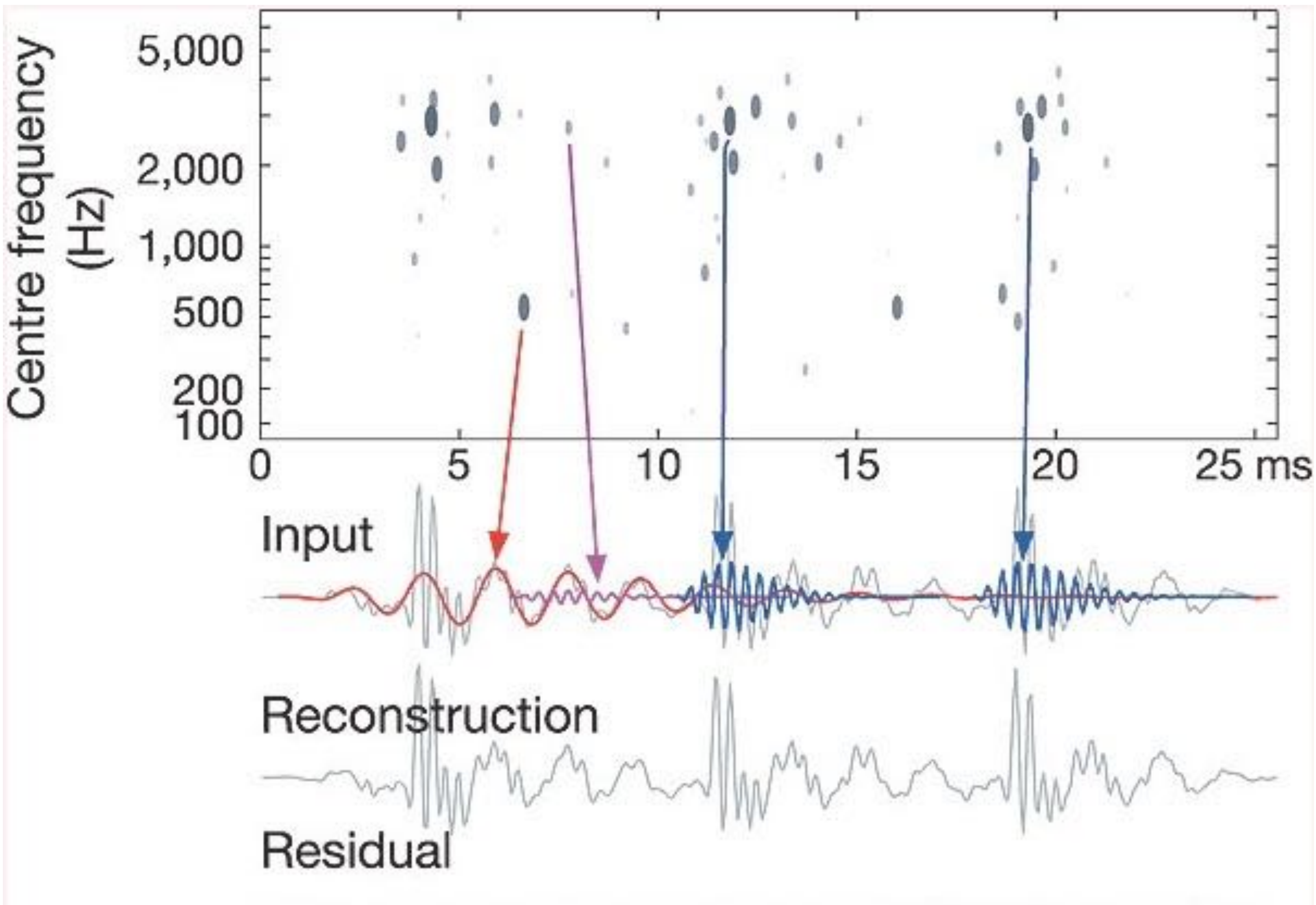
$\phi_i(t)$



Sparse coding of natural sounds
(Smith & Lewicki 2006)

$a_i(t)$

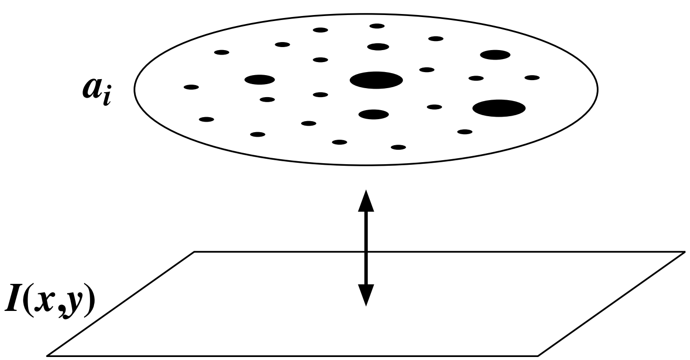
$s(t)$

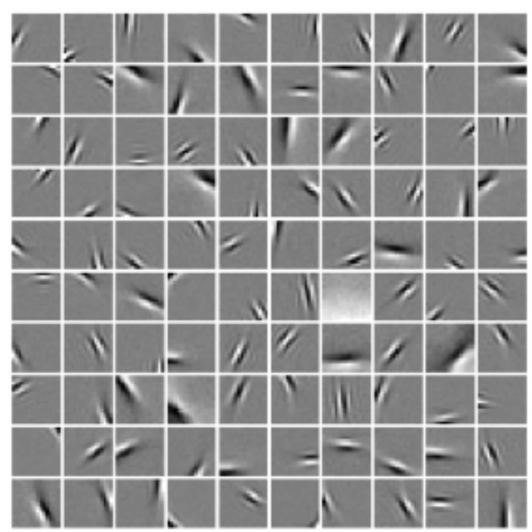


Applications of sparse coding

Theory of sparse coding (Barlow 1972) → Natural scene statistics (Field 1990's)

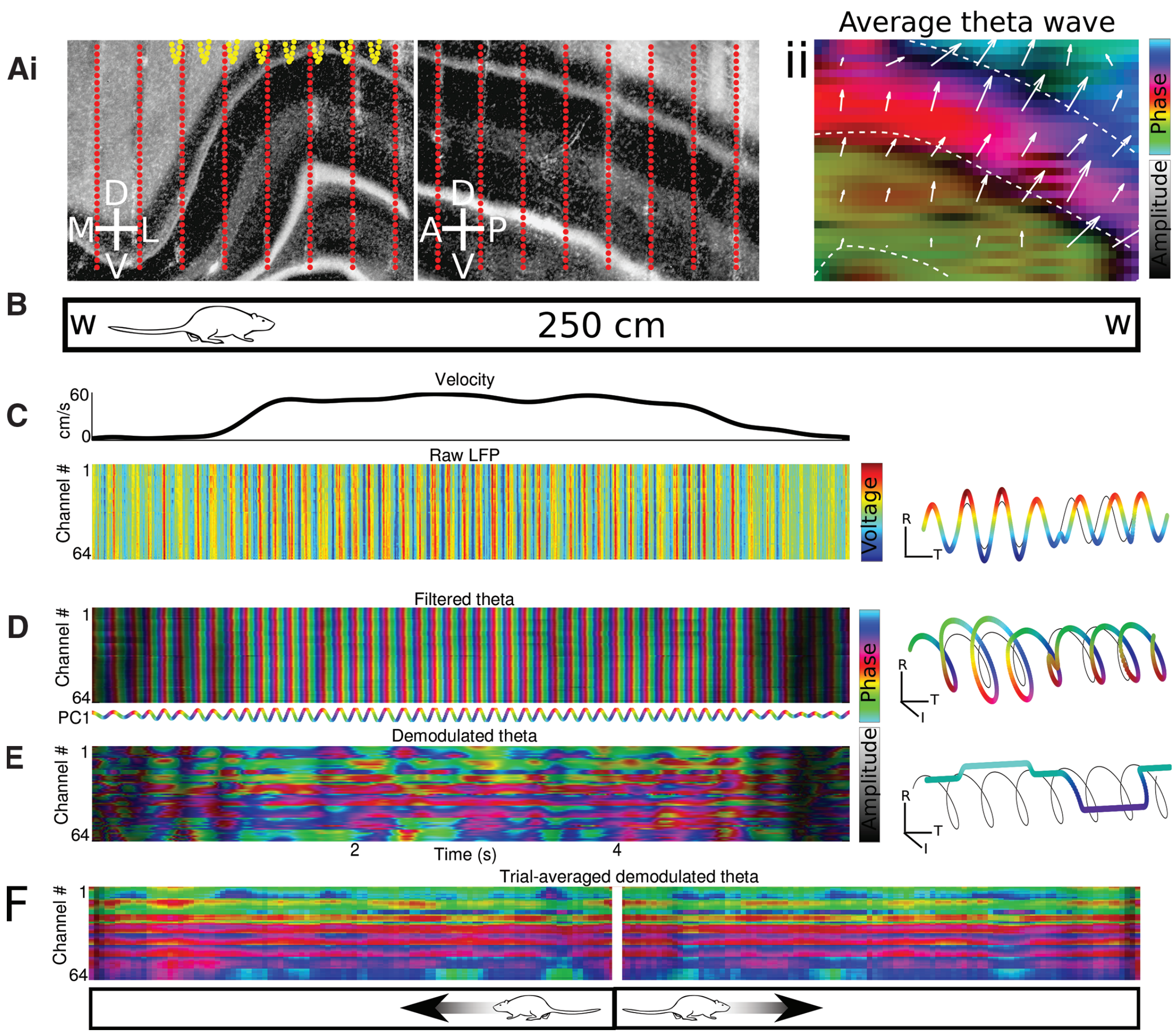
Natural scene statistics (Field 1990's) →
- Compressed sensing
- Signal/image restoration
- Computer vision
- Neural data analysis
- Deep learning



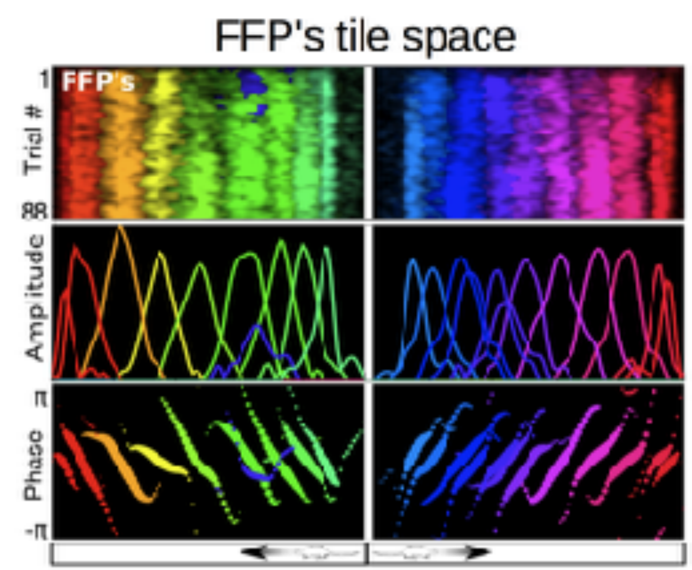


Sparse coding of demodulated LFP reveals ‘place cell’ components

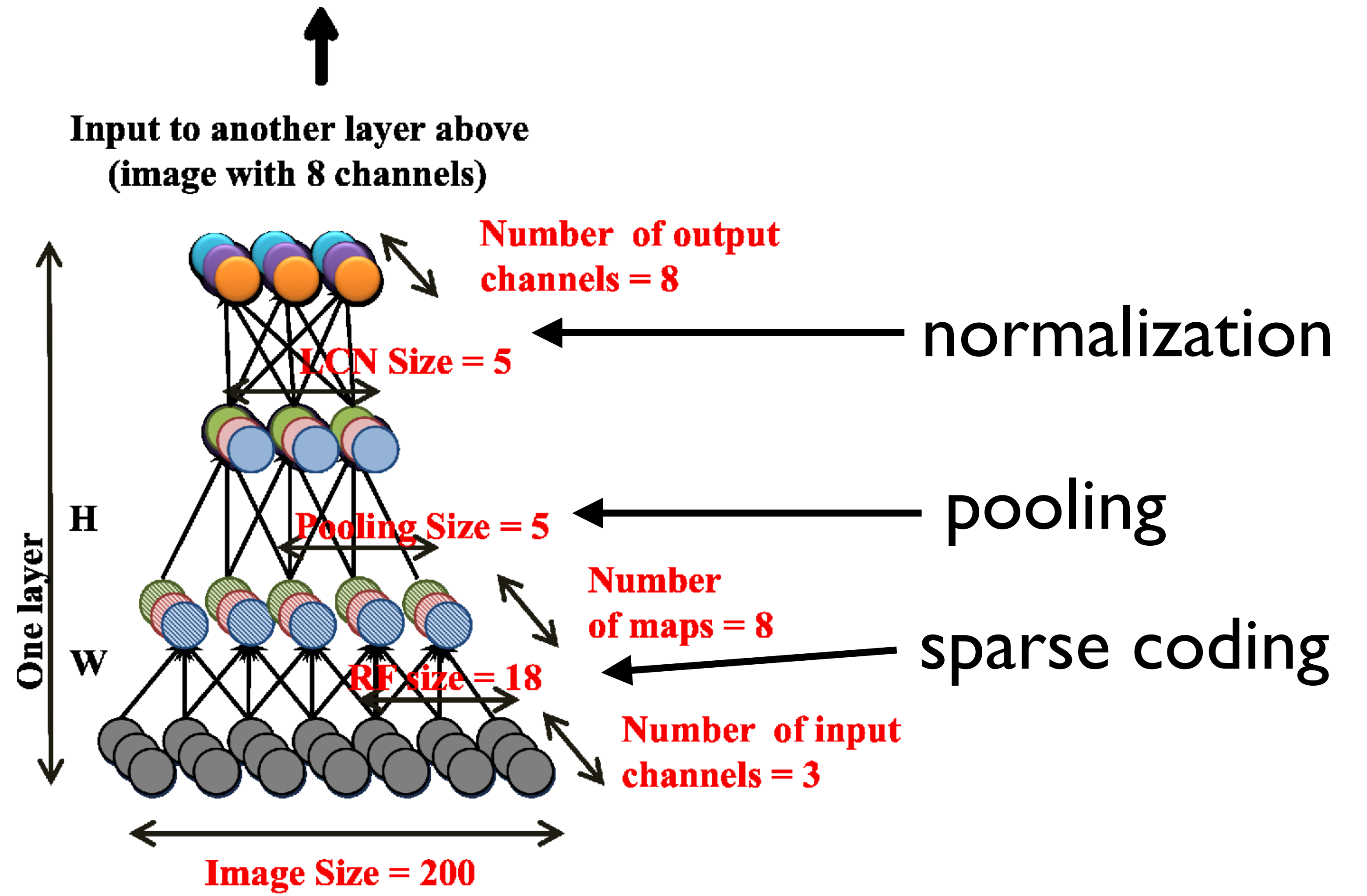
(Agarwal, Stevenson, Berényi, Mizuseki, Buzsáki & Sommer, 2014)



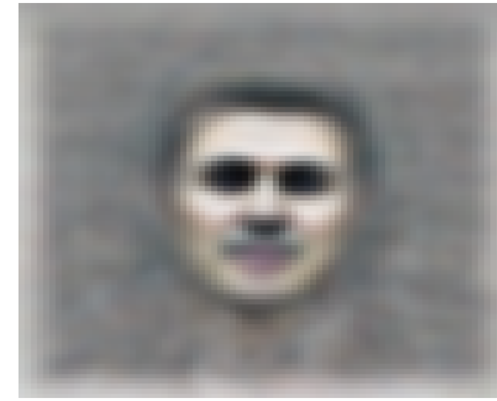
sparse coding

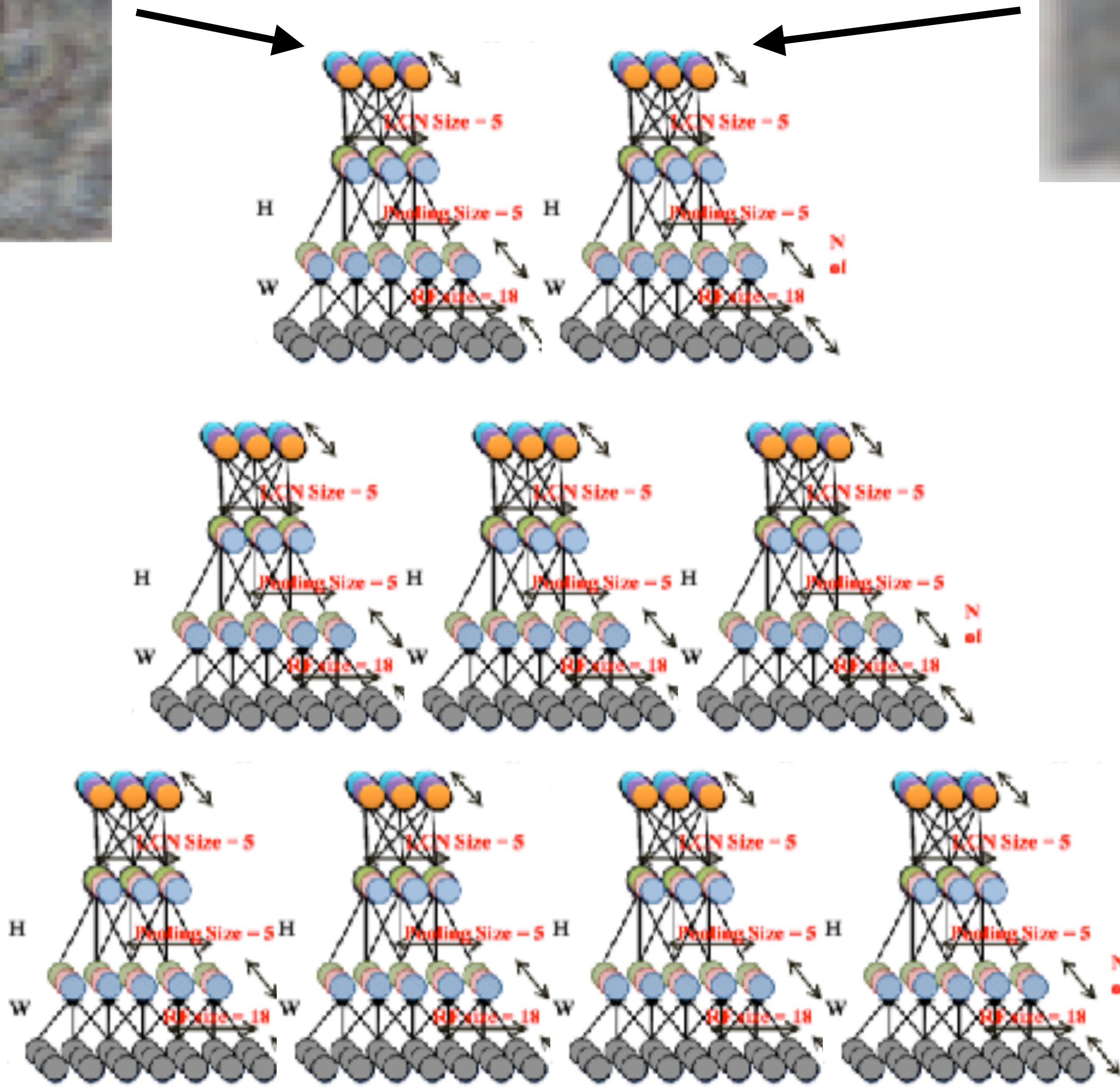


‘Google Brain’
(Quoc Le et al. 2012)

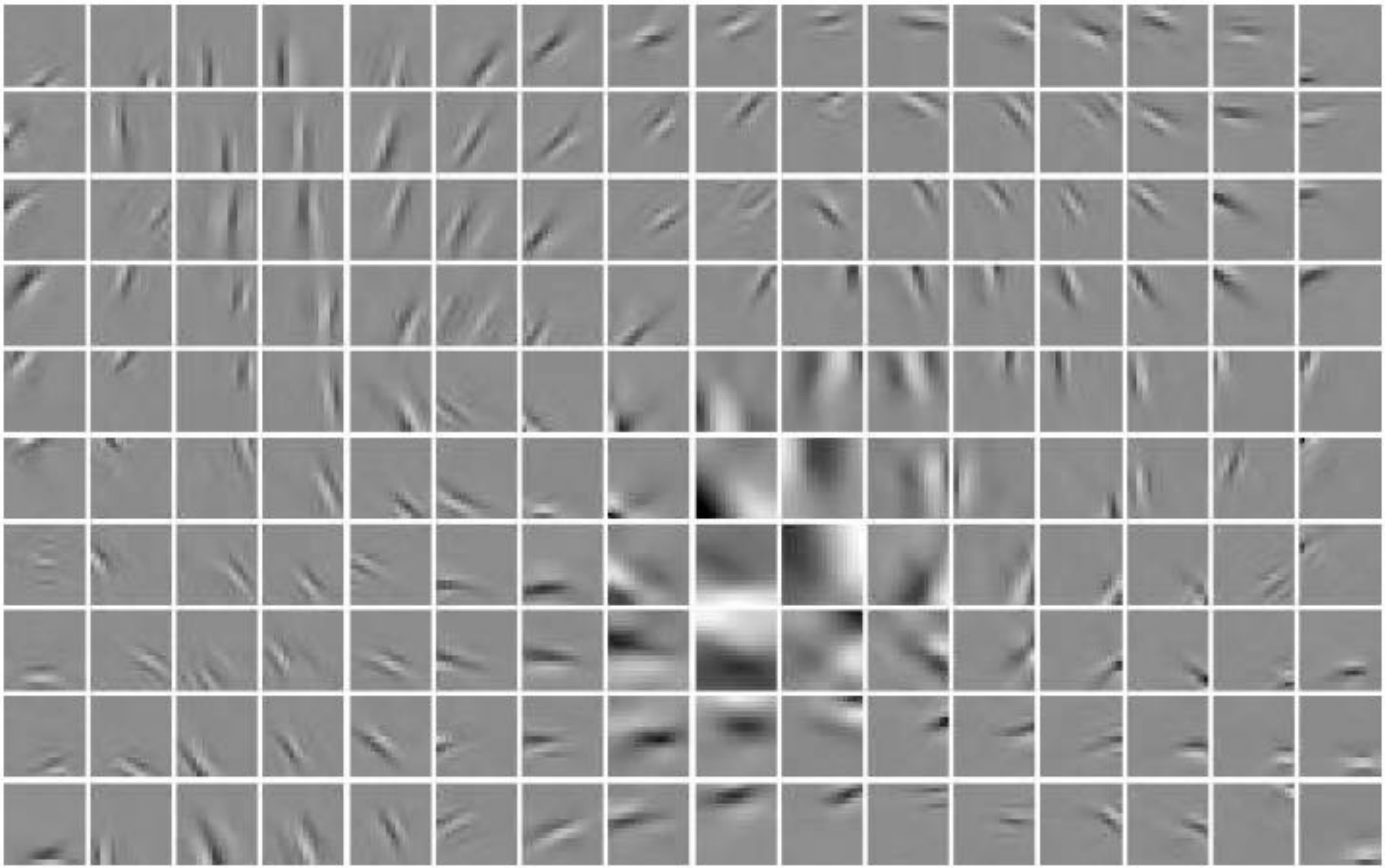


‘Google Brain’
(Quoc Le et al. 2012)

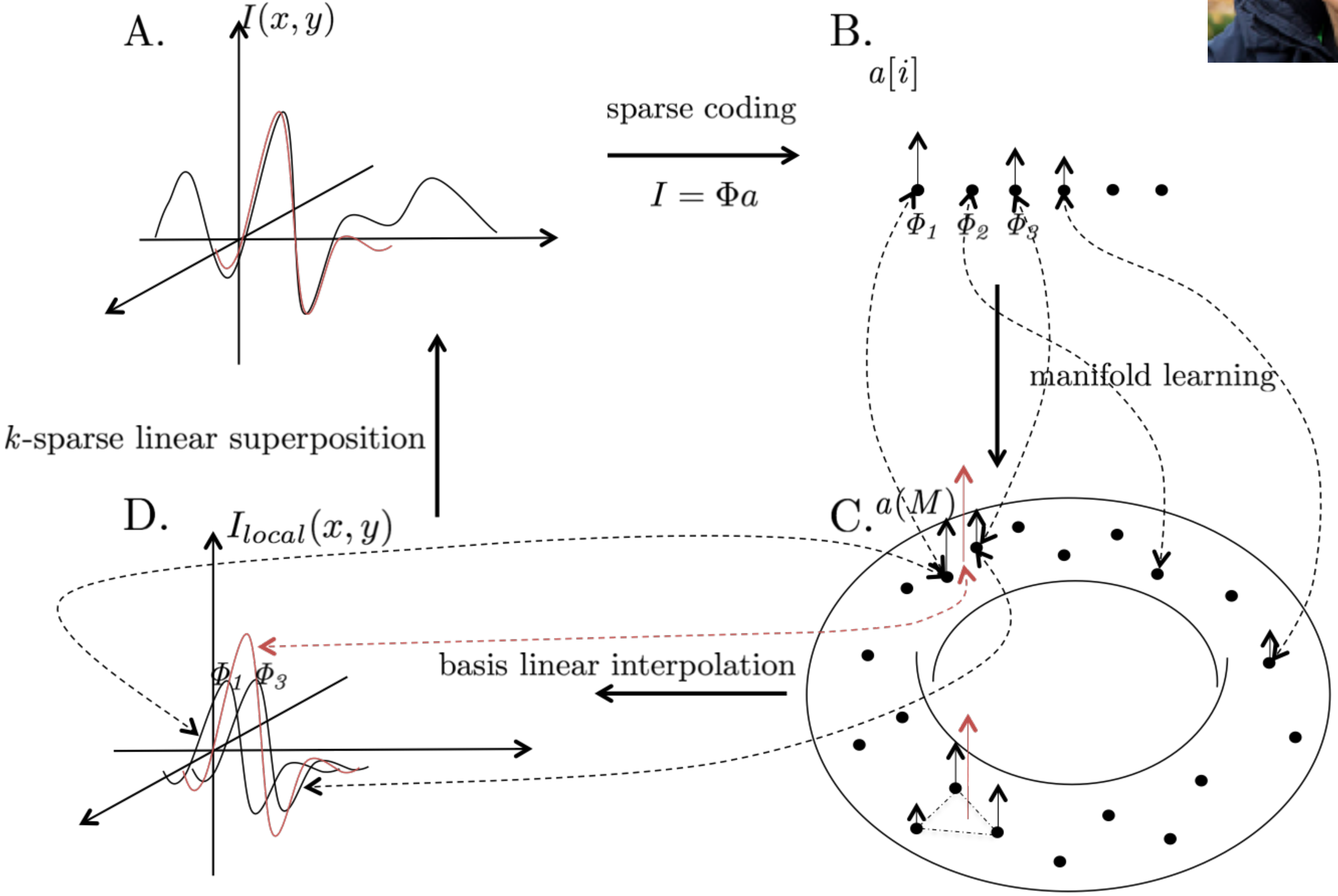




‘Topographic ICA’
(Hyvärinen & Hoyer 2001)



Sparse Manifold Transform
(Yubei Chen, Ph.D. thesis)



Learning to represent the function on the manifold

We desire: $$Pa_t \approx \frac{1}{2}Pa_{t-1} + \frac{1}{2}Pa_{t+1}$$

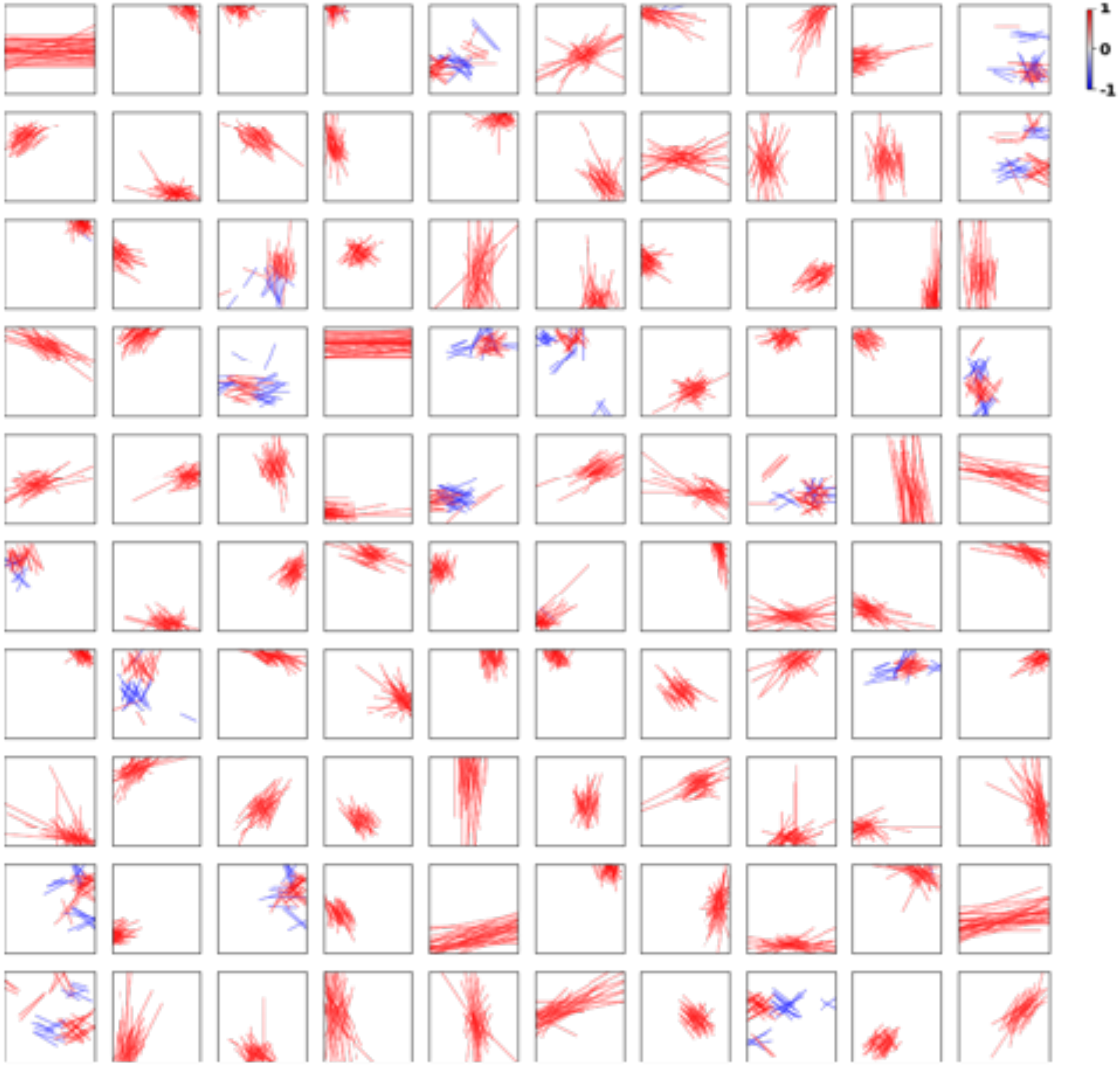
Objective function:

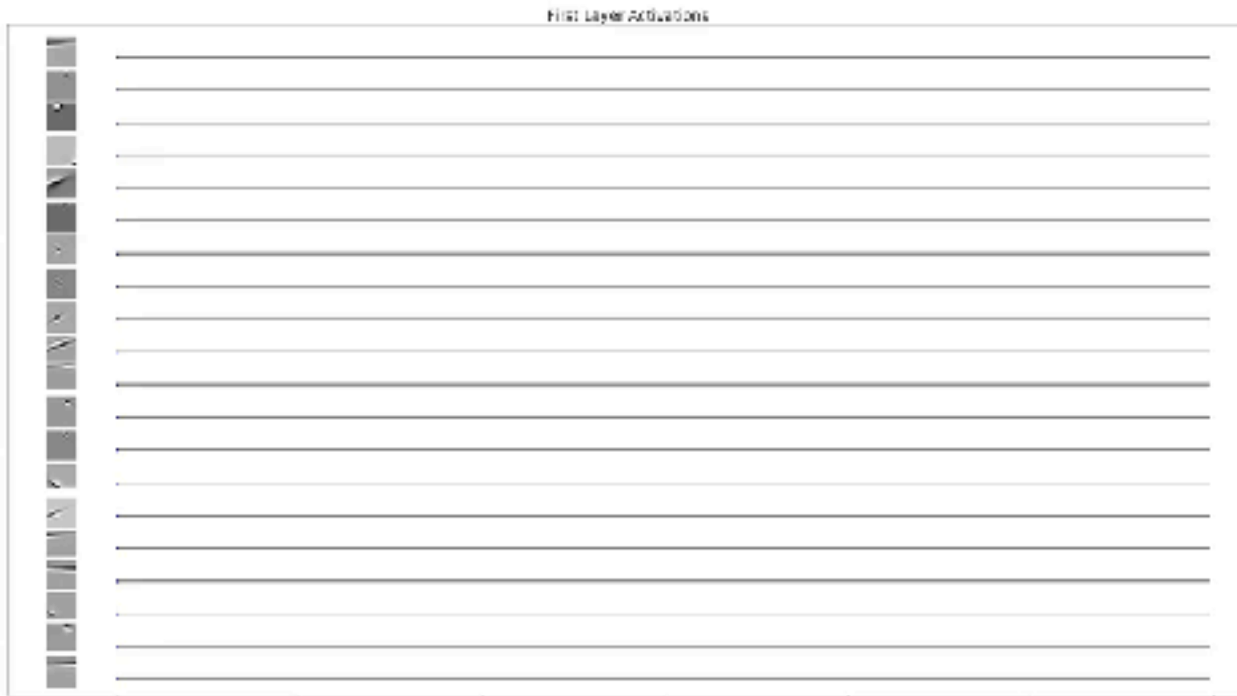
$$\min_P \; ||PAD||_F^2 - \gamma_1||PA||_F^2 + \gamma_2||PV||_1$$

s.t. $$PVP^T = \frac{1}{K}I$$

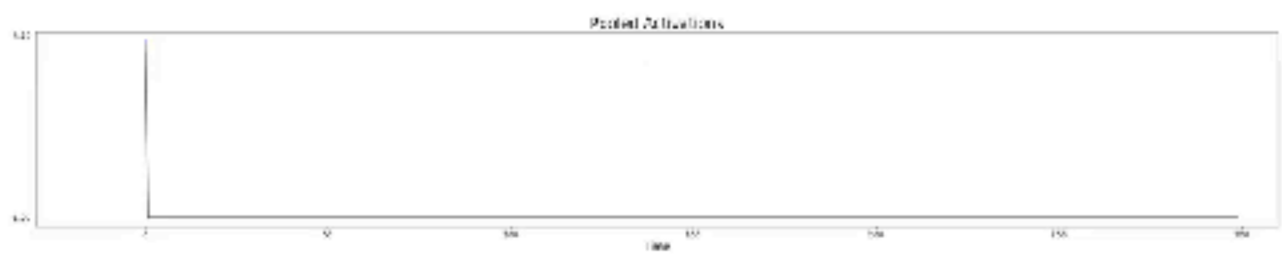
$$V = \mathrm{diag}(\bar{a})$$

$$D = \begin{bmatrix} 1 & -\frac{1}{2} & 0 & 0 & \dots & 0 \\ -\frac{1}{2} & 1 & -\frac{1}{2} & 0 & \dots & 0 \\ 0 & -\frac{1}{2} & 1 & -\frac{1}{2} & \dots & 0 \\ 0 & 0 & -\frac{1}{2} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & -\frac{1}{2} & 1 \end{bmatrix}$$

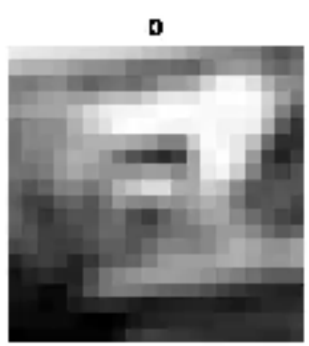

1
0
-1



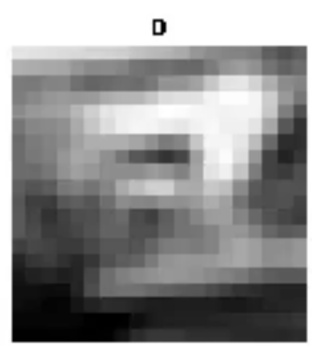
sparse
coefficients
$a_i(t)$



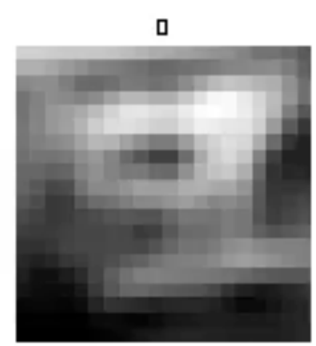
pooling
function
output



original



reconstr. $a_i(t)$



reconstr. pooling