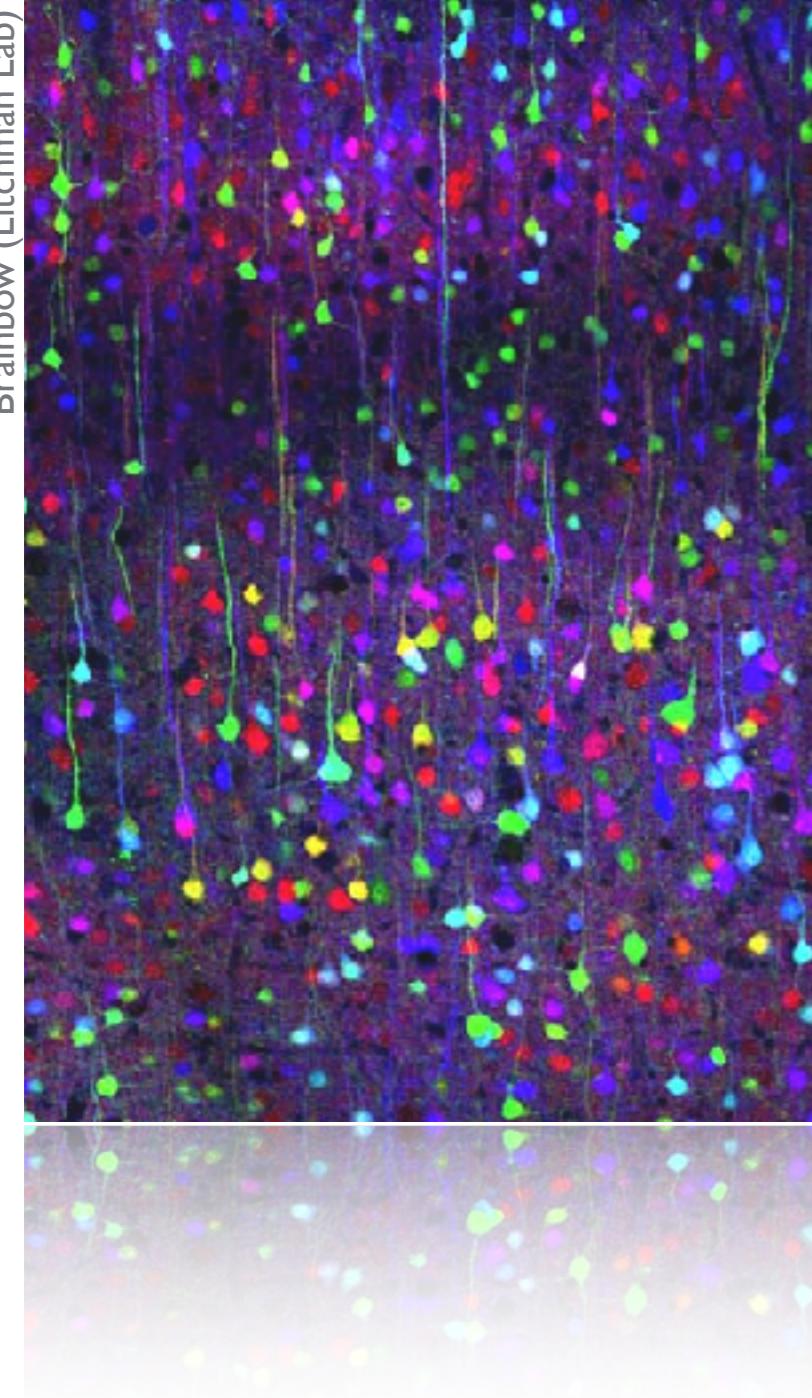


# Neural Information Processing 2018/2019



## Lecture 13 Neural circuits and learning: Unsupervised learning: Sparse coding and autoencoders

# **Extra lab session**

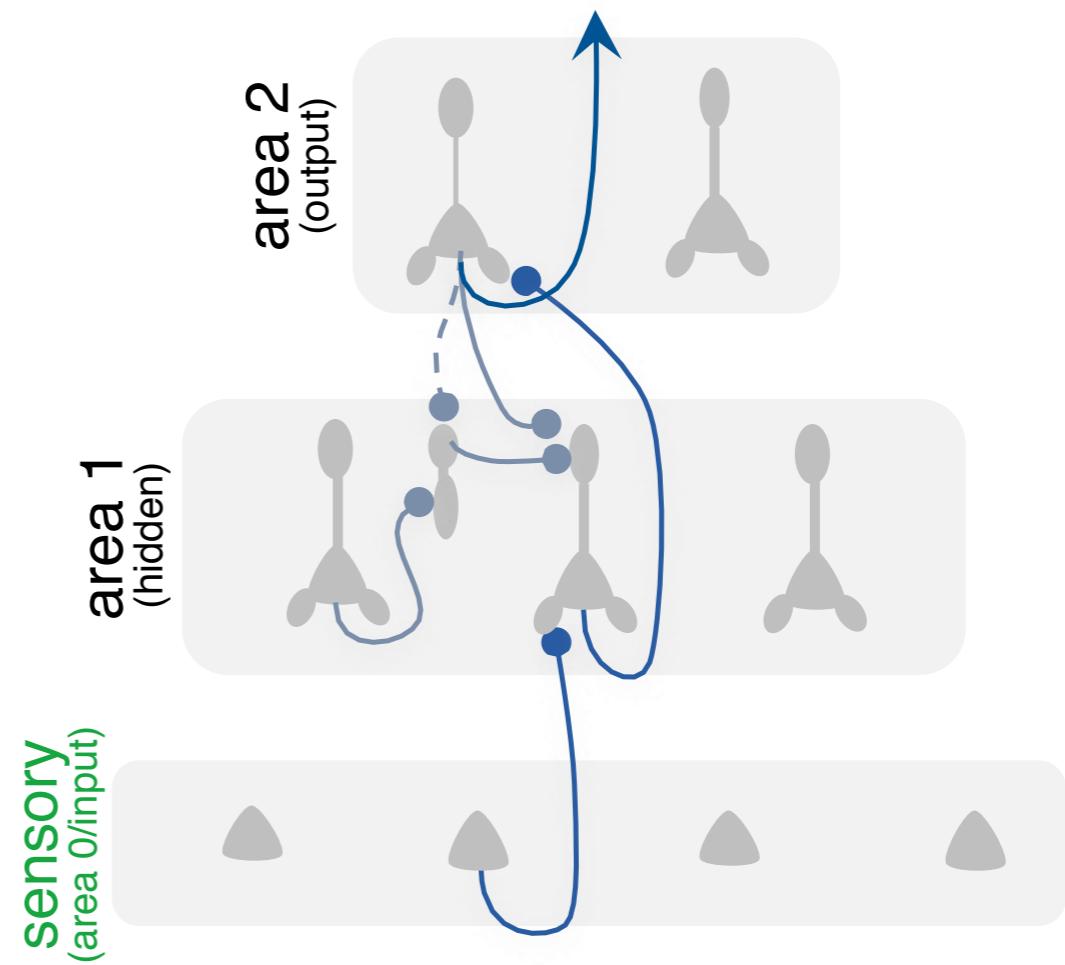
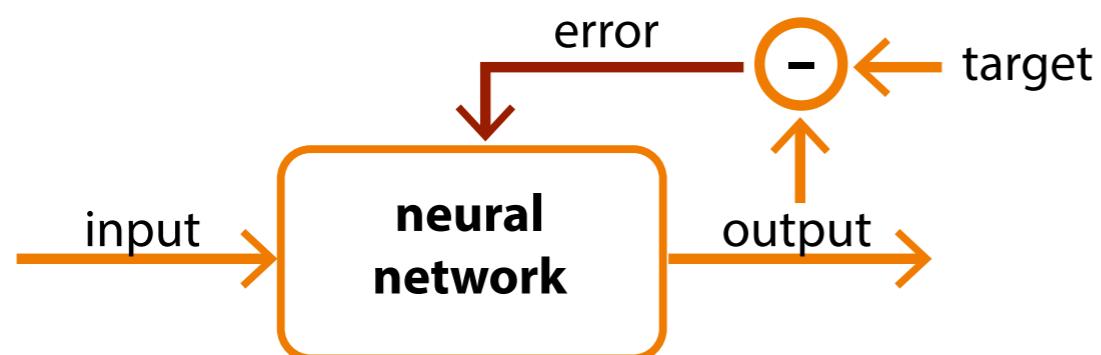
This Wednesday (12-2pm, 5th of December)!

Sign up for help with course work:

<https://doodle.com/poll/tbky2r8st4xxc3kb>

# Previously on Neural Information Processing...

**Supervised learning and backprop:**  
Relies on a teaching signal



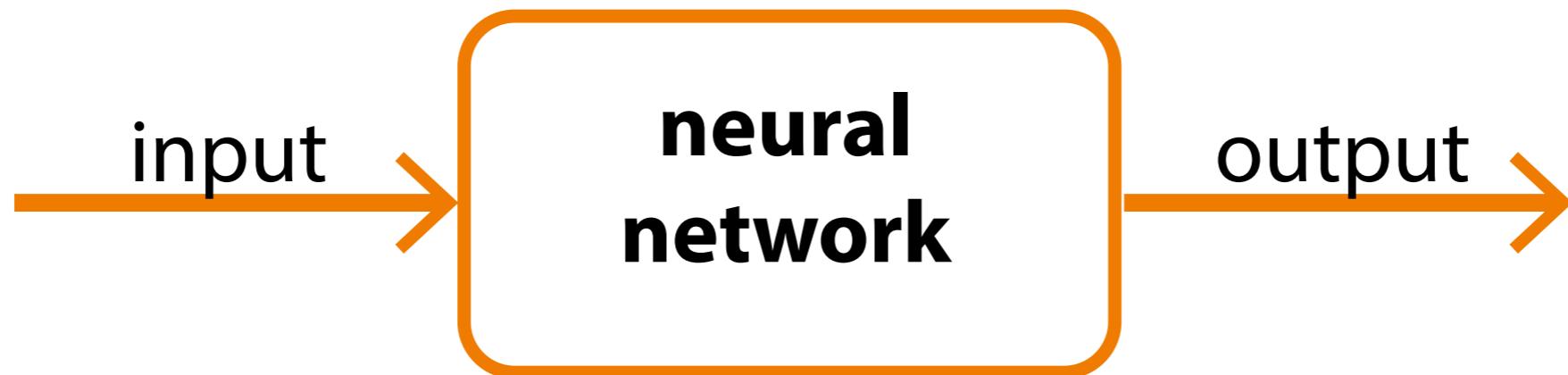
But, most sensory data is unsupervised  
Every few milliseconds we experience a new input!

<https://youtu.be/Plx9IjIKOsg>



But, most data animals experience is unsupervised

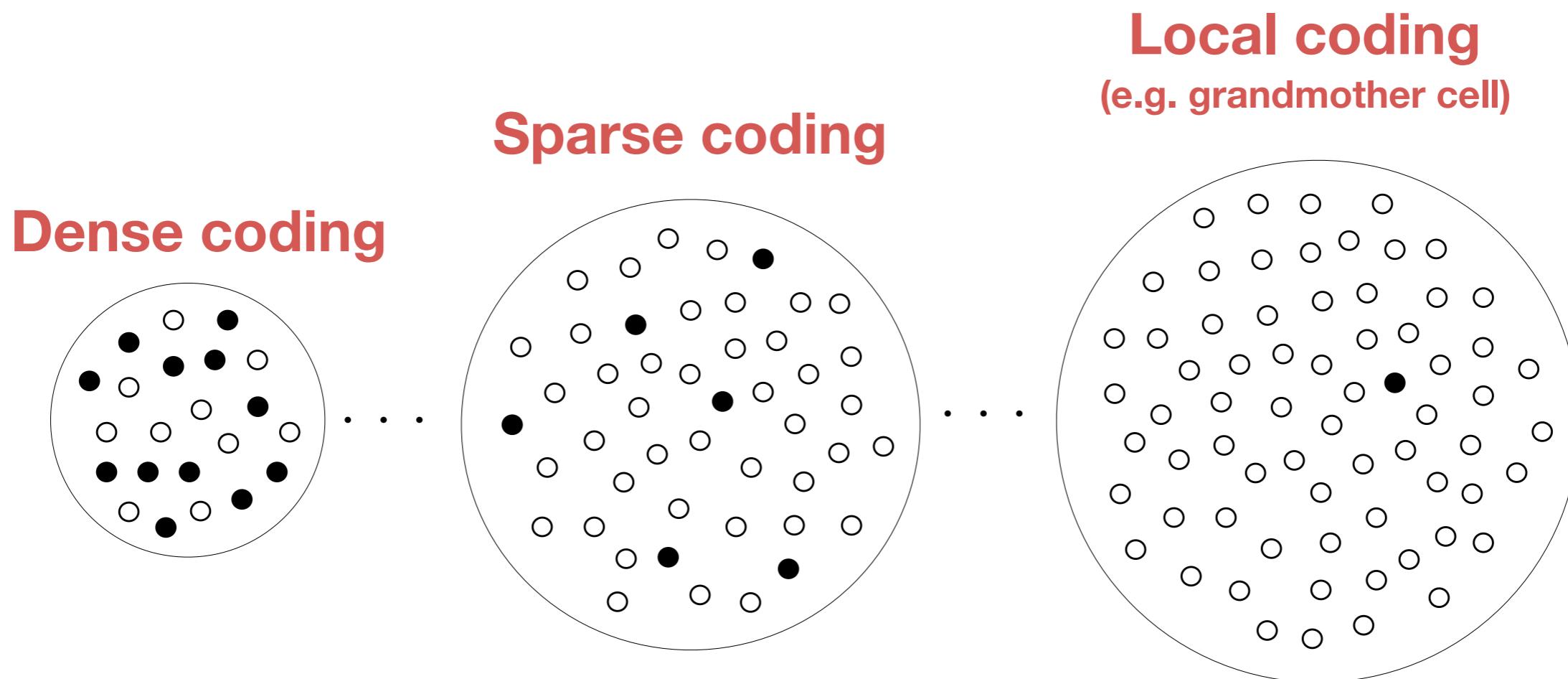
**Unsupervised Learning:**  
Extracts useful representations of input



# Outline

- 1. Sparse coding**
- 2. Sparse coding in the brain**
- 3. Autoencoders**

# Sparse coding



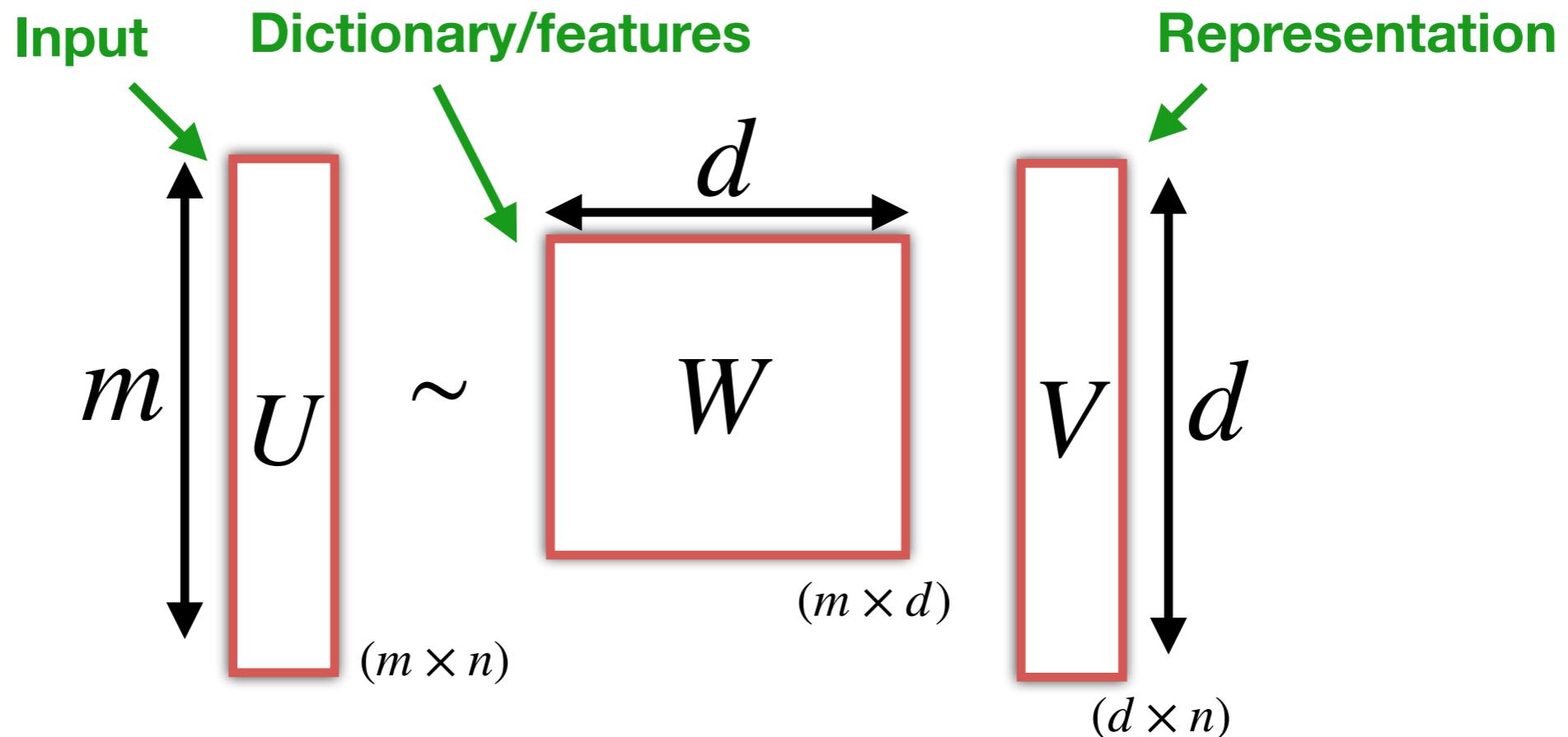
Couzinie-Devy, (2010)

# Sparse coding

- **Sparse coding** provides a framework for learning sparse representations of the input, typically through unsupervised learning.
- *Definition:* '**Sparse coding** is the representation of items by the strong activation of a relatively small set of neurons. For each stimulus, this is a different subset of all available neurons.'

Foldiak and Dominik, Scholarpedia (2008)

# Sparse coding



Key properties of sparse coding:

- **Overcomplete dictionary  $W$ :  $d > m$**  (i.e. more dictionary elements than input elements)
- **Sparseness**: # of zero elements in  $v \gg$  # non-zero elements in  $v$
- Overcomplete and sparse representation leads to richer features

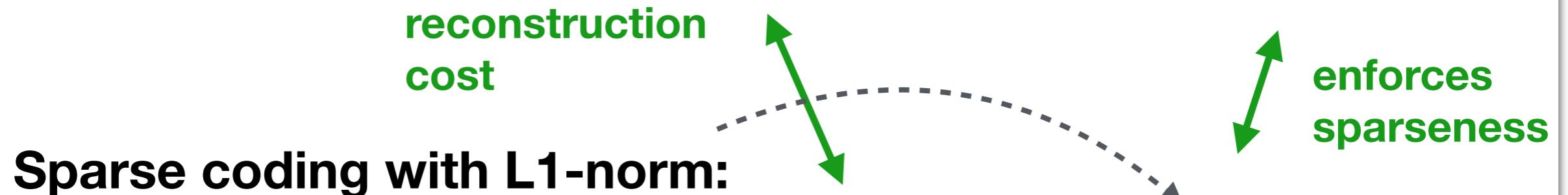
**Notes:**

$$\|X\|_2^2 = \sum_{i=1}^n |x_i|^2$$

$$\|X\|_1 = \sum_{i=1}^n |x_i|$$

## Sparse coding L1-norm

cost = [preserve information] +  $\lambda$ [sparseness]



$$\operatorname{argmin}_{W,V} = \|U - WV\|_2^2 + \lambda \|V\|_1$$

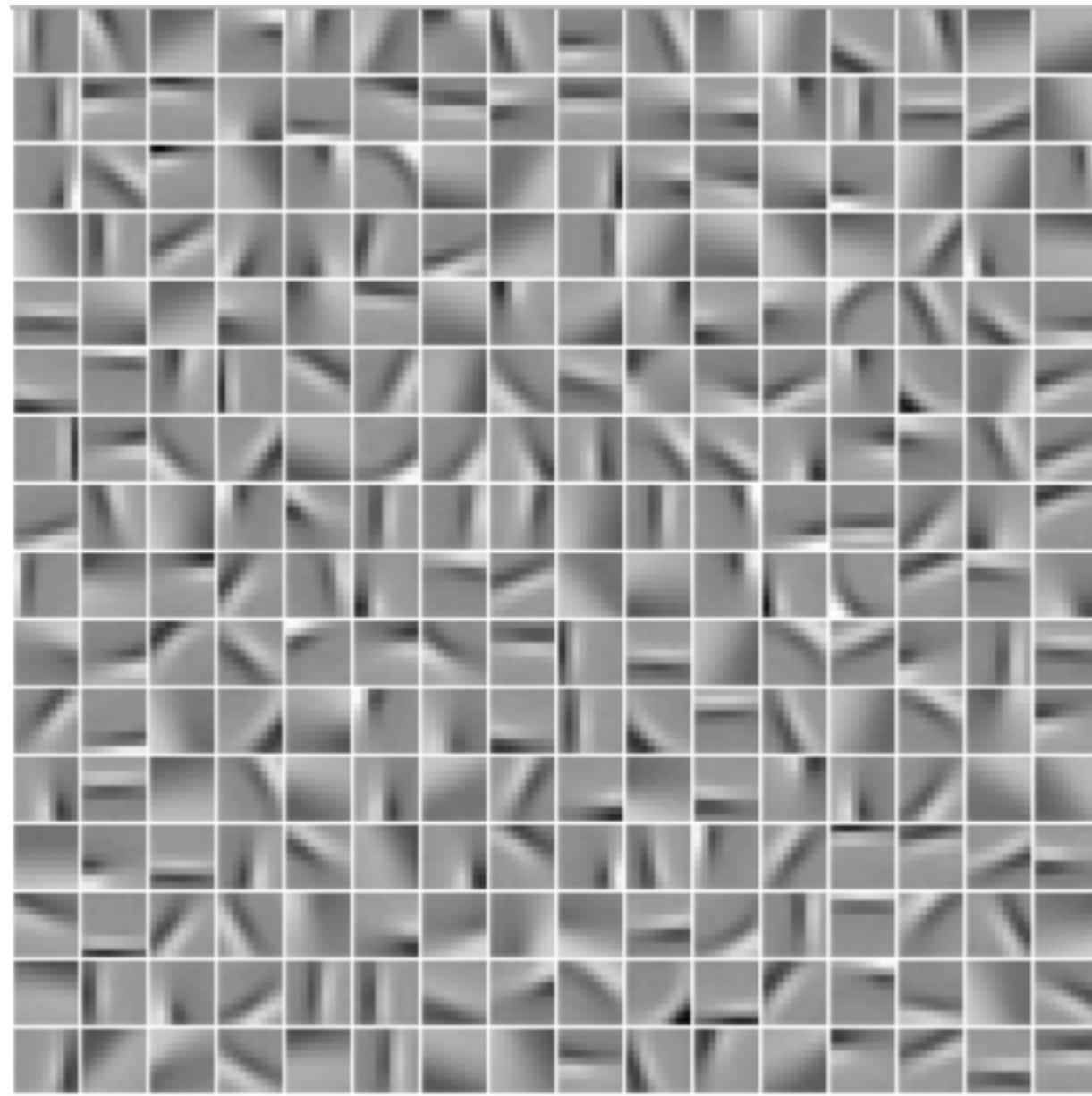
subject to  $\|W_i\|_2^2 \leq 1 \quad \forall i = 1, \dots, m$  ← avoids arbitrarily high  $W$

- To find a good sparse code we need to optimise both the dictionary,  $W$  and the representation  $V$
- This optimisation problem is convex in  $V$  and  $W$  separately, but not both. Solution: interleaved optimisation of  $V$  and  $W$
- Several methods exist to optimise  $V$  and  $W$  (we will discuss one)
- $\lambda$  controls the tradeoff between sparseness and reconstruction

# Sparse coding

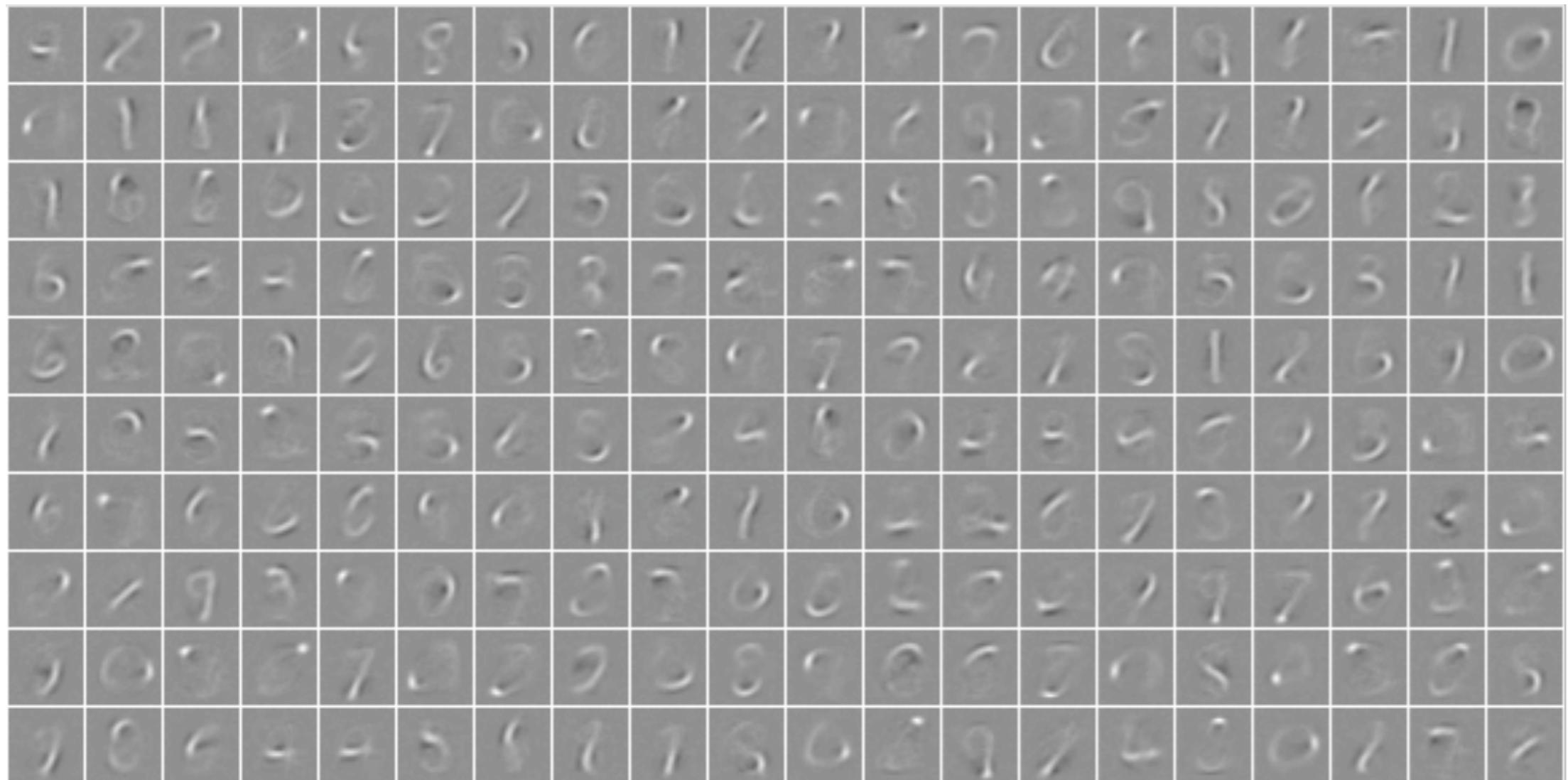
Input: Natural images

**Simple cell like features:**



# Sparse coding

Input: MNIST dataset (handwritten digits)



# Sparse coding

## Applications: denoising

Original



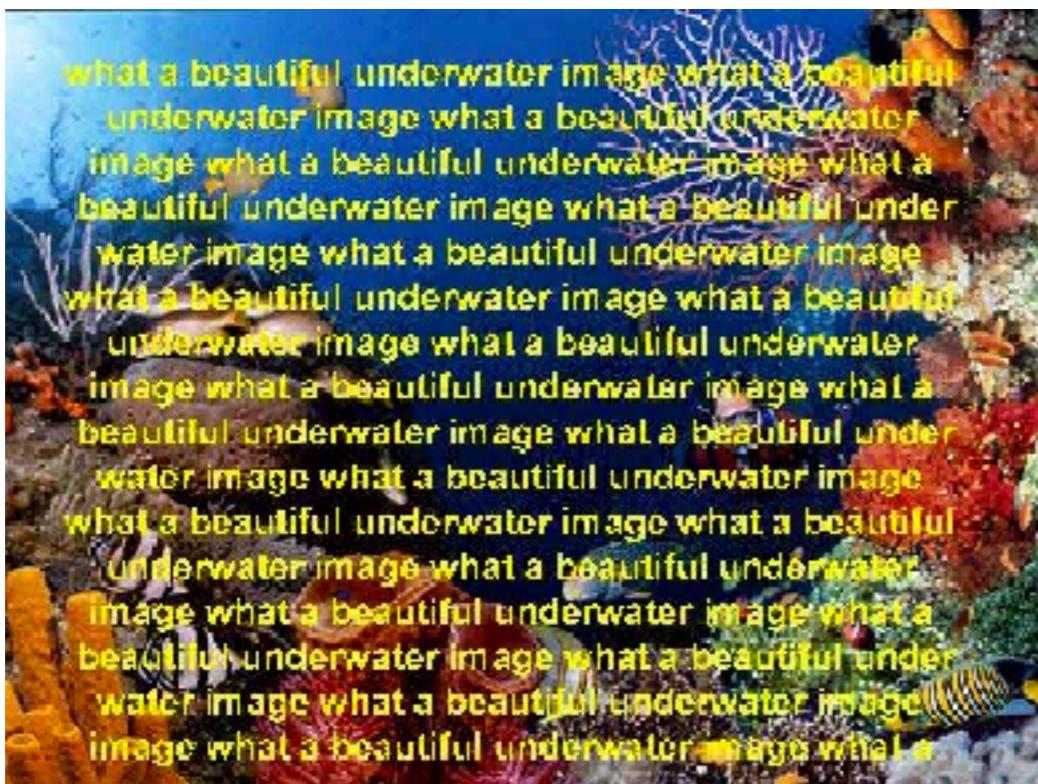
Sparse coding method



# Sparse coding

## Applications: image restoration

## Original



## Sparse coding method



Fig. 8. Underwater. Left: Image containing many high contrast edges, with text covering 16.19% of its area. Right: restored

Richard et al. VIIP 2001

# Sparse coding

## Applications: digital zooming

Original

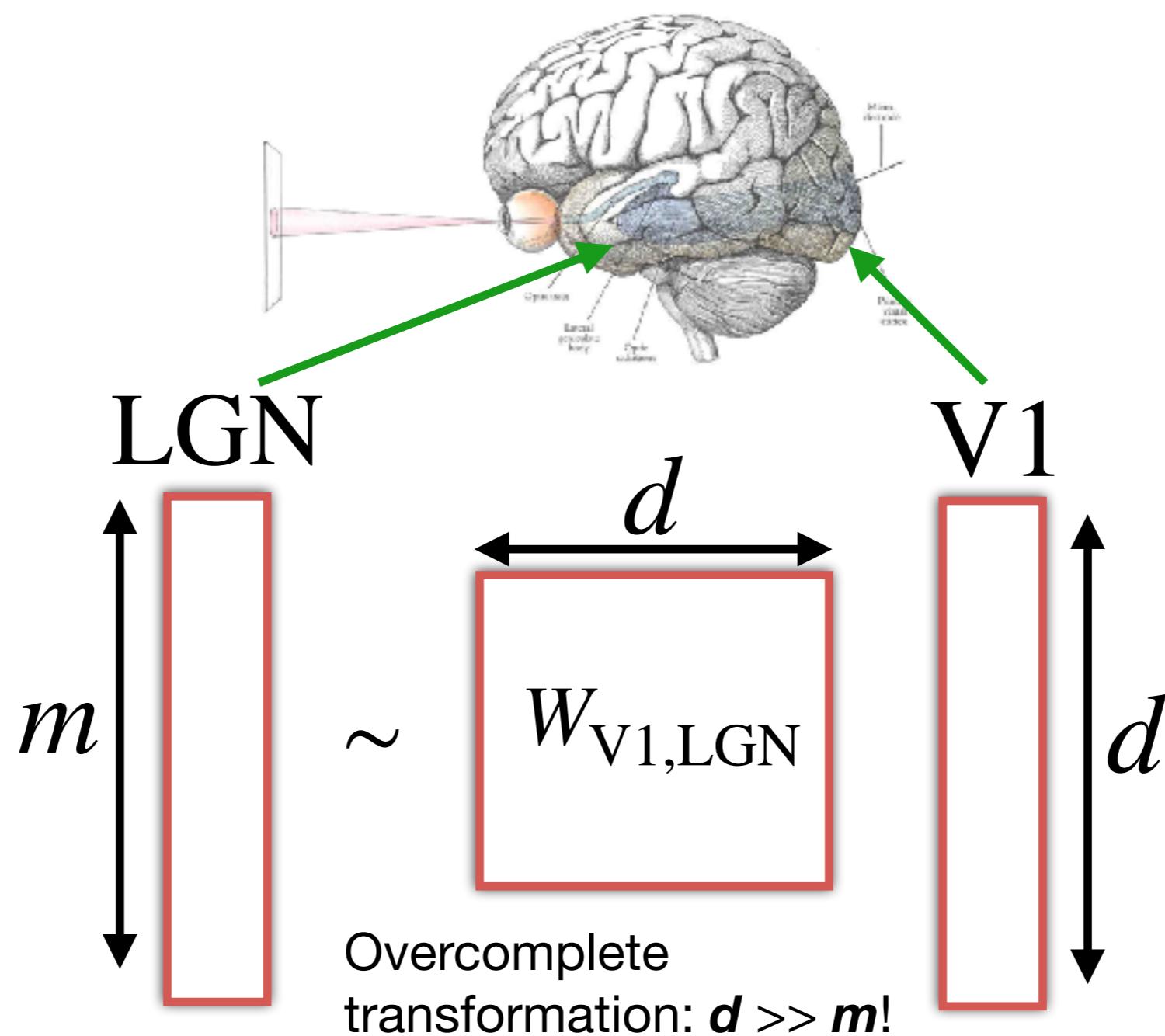


Sparse coding method



Couzinie-Devy, 2010

# Sparse coding in the visual system



# Sparse coding in the brain

Olshausen and Field 1996

New cost function:

$$\text{argmin}_{W,V} = \underbrace{\|U - WV\|_2^2 + \lambda \sum S(v_i)}_{\text{Cost}} \quad \text{new sparseness cost}$$
$$S(v_i) = \log(1 + v_i^2)$$

Notation in Olshausen and Field paper:

$$\text{argmin}_{\Phi,A} = \|I - \Phi A\|_2^2 + \lambda \sum_i C(a_i)$$

Olshausen and Field, Nature (1996)

# Sparse coding in the brain

Olshausen and Field 1996

Compute  $V$  via gradient descent, and get dynamics for  $v$ :

$$\dot{v}_i = - \frac{\partial \text{Cost}}{\partial v_i}$$
$$= w_i U - \sum_{j \neq i} G_{ij} v_j - f_\lambda(v_i)$$

where

$$G_{ij} = w_i w_j$$

$$f_\lambda(v_i) = v_i + \lambda S'(v_i)$$

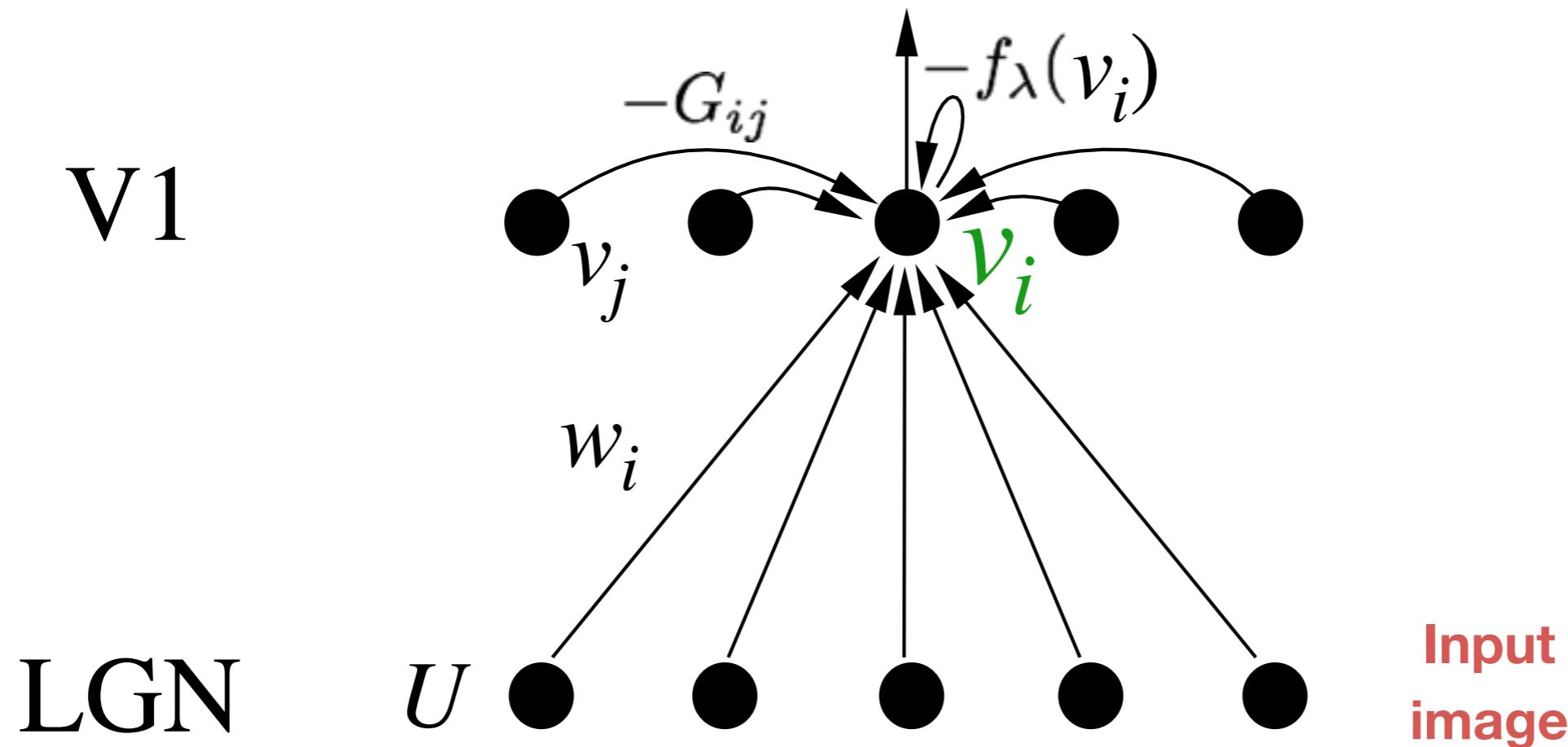
Olshausen and Field, Nature (1996)

# Sparse coding in the brain

Olshausen and Field 1996

Potential network implementation of:

$$\dot{v}_i = w_i U - \sum_{j \neq i} G_{ij} v_j - f_\lambda(v_i)$$



Olshausen and Field, Nature (1996)

# Sparse coding in the brain

Olshausen and Field 1996

With  $v$  we obtain the estimate, on a second phase we need to optimise the weights as:

$$\begin{aligned}\Delta w_i &= -\eta \frac{\partial \text{Cost}}{\partial w_i} \\ &= \eta < [U - \hat{U}]v_i > \\ &= \eta < [U - \sum_i^m w_i v_i]v_i >\end{aligned}$$

reconstructed image

Learning rule is local, and hebbian:

$$\Delta w \propto \underbrace{UV}_{\text{Hebbian term}} - \underbrace{WV^2}_{\text{post. alone}}$$

Note: Hebbian learning rules depend on presynaptic and postsynaptic activity, e.g.:

$$\Delta w = UV$$

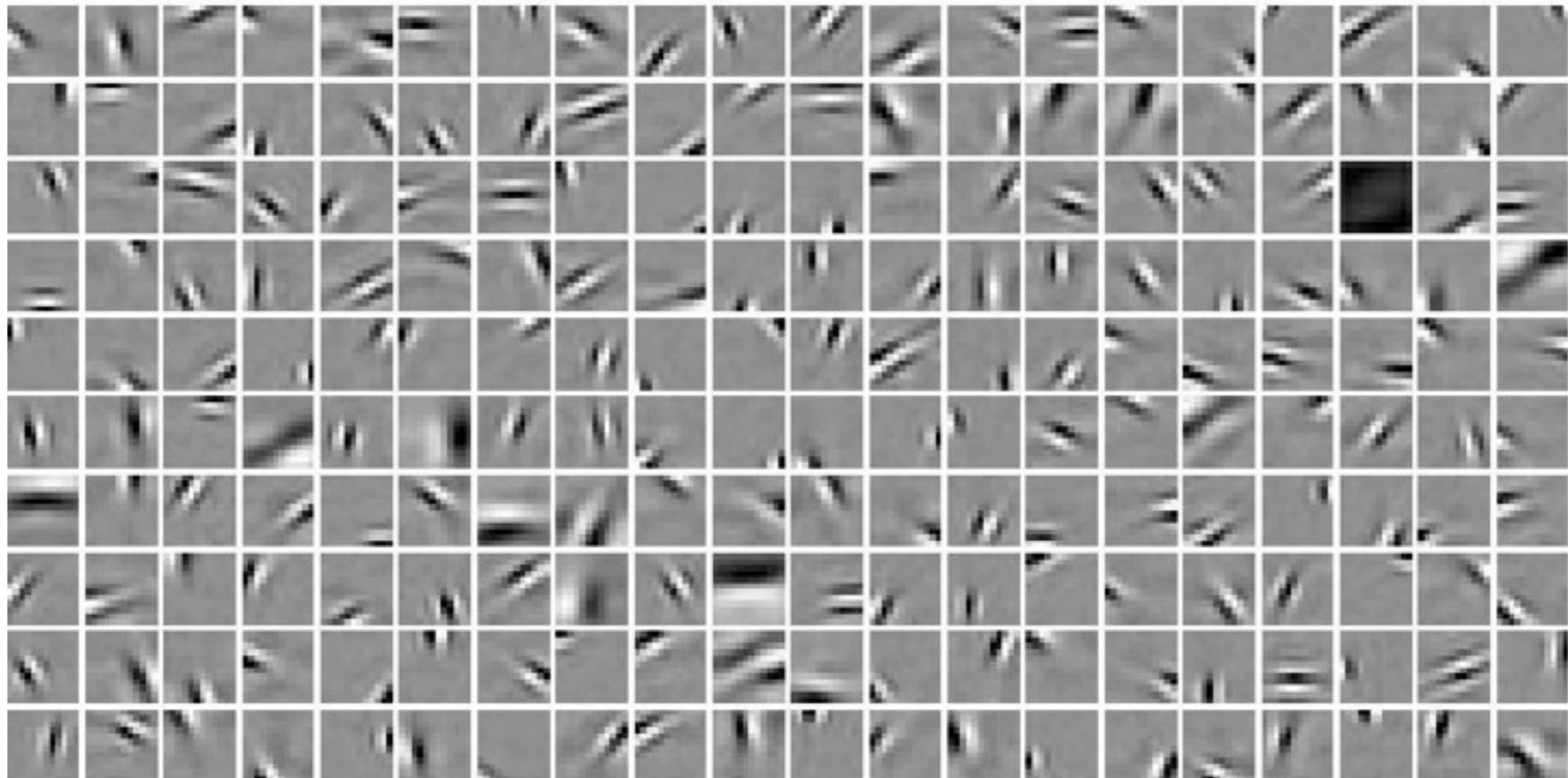
Hebbian plasticity: Cells that fire together, wire together

Olshausen and Field, Nature (1996)

# Sparse coding in the brain

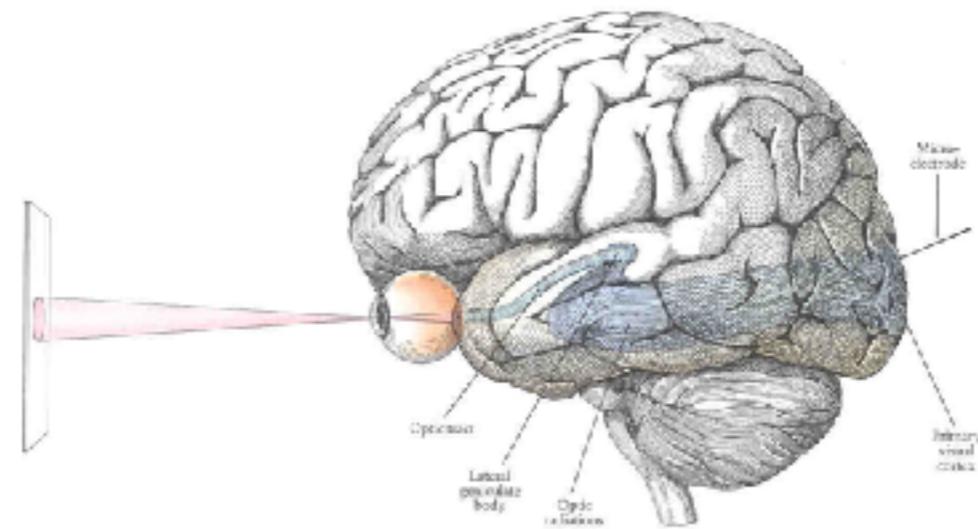
Olshausen and Field 1996

Learned dictionaries/weights (simple cell-like):



Olshausen and Field, Nature (1996)

## Group discussion groups of 2-3 (5min)



Can you think of advantages of using **sparse coding** in the brain?

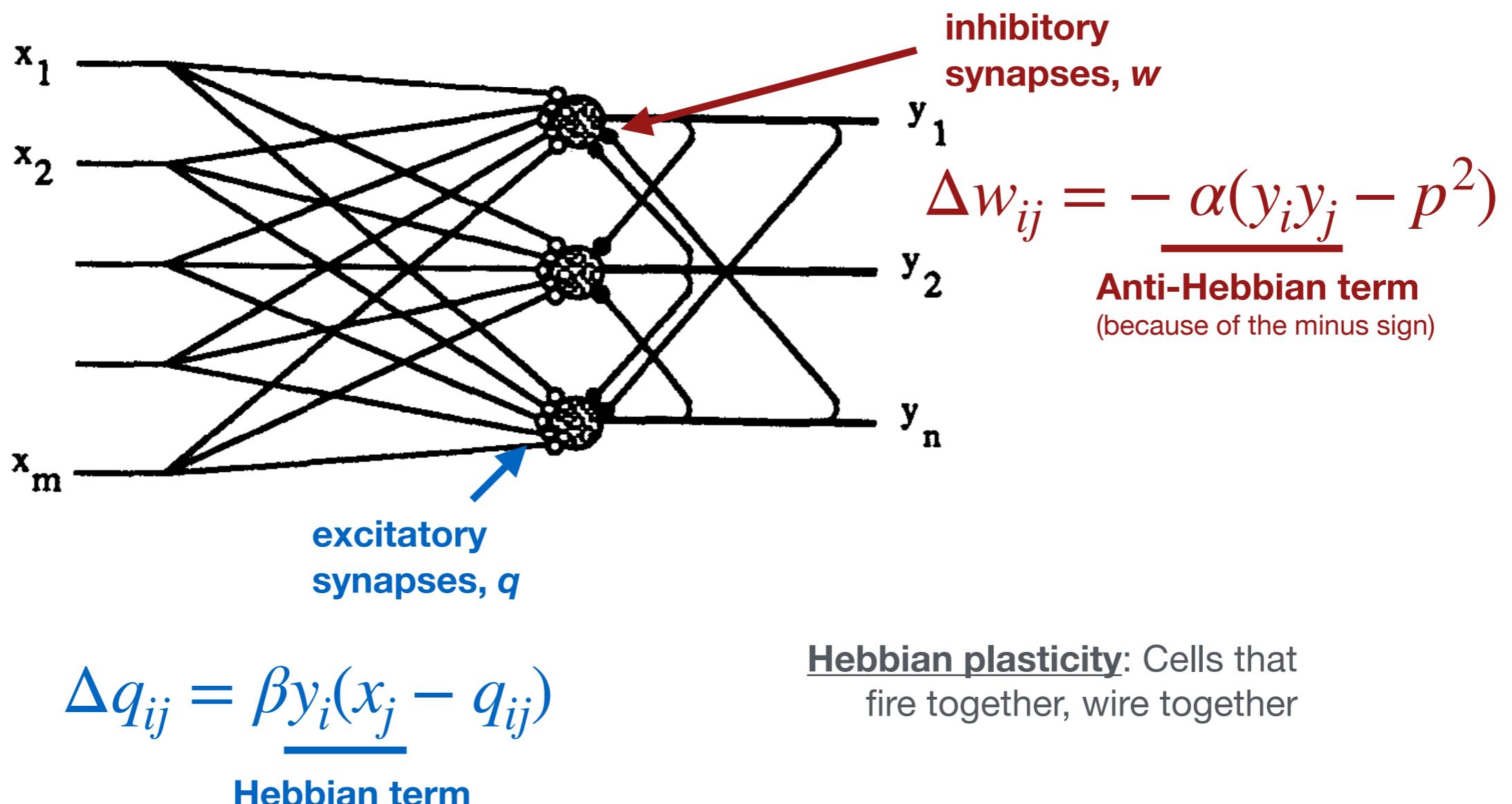
## Group discussion groups of 2-3 (5min)

Can you think of other advantages of using  
**sparse coding** in the brain?

- **Good representations** that can be used for decision making/discrimination
- Sparse coding is **energy efficient** (only a few neurons active at a time)
- **More robust** to perturbations than purely localist codes

# Sparse coding

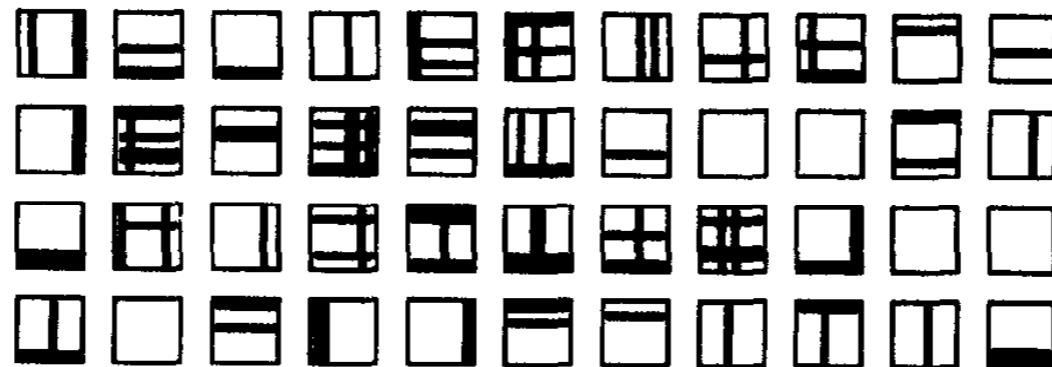
## Through Hebbian learning



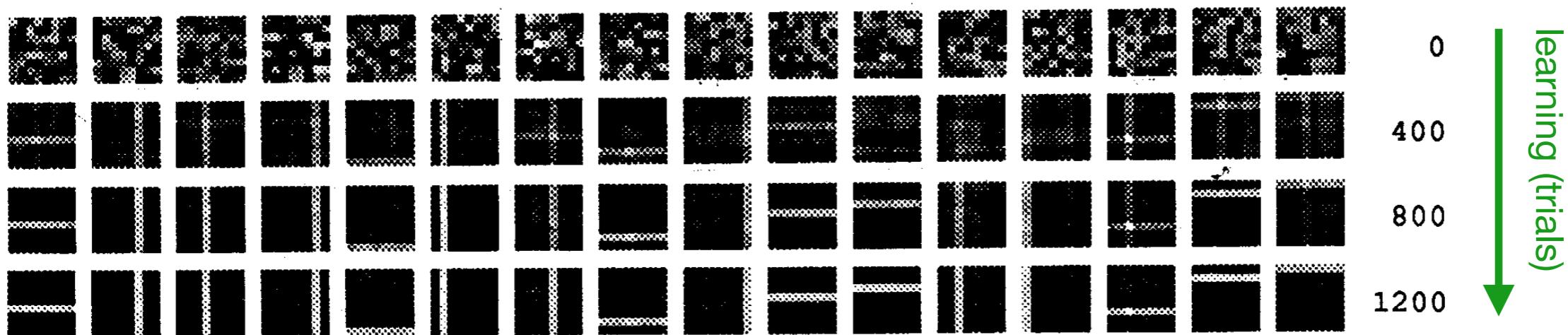
Foldiak, Biological Cybernetics (1990)

# Sparse coding Through Hebbian learning

**Input defined as random patterns of vertical and horizontal lines:**



**Learned features:**



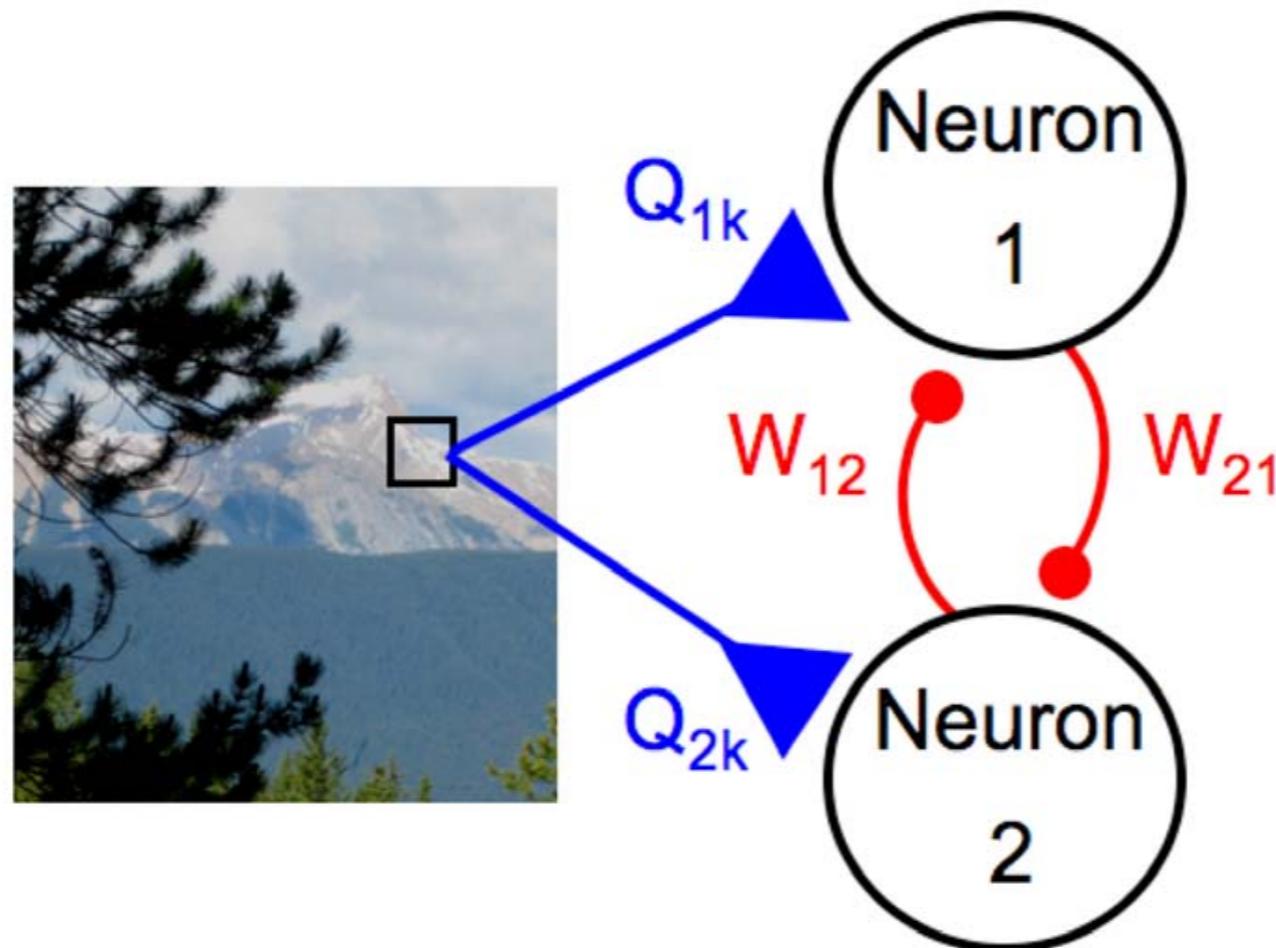
Foldiak, Biological Cybernetics (1990)

# Sparse coding

## Spiking neural network: SAILnet

SAILnet:

A spiking version of Foldiak 1990 network with **inhibitory** and **excitatory** synapses and local learning rules:

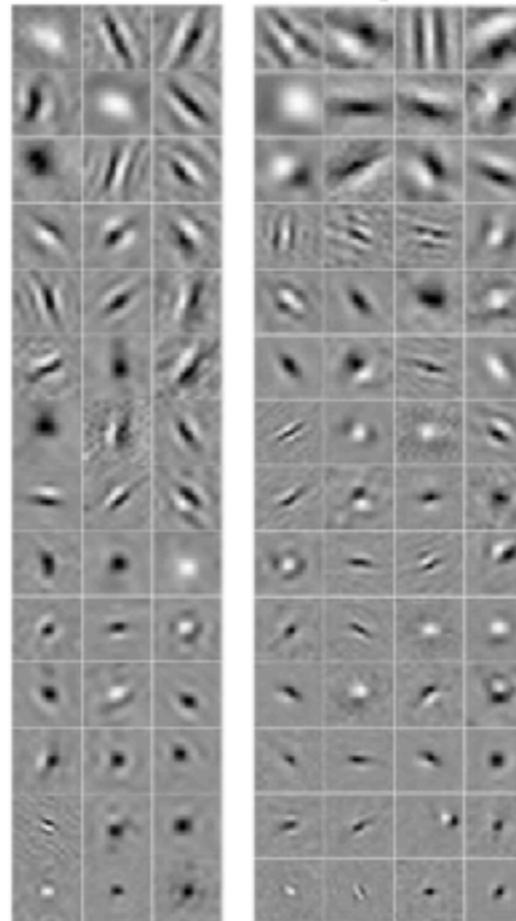


Zylberberg et al. PLoS CompBio (2011)

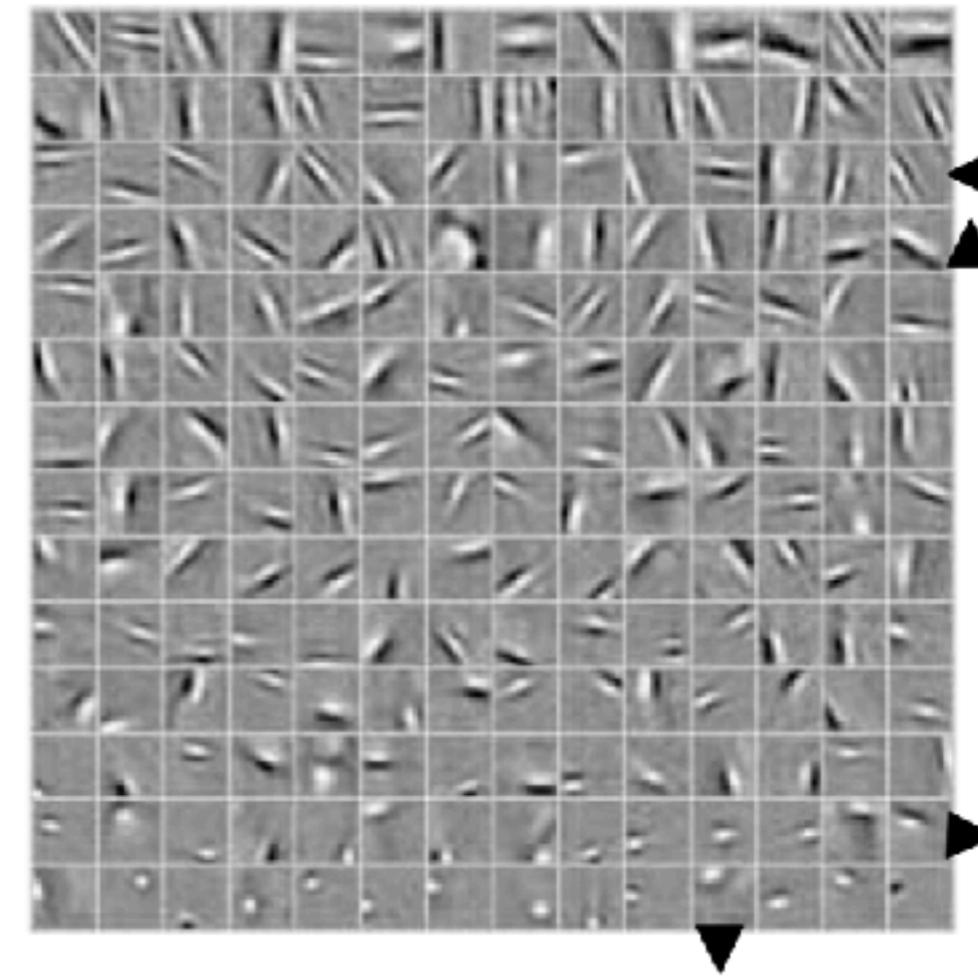
# Sparse coding SAILnet

Captures the diversity of receptive fields observed in animals:

A Macaque



B SAILnet



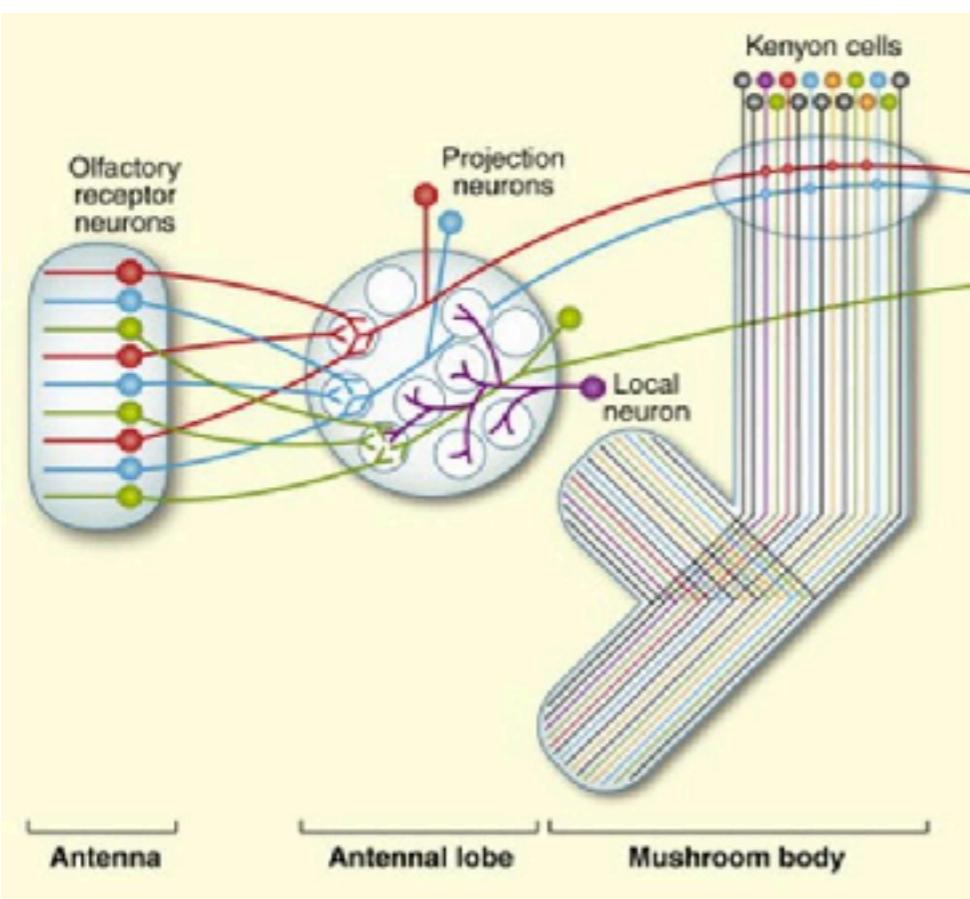
Zylberberg et al. PLoS CompBio (2011)

# Sparse coding across the animal kingdom: from mammals to insects..



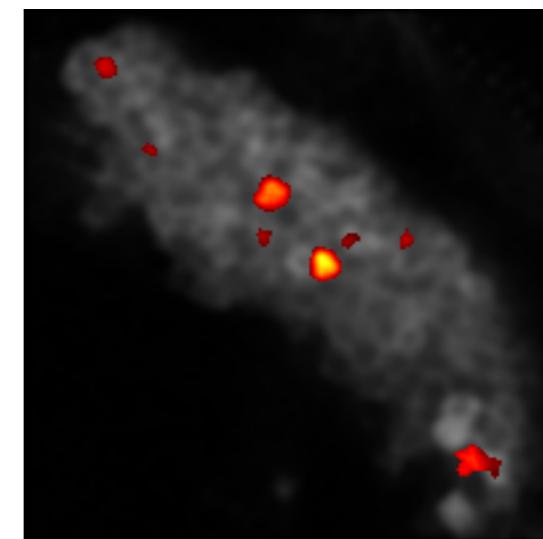
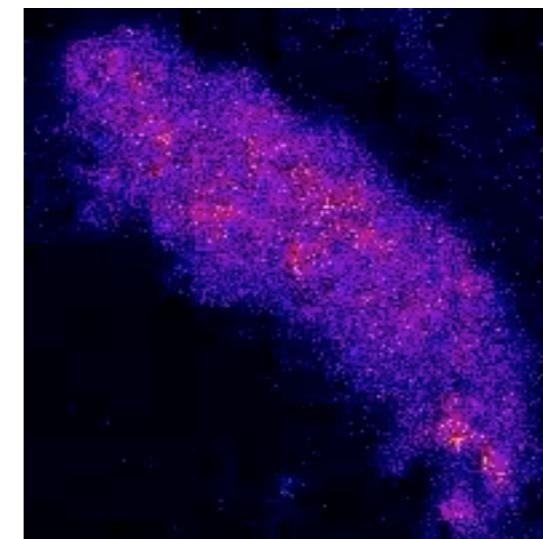
# Sparse coding in fruit flies

## Smell (odor) system in *Drosophila*



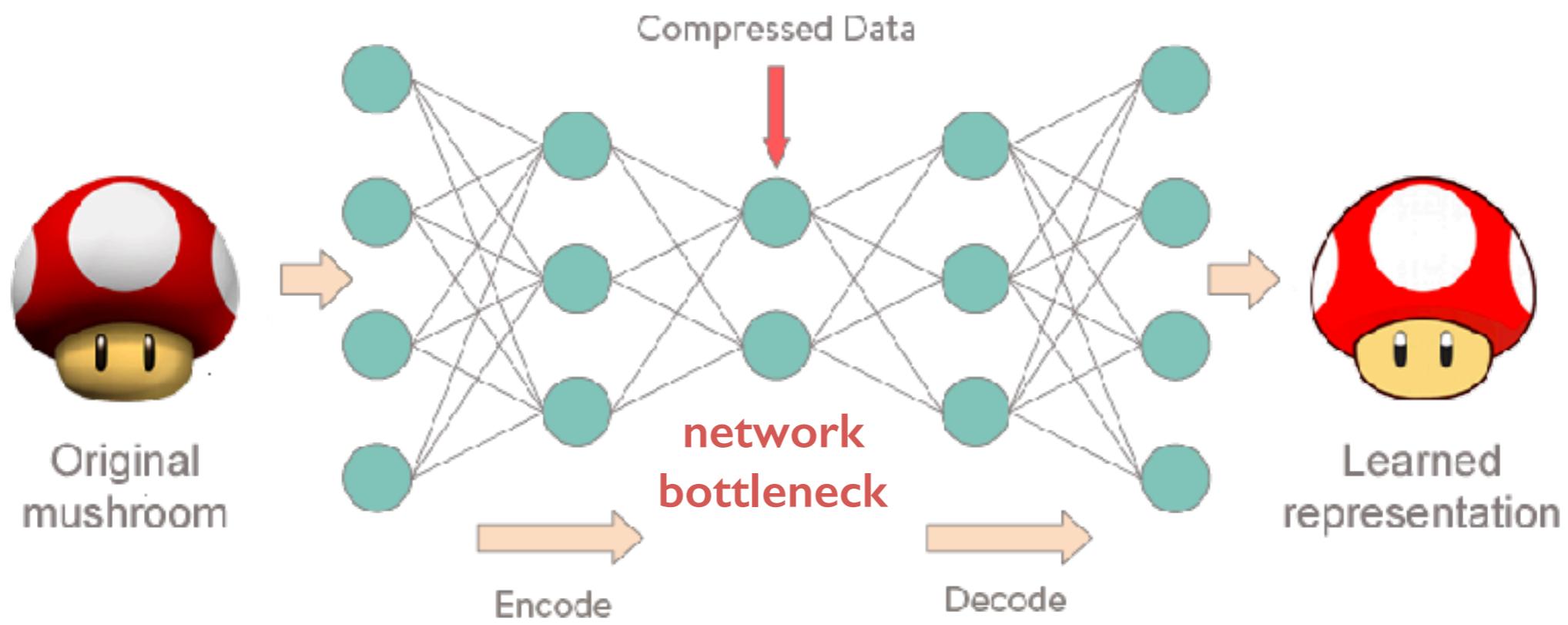
Masse et al. Current Biology (2009)

## Sparse coding in Kenyon cells



Lin et al. Nature Neuroscience (2014)

# Autoencoders



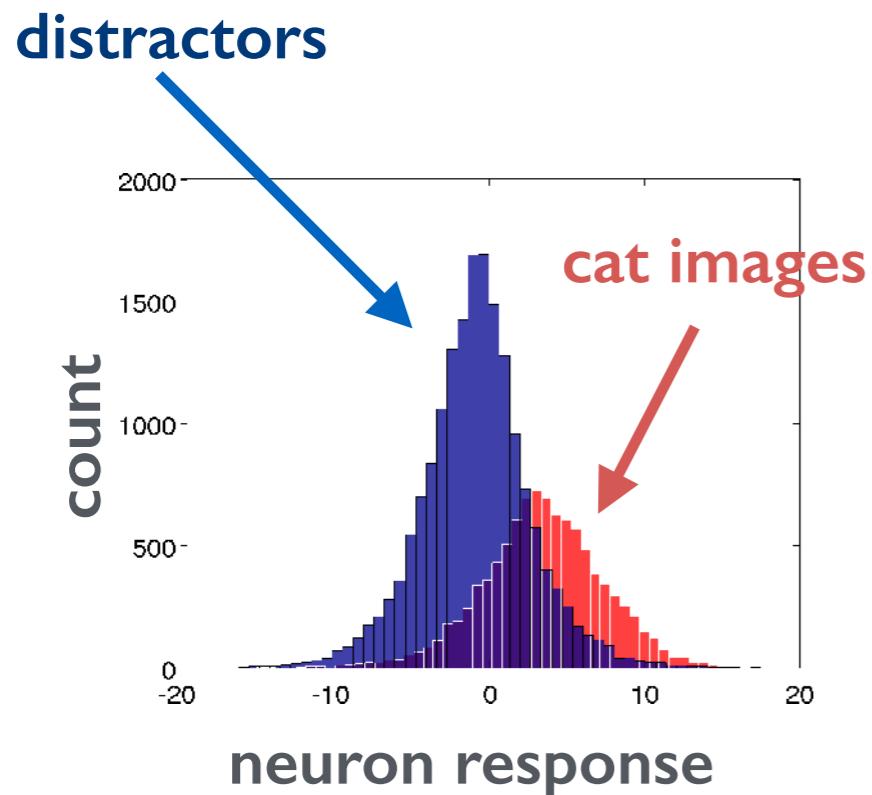
- Autoencoders also aim at reconstructing the input, but without an explicit constrain on sparsity (instead uses other forms of regularization)
- Autoencoders rely on a specific model  $f$  (i.e. neural network with bottleneck)
- If an autoencoders uses a sparsity constrain, then are equiv. to sparse coding

# Autoencoders

## Cat neuron

Autoencoder trained on 10 million images from YouTube movies.

Showing that it is possible to learn high level feature detectors from unlabelled data (akin to grandmother cells).



Le et al. ICML (2012)

# Summary

1. **Sparse coding** is an **optimisation** method that develops useful representations of the input
2. Olshausen and Field introduced a sparse coding framework that can be mapped onto the brain
3. **Sparse coding** develops representations akin to the brain
4. Autoencoders can also be used to extract useful representations, but do not (typically) rely on a sparsity cost and assume a specific neural network

# References

## **Text books:**

Natural Image Statistics, by Hyvarinen et al. 2009 [good reference on unsupervised learning from natural images]

Theoretical neuroscience: Dayan and Abbott 2001 [Unsupervised learning: chapter 3]

Deep Learning by Courville et al. 2015

## **Relevant papers:**

- Foldiak, *Biological Cybernetics* (1990) [classical work on developing sparse representations with local Hebbian learning rules]
- Olshausen and Field, *Nature* (1996) [seminal work on sparse coding using natural images]
- Zylberberg et al., *PLoS Computational Biology* (2001) [more recent work on sparse coding in spiking networks with local learning rules]

# Upcoming lectures

- L10: Neural circuits and learning: introduction
  - Visual processing
    - L11: Visual cortex
    - L11: Convolutional neural networks
  - Learning in the brain
    - L12: Supervised learning: The backpropagation algorithm/cerebellum
    - L13: Unsupervised learning: Sparse coding and Boltzmann Machines
    - **L14: Reinforcement learning: TD learning, Q learning, deep RL and dopamine**
  - Temporal processing in the brain
    - L14: Auditory cortex and recurrent neural networks
    - L15: Gated recurrent neural networks