Artificial and robotic vision

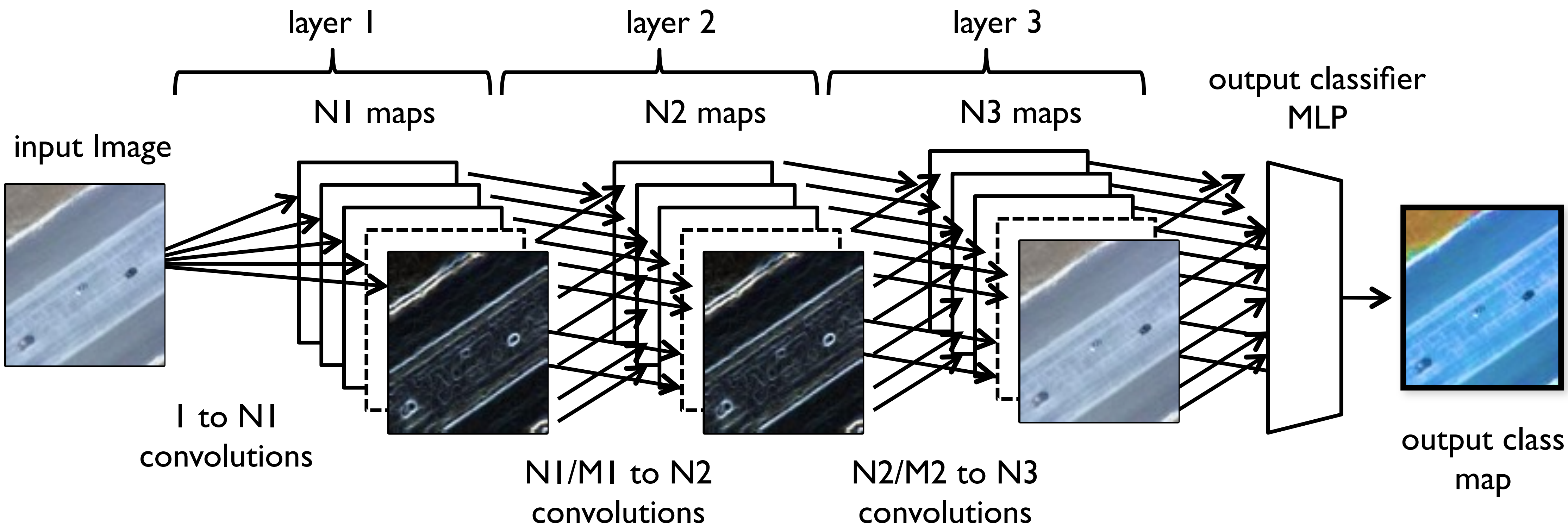
Spring 2013

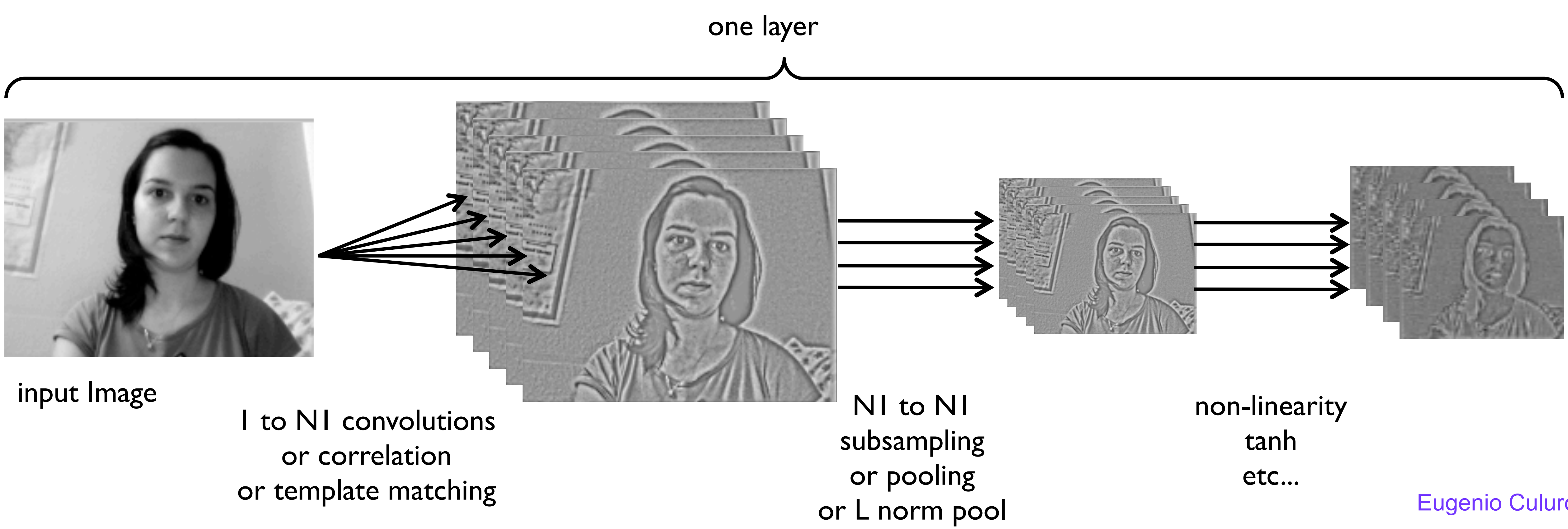
Lecture 6: unsupervised learning

e-Lab

Eugenio Culurciello

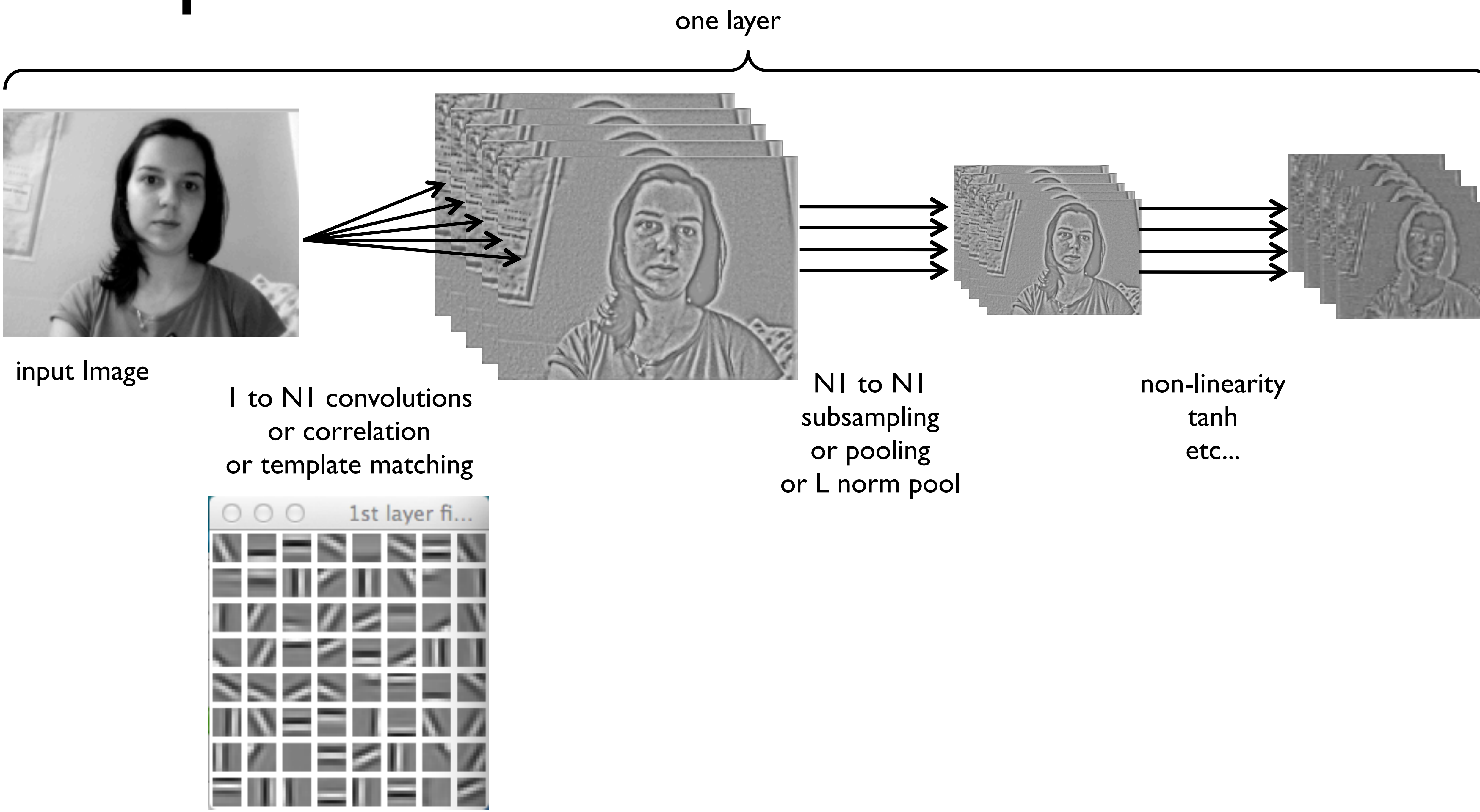
deep networks





Eugenio Culurciello

deep networks

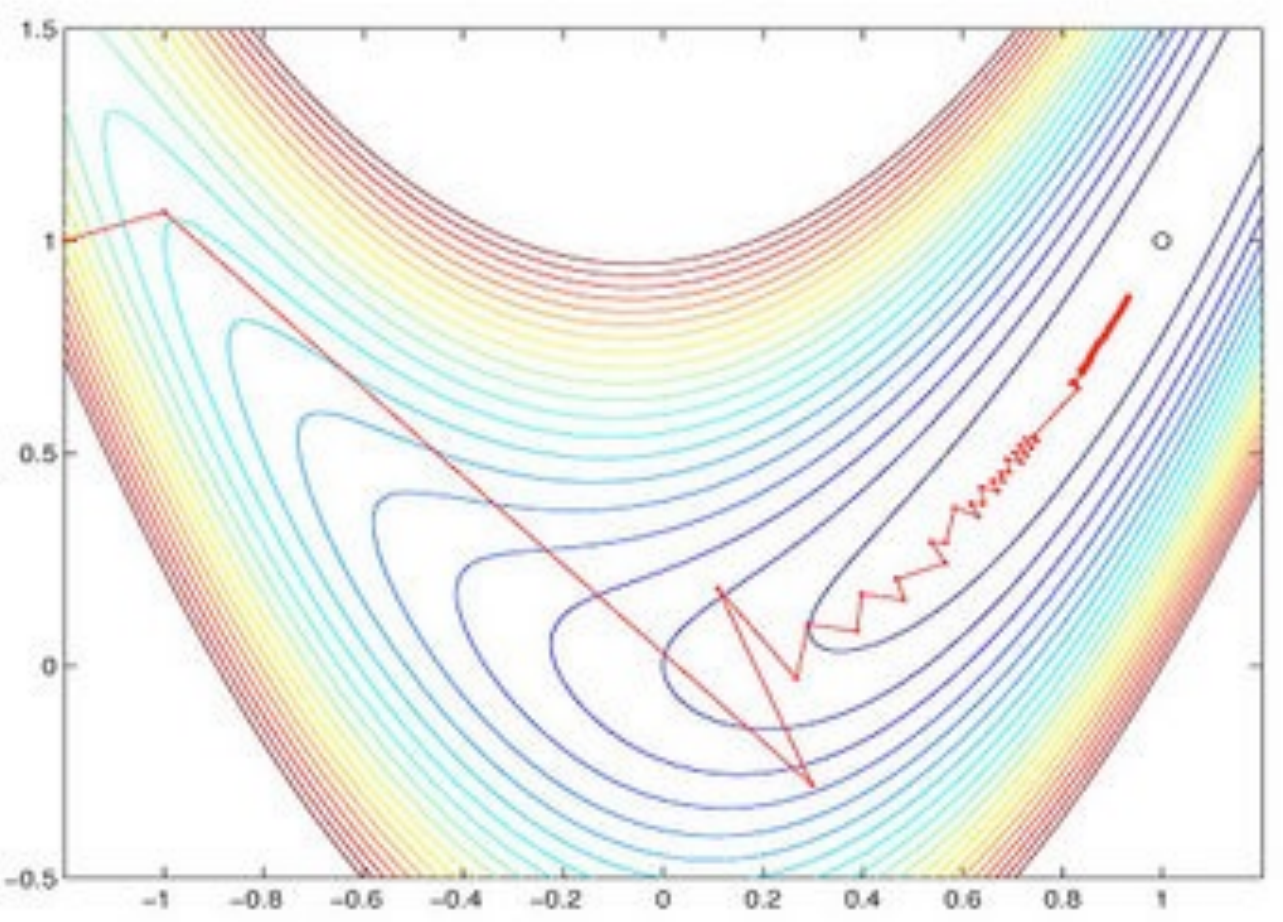


How do we compute these filters?

Eugenio Culurciello

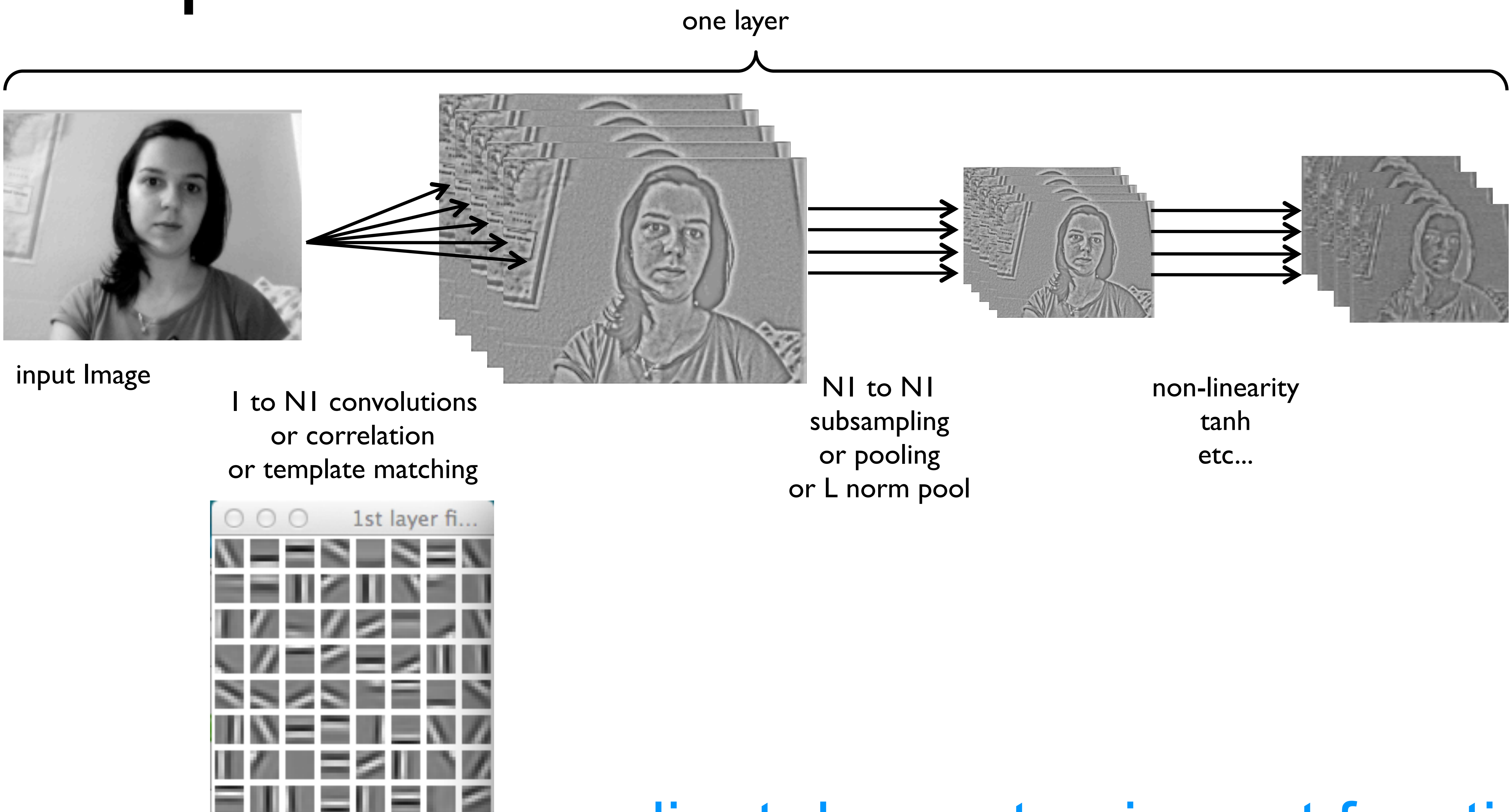
supervised training

dataset



stochastic gradient descent

deep networks



gradient descent, min cost function

$$\underset{W}{\text{minimize}} \quad \lambda \sum_{t=1}^{N-1} \|\mathbf{p}^{(t)} - \mathbf{p}^{(t+1)}\|_1 + \sum_{t=1}^{N} \|\mathbf{x}^{(t)} - W^T W \mathbf{x}^{(t)}\|_2^2 \quad + \gamma \sum_{t=1}^{N} \|\mathbf{p}^{(t)}\|_1$$

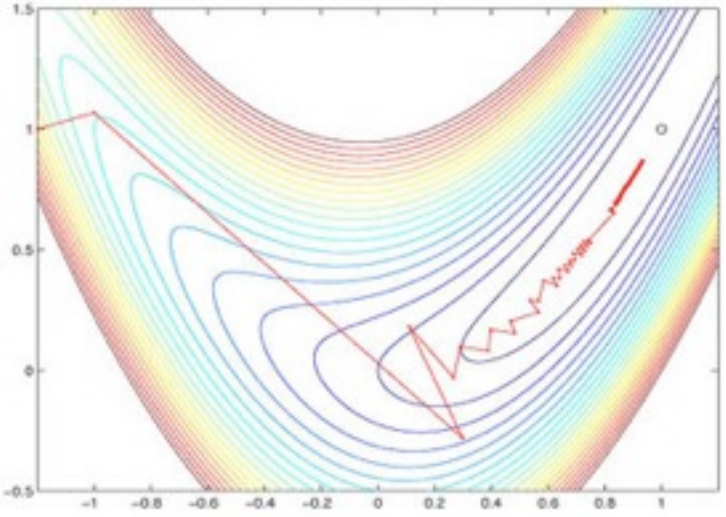
p = features, x = input

Eugenio Culurciello

supervised training issues:

- need labeled data!
- takes a long time
- lots of effort
- in videos it is crazy
- we cannot scale up!

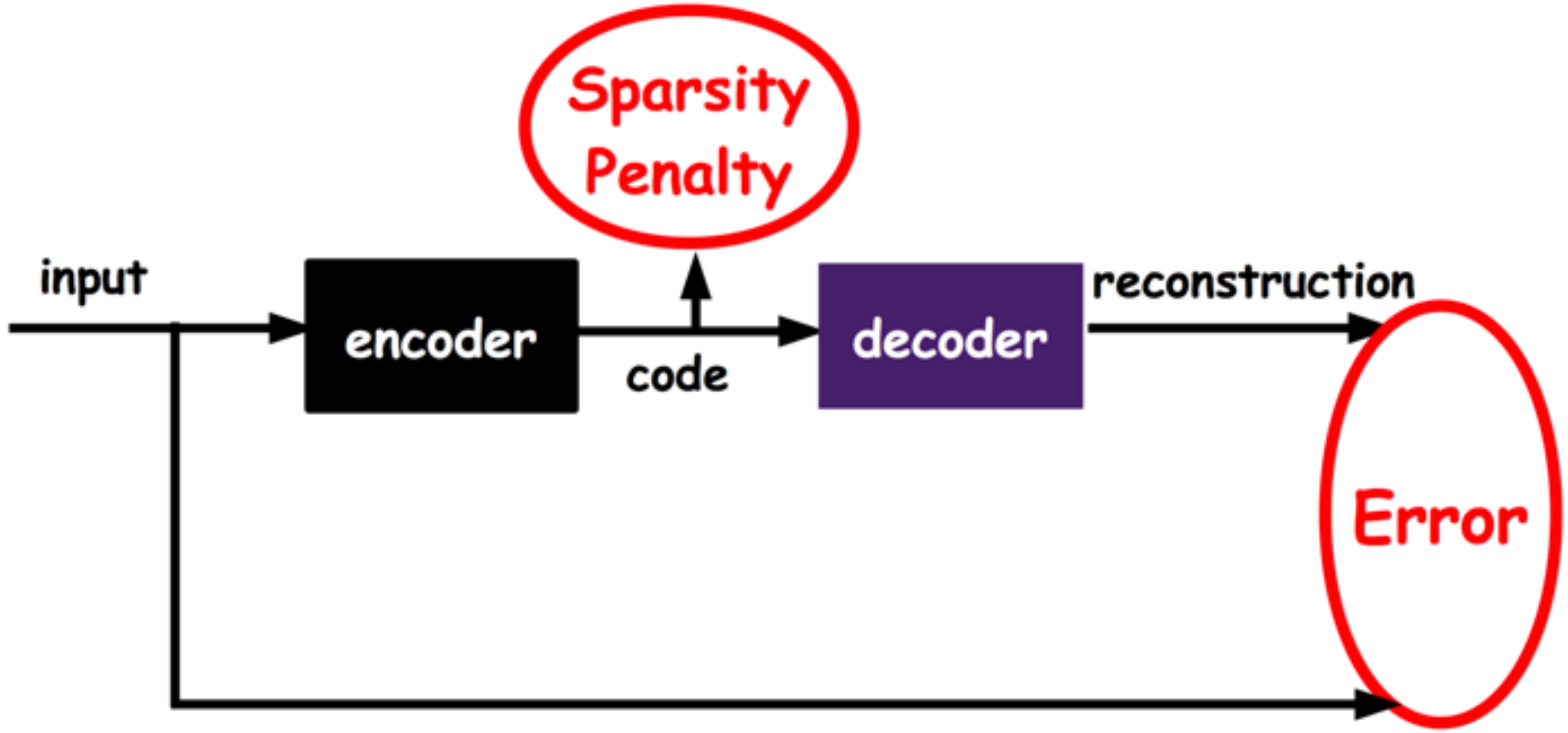
- ~all use gradient descent
- not related to learning in the brain
- use global error propagation
- not local err. prop.
- math-heavy techniques
- they take a long time to compute



Eugenio Culurciello

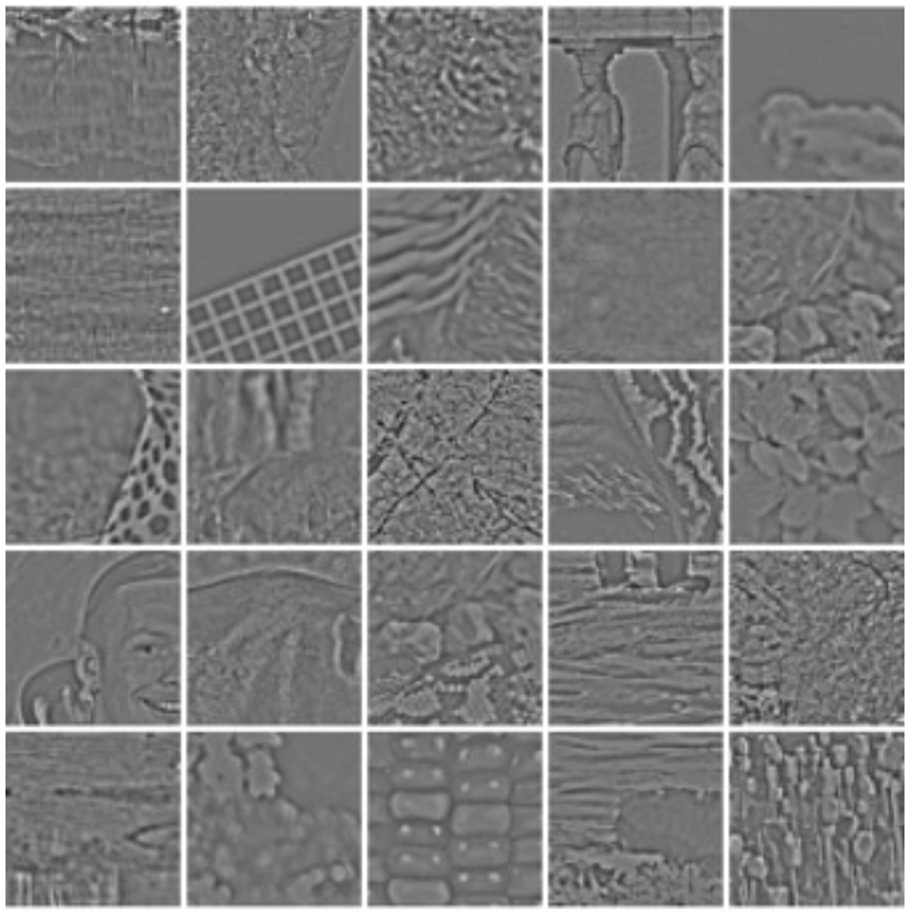
unsupervised training: autoencoders

main idea: learn to reconstruct the input with some sparse base functions

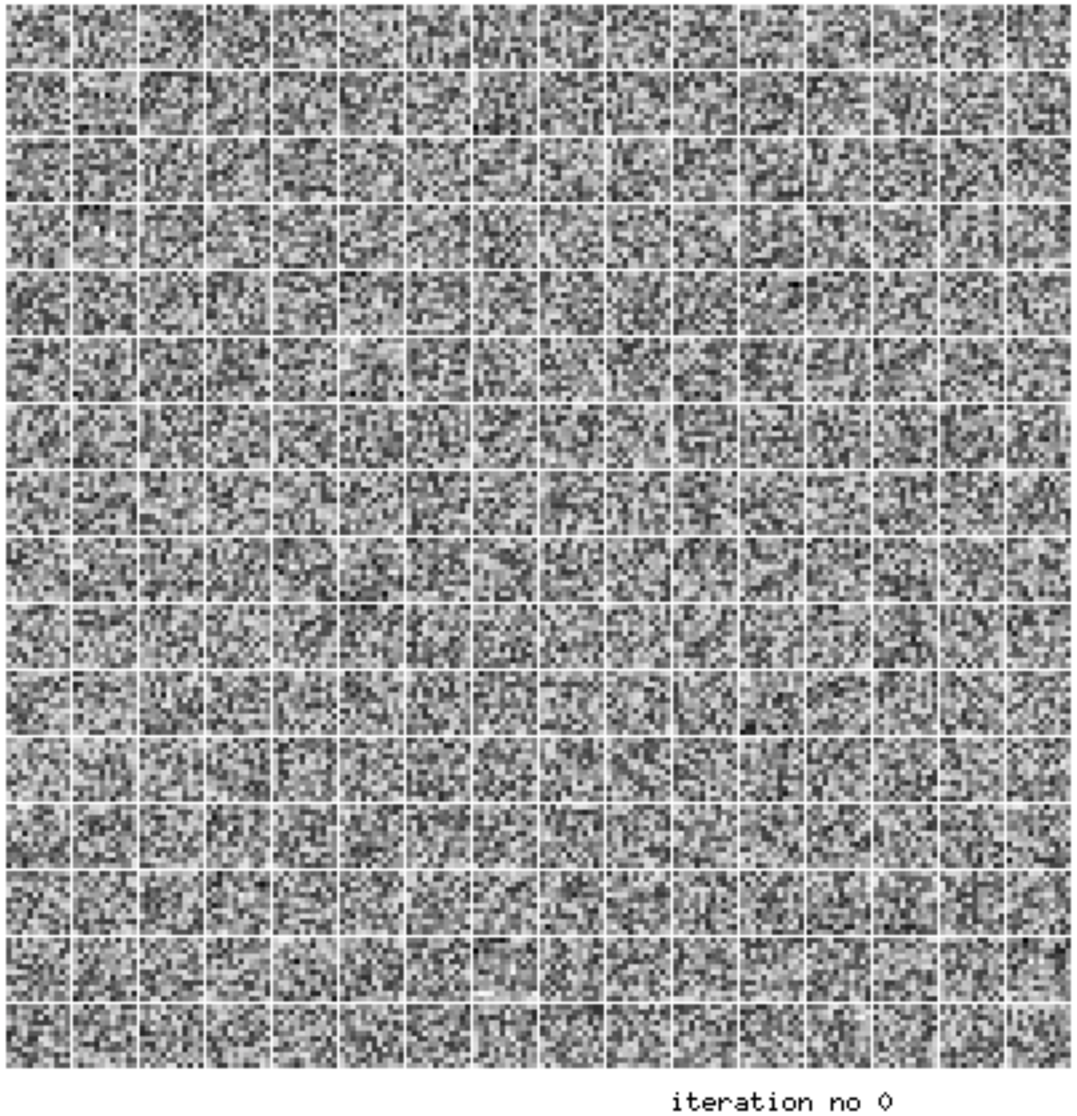


Karol Gregor, Marc'Aurelio Ranzato, Koray Kavukcuoglu, Yann LeCun

unsupervised training: autoencoders



the dataset is just unlabeled data!
of any form: video, frames, etc
easy to get!
easy to consume

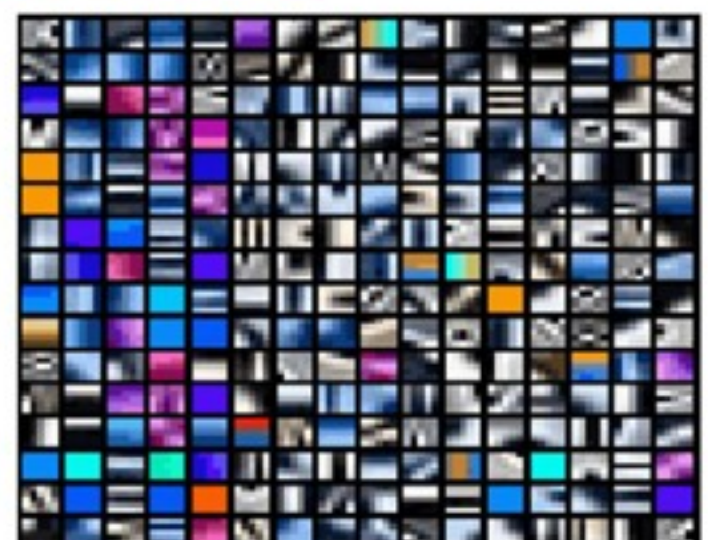


train by finding base-functions
or the basic blocks of images

unsupervised training: autoencoders

related to:
vector quantization
clustering

used for:
compression
de-noising



Eugenio Culurciello

sparse coding

Sparse coding (Olshausen & Field,1996). Originally developed to explain early visual processing in the brain (edge detection).

Training: given a set of random patches x, learning a dictionary of bases [$\Phi_{1,}$ $\Phi_{2,}$ …]

$$\min_{a,\phi} \sum_{i=1}^{m} \left\| x_i - \sum_{j=1}^{k} a_{i,j}\phi_j \right\|^2 + \lambda \sum_{i=1}^{m} \sum_{j=1}^{k} |a_{i,j}|$$

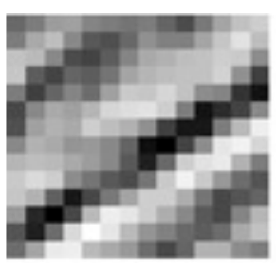
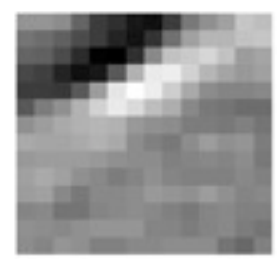
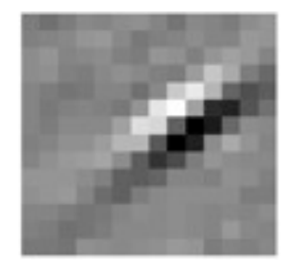
Kai Yu -Baidu Univ.

sparse coding

Input: Novel image patch x (in R^d) and previously learned ϕ_i's

Output: Representation $[a_{i,1}, a_{i,2}, \ldots, a_{i,\kappa}]$ of image patch x_i.

$$\min_a \sum_{i=1}^{m} \left\| x_i - \sum_{j=1}^{k} a_{i,j} \phi_j \right\|^2 + \lambda \sum_{i=1}^{m} \sum_{j=1}^{k} |a_{i,j}|$$

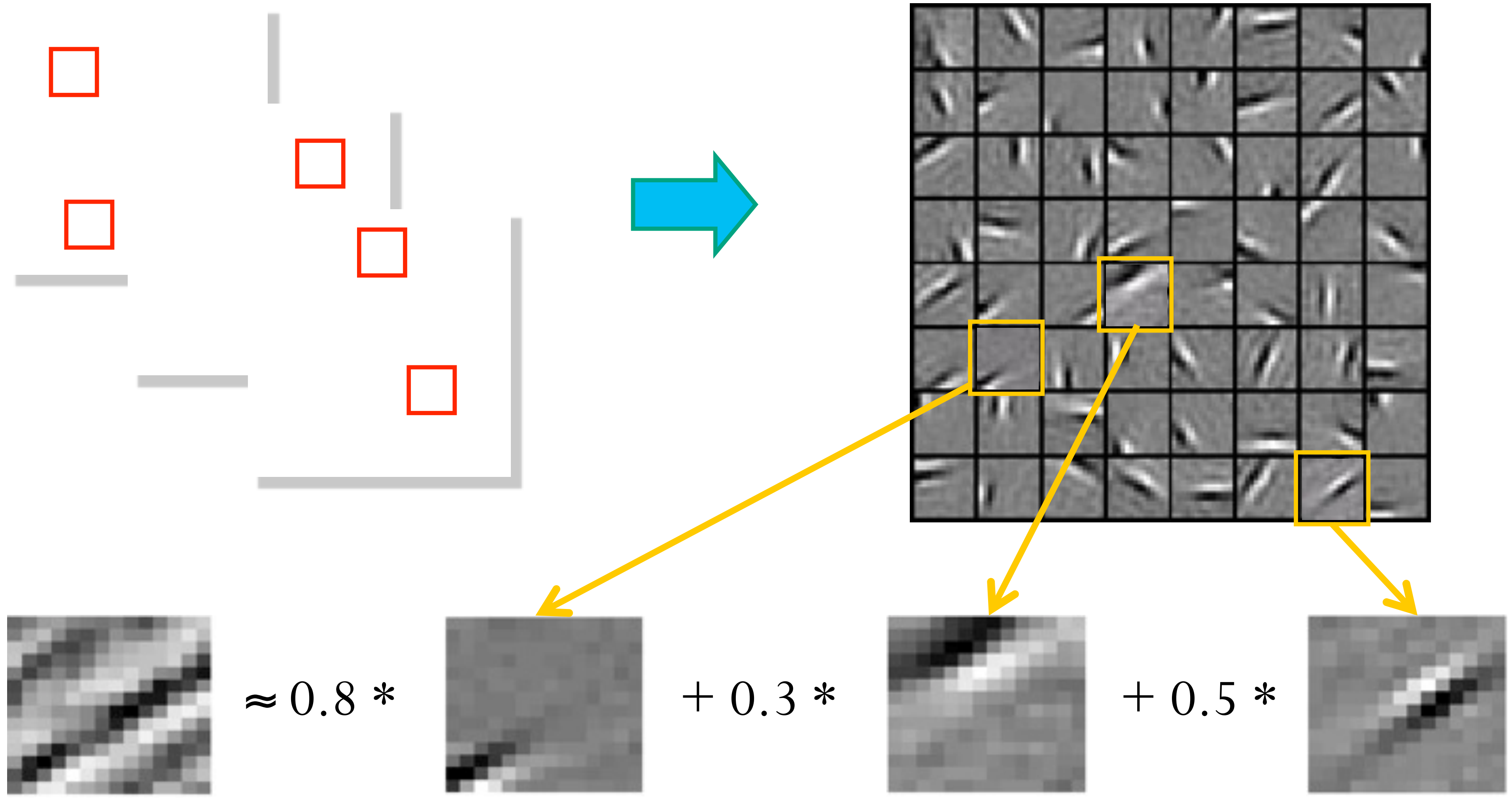
 $\approx 0.8 *$  $+ 0.3 *$  $+ 0.5 *$ 

Represent x_i as: a_i = [0, 0, …, 0, **0.8**, 0, …, 0, **0.3**, 0, …, 0, **0.5**, …]

Kai Yu -Baidu Univ.

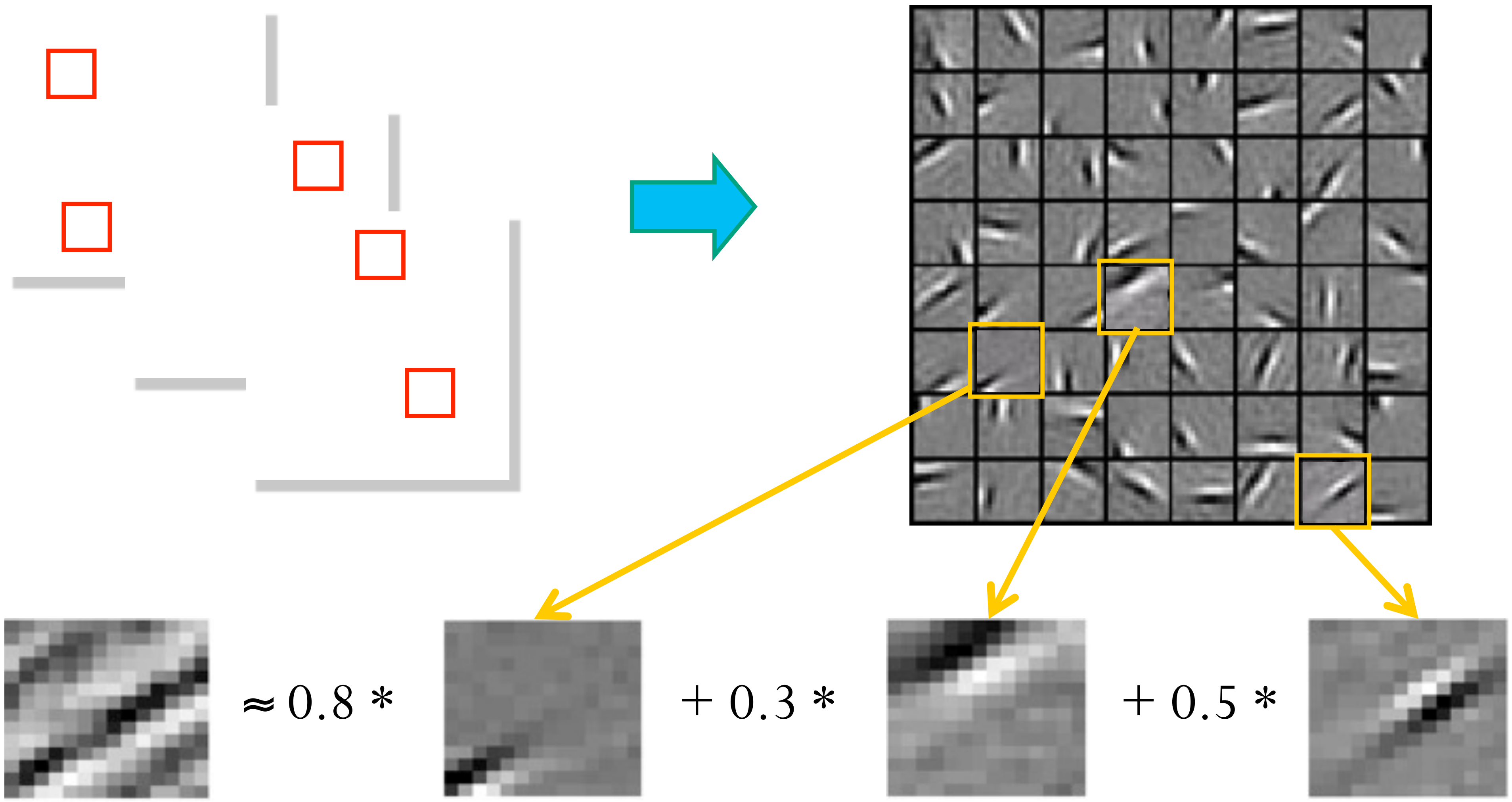
Eugenio Culurciello

sparse coding



Slide credit: Andrew Ng

sparse coding

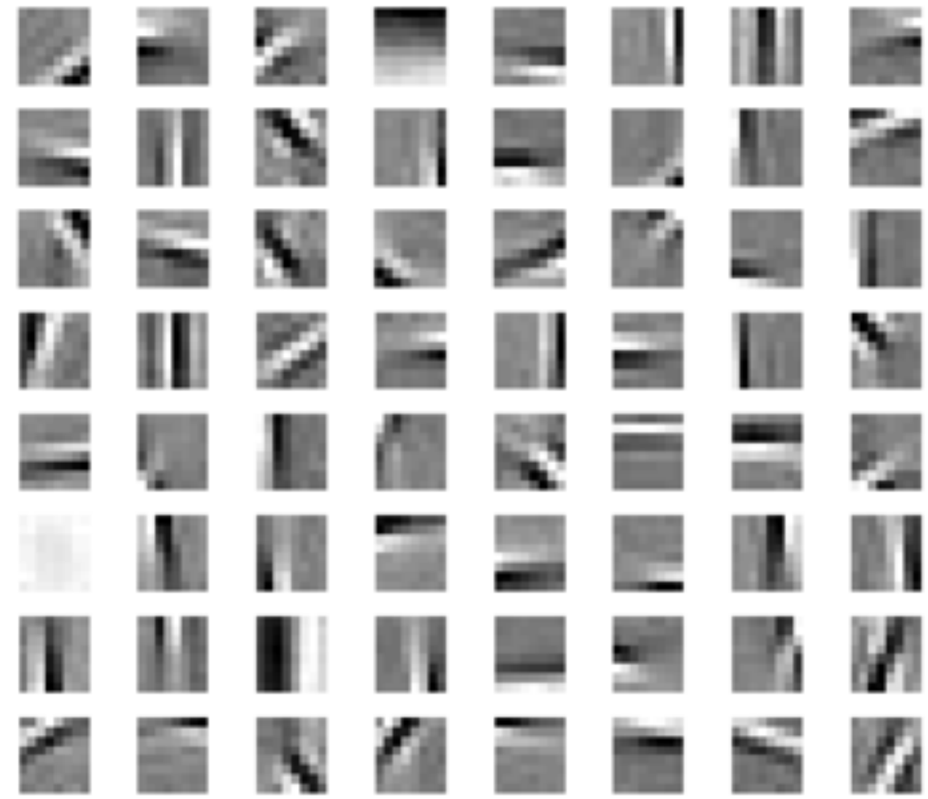


$[a_1, \ldots, a_{64}] = [0, 0, \ldots, 0, \mathbf{0.8}, 0, \ldots, 0, \mathbf{0.3}, 0, \ldots, 0, \mathbf{0.5}, 0]$
(feature representation)

Compact & easily interpretable

Slide credit: Andrew Ng

sparse coding: example

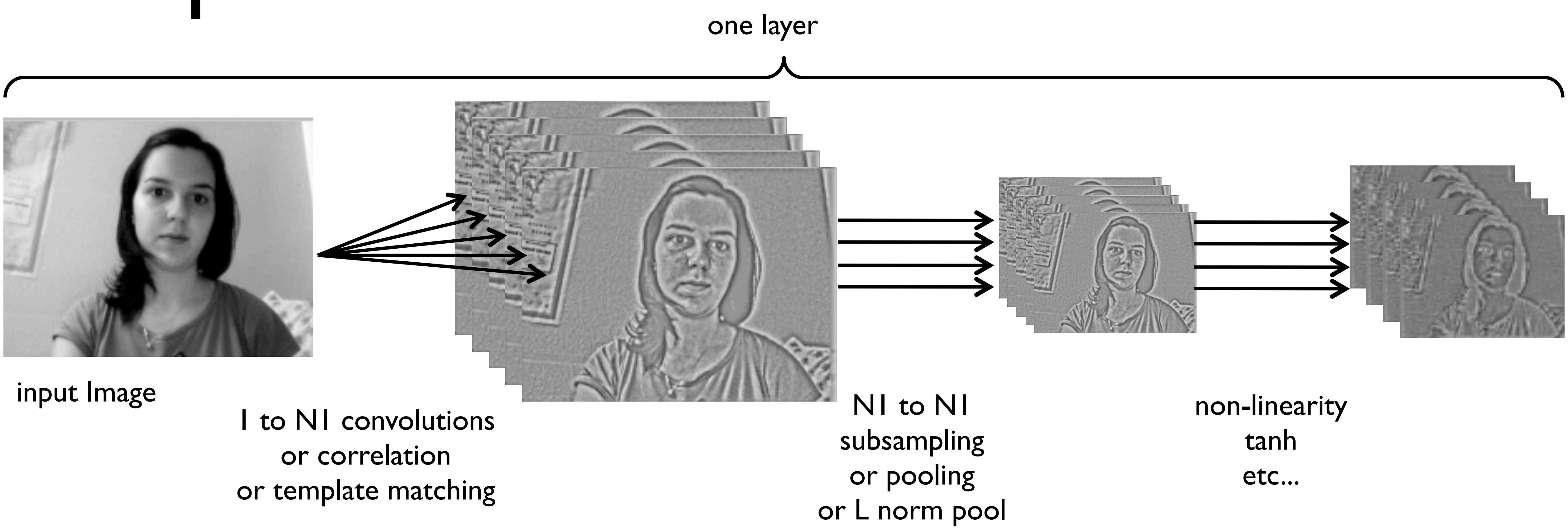


64 bases functions of 8x8 pixels

The bases seem to capture the intrinsic structure of the building elements, that are mainly composed of vertical, horizontal, slanting edges and corners.

Eugenio Culurciello

deep networks



Multiple layers of deep network:

Repeat for each layer:

1- sample output of previous layer (new input)

2- learn dictionary of inputs = filters

3- use filters to generate outputs

Eugenio Culurciello

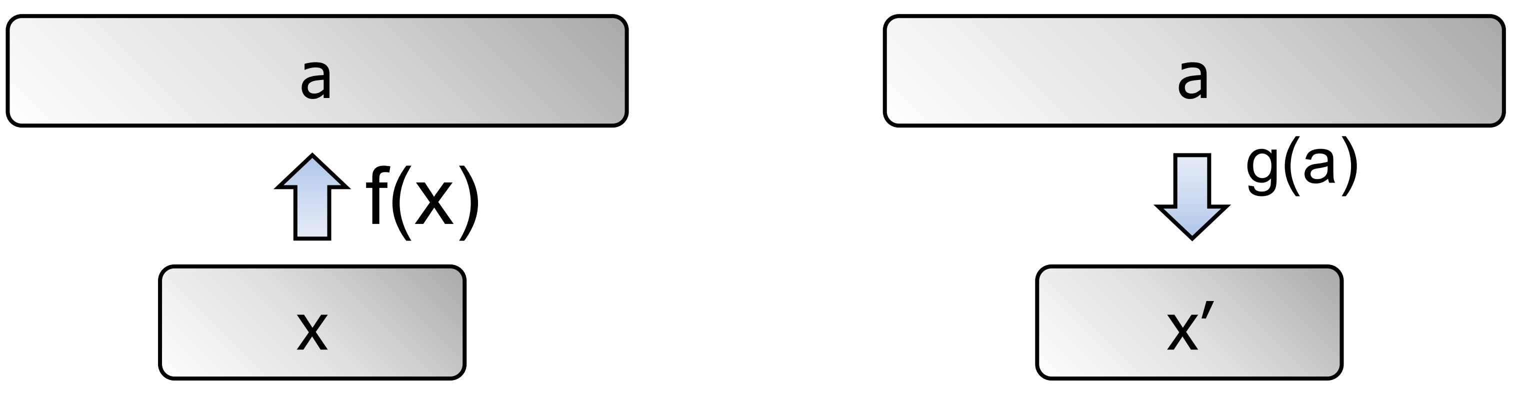
sparse coding: deep networks

Any feature mapping from x to a, i.e. a = f(x), where

-a is sparse (and often higher dim. than x)

-f(x) is nonlinear

-reconstruction x'=g(a) , such that x'≈x



Therefore, sparse RBMs, sparse auto-encoder, even VQ can be viewed as a form of sparse coding.