Artificial and robotic vision

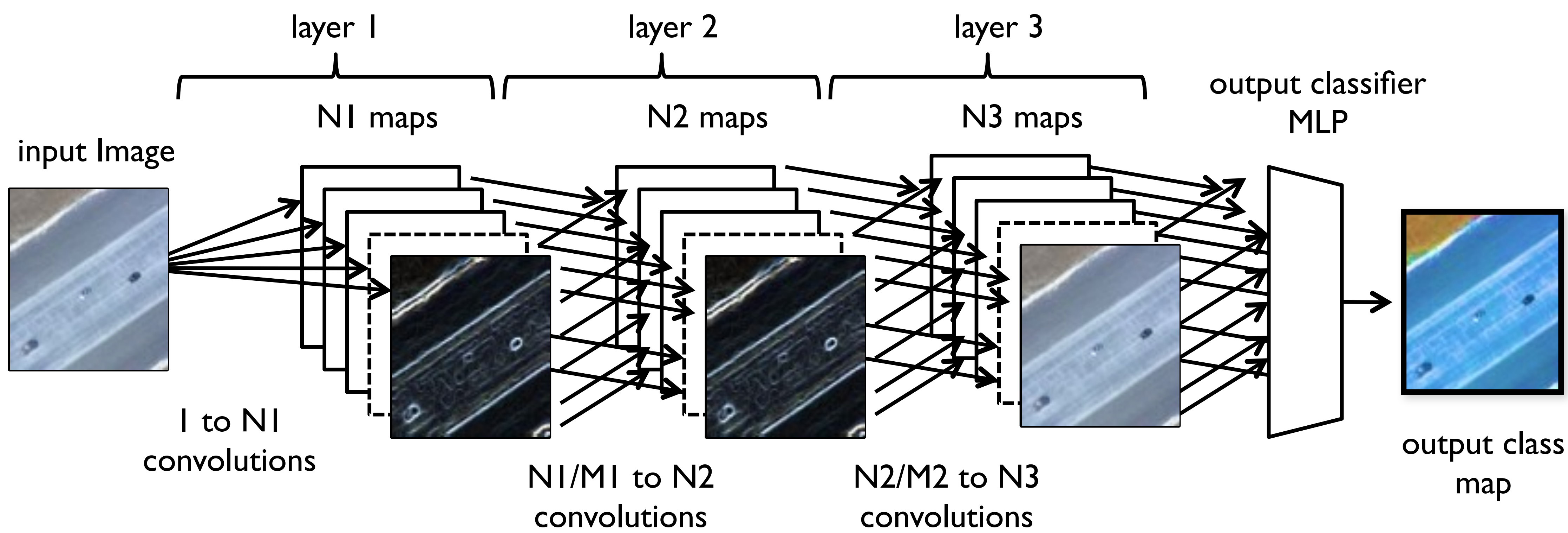
Spring 2013

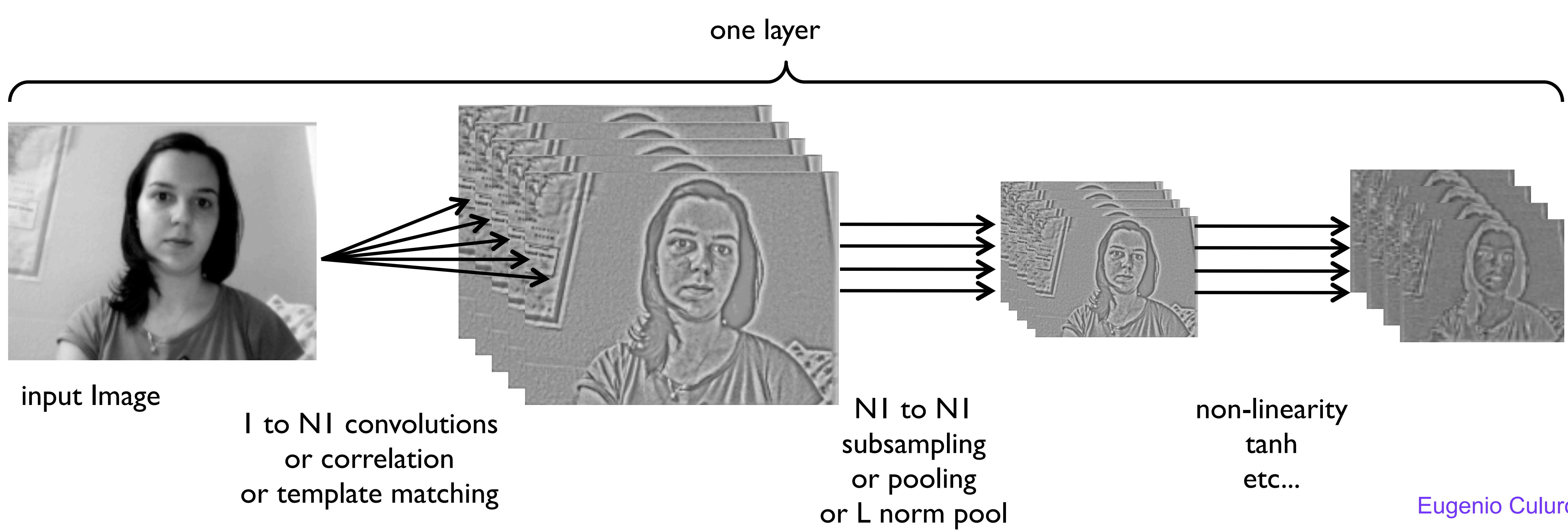
Lecture 8: thoughts and tips on Deep Learning

e-Lab

Eugenio Culurciello

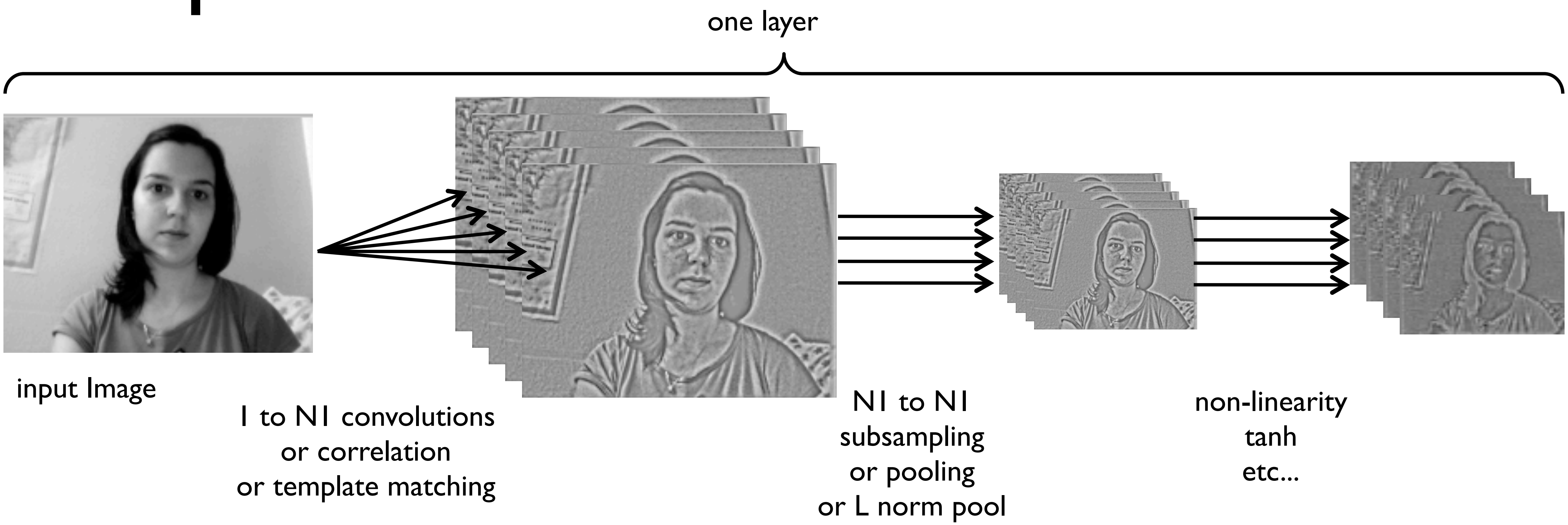
deep networks





Eugenio Culurciello

deep networks



Multiple layers of deep network:

Repeat for each layer:

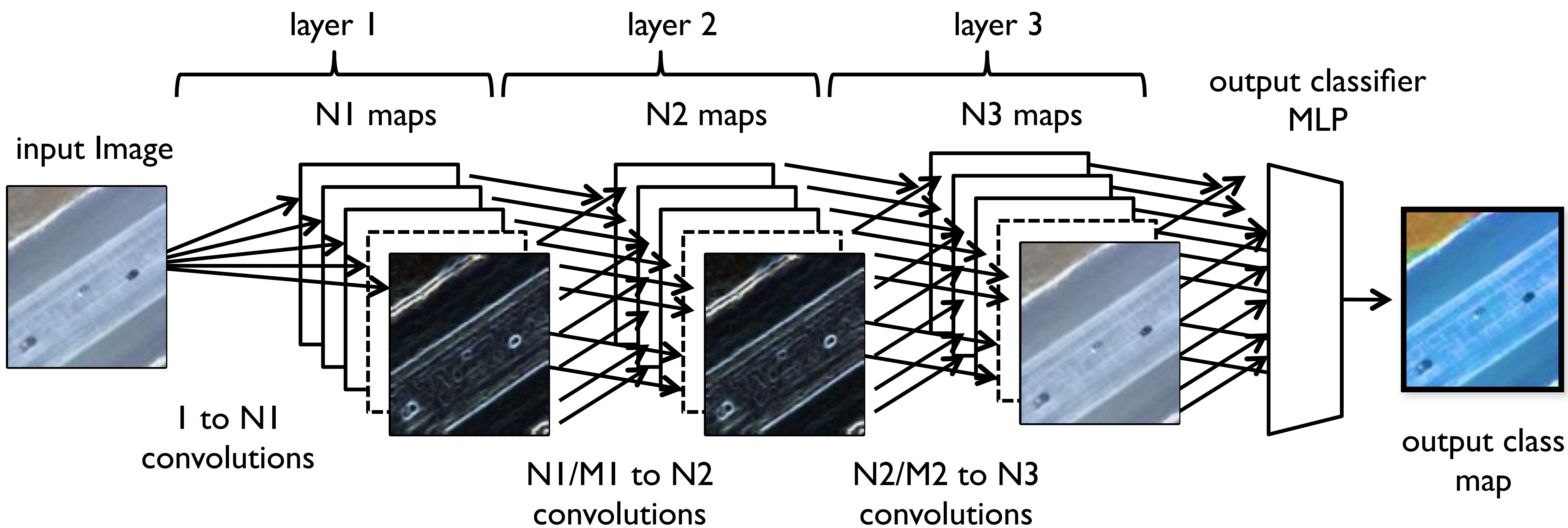
1- sample output of previous layer (new input)

2- cluster these inputs = filters

3- use filters to generate outputs

Eugenio Culurciello

deep networks



Deep networks are defined by:

- size of inputs and layer maps/neuron-count
- number of filters, layers, connections
- kernels/filters at each layer
- connection between layers
- pooling layers

deep networks

Deep networks are defined by:

Architecture:

- size of inputs and layer maps/neuron-count
- number of filters, layers, connections
- pooling layers

Learning/Training

- kernels/filters at each layer
- connection between layers

deep networks

Deep networks are defined by:

Architecture:
- size of inputs and l … count
- nu … connections
- poo … layers

need more work here!

Learning/Training
- kernels/filters at ea …
- con … yers

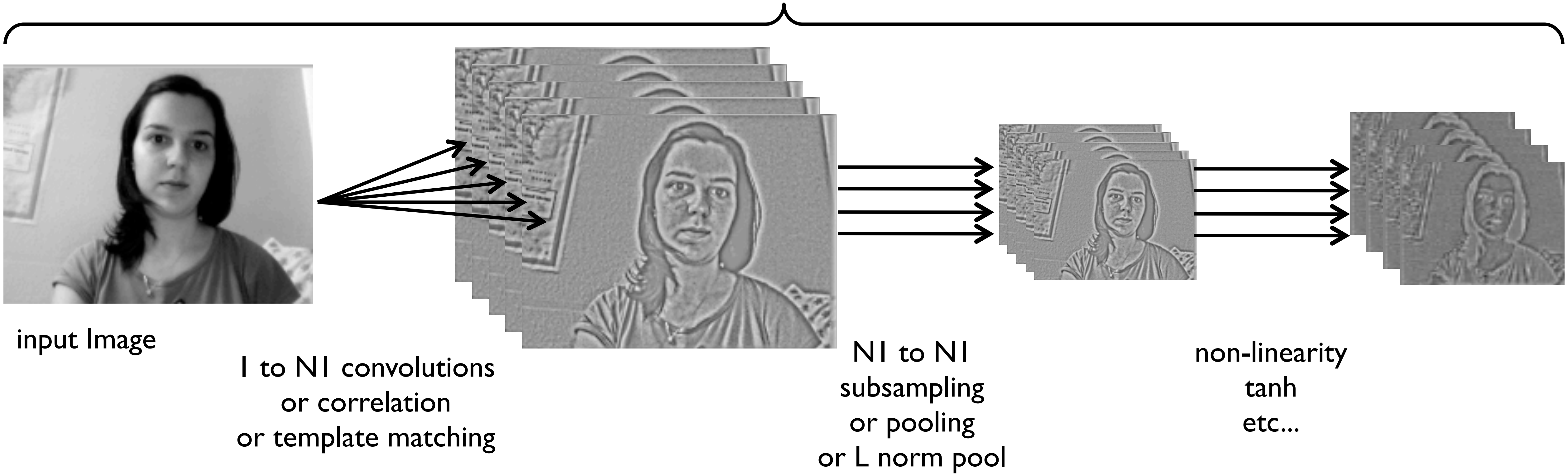
Really all should be learned!!!

deep networks

The next slides are my findings
after several experiments
on unsupervised deep-nets

For more details and references
refer to tips, tricks and ideas
from the list at the end of this lecture

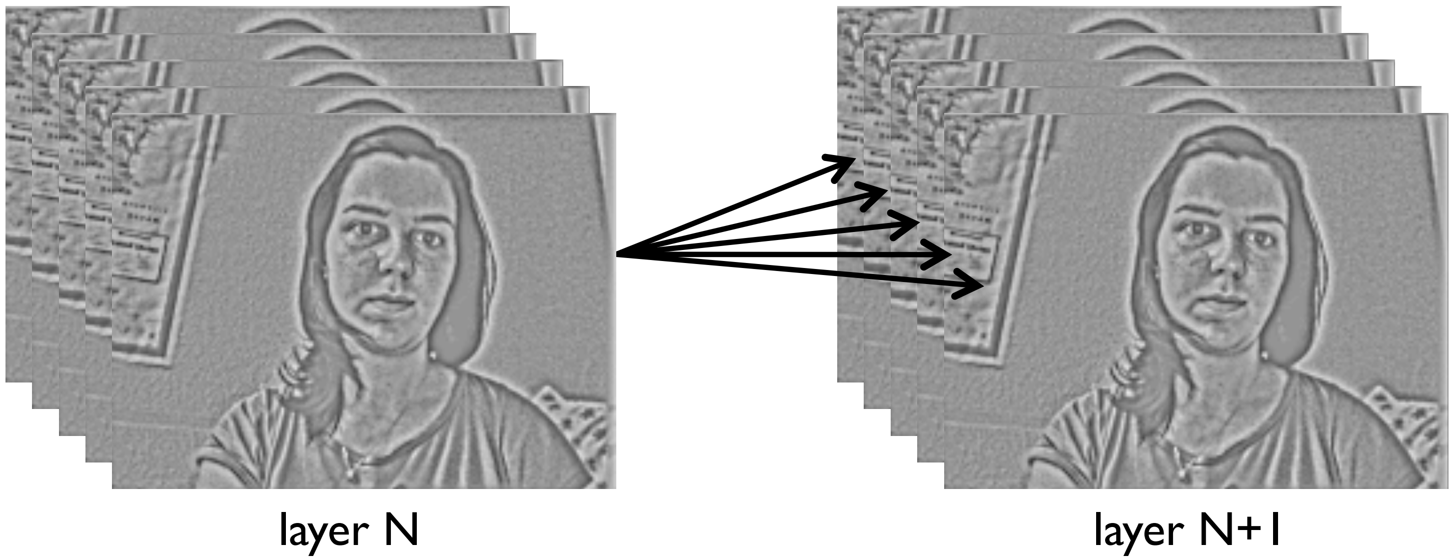
connections!



Unsupervised techniques:
=>they train filters

- how about connections?
- between layers?
- need to learn them!

connections!

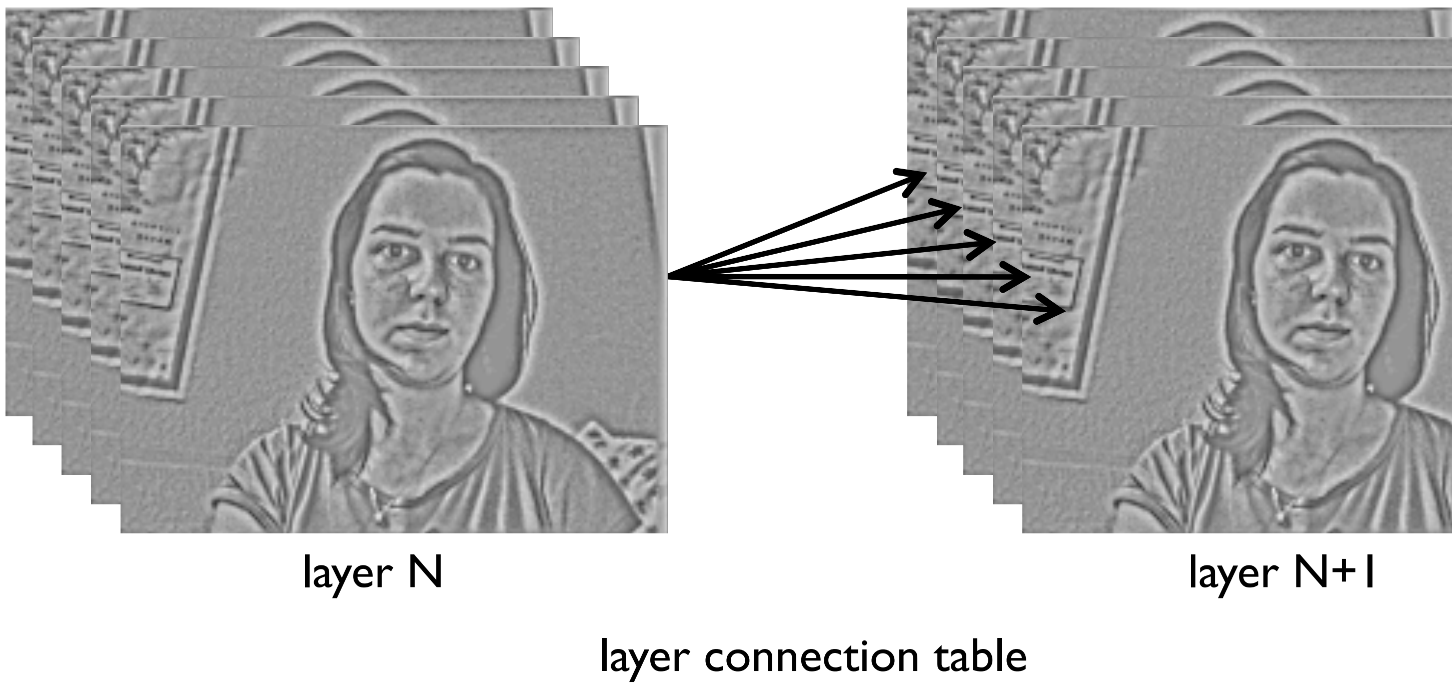


layer connection table

Layer connections tables:

- 1 to 1
- N to M
- 1 to ALL (fully connected)

connections!



- 1 to 1
  limited use: few neuron maps in next layer use previous layer features!

- N to M
  typical random connections: each next layer neuron connects to N previous layer neurons chosen at random

- 1 to ALL (fully connected)
  slower because of large amount of computation, also can reduce previous layer strong features by averaging them with lots of other non-strong ones

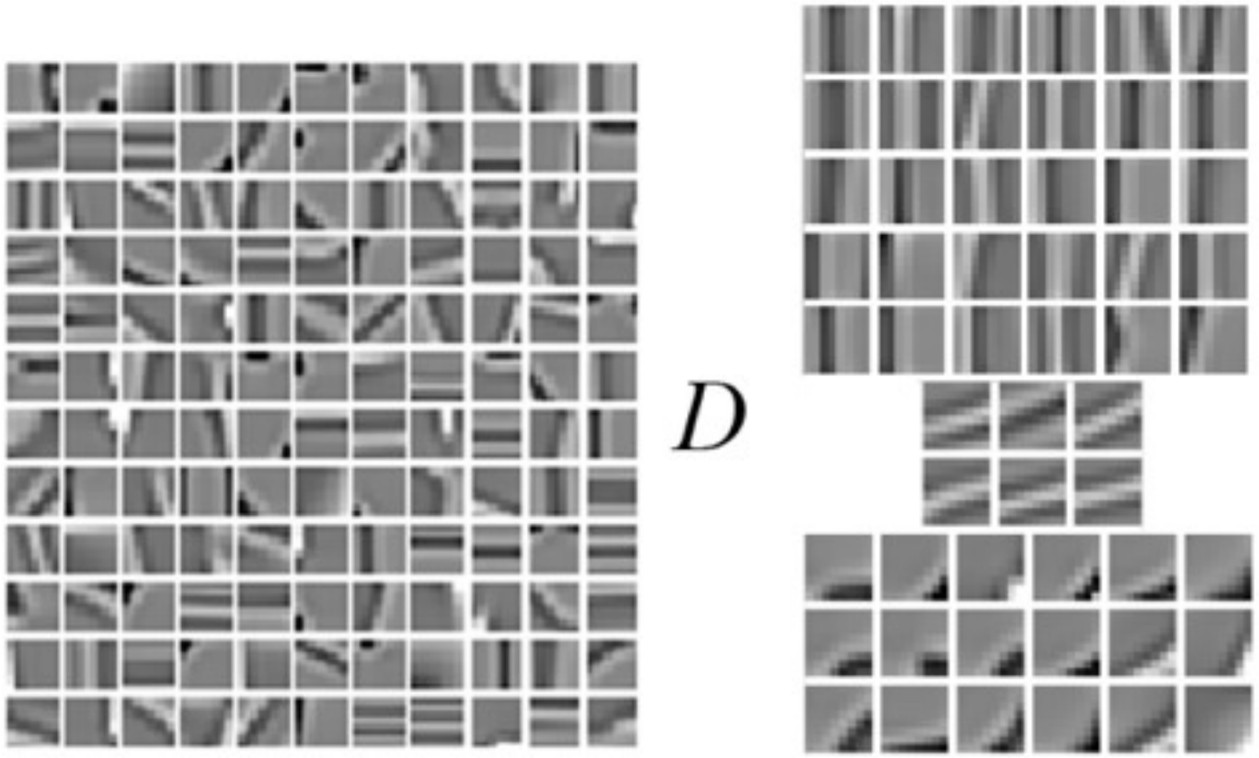
Eugenio Culurciello

connections!

Emergence of Object-Selective Features in Unsupervised Feature Learning

Adam Coates, Andrej Karpathy, Andrew Y. Ng

Single Link Agglomerate Clustering



group similar units
max pool them

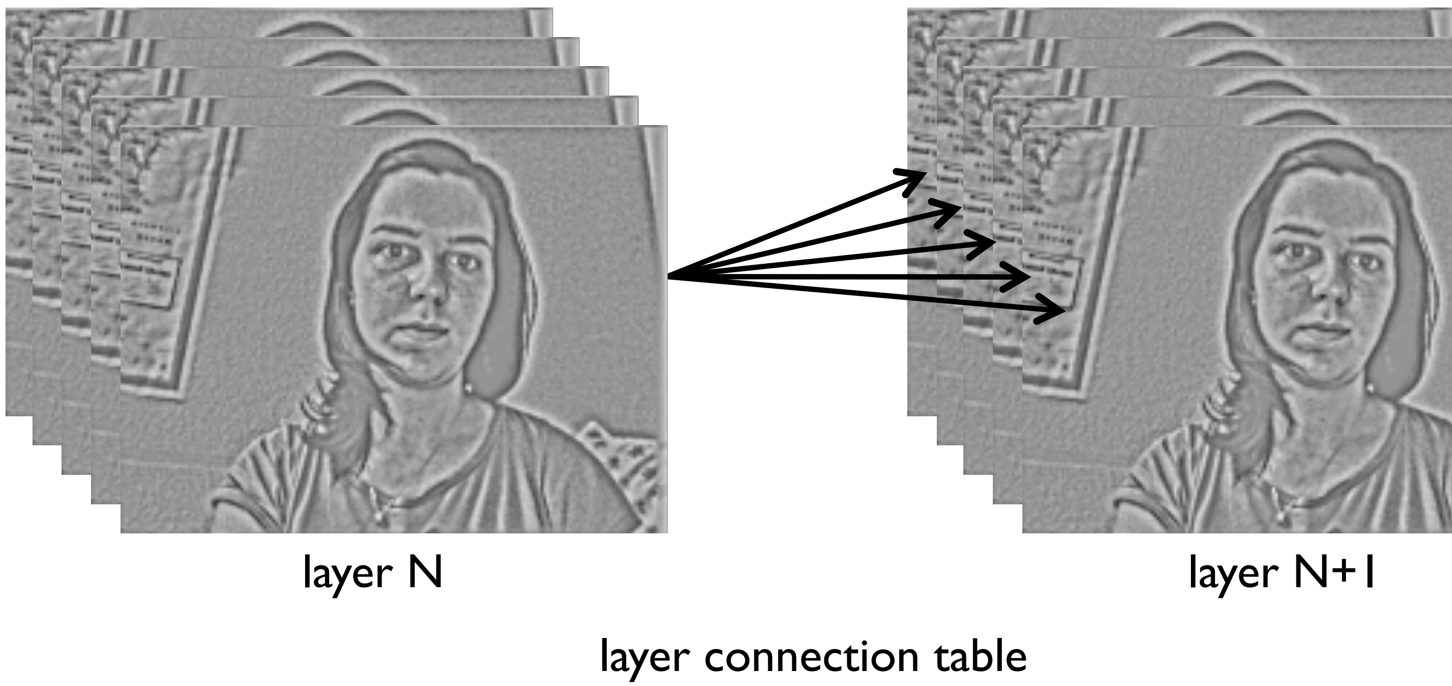
To construct invariant complex cell features a common approach is to create “pooling units” that combine the responses of lower-level simple cells. In this work, we use max-pooling units

Each group G_j should specify a set of simple cells that are, in some sense, similar to one another.

In our system, we will use linear correlation of simple cell responses as our similarity metric, $E[a_k a_l]$, and construct groups G_j that combine similar features according to this metric.

Eugenio Culurciello

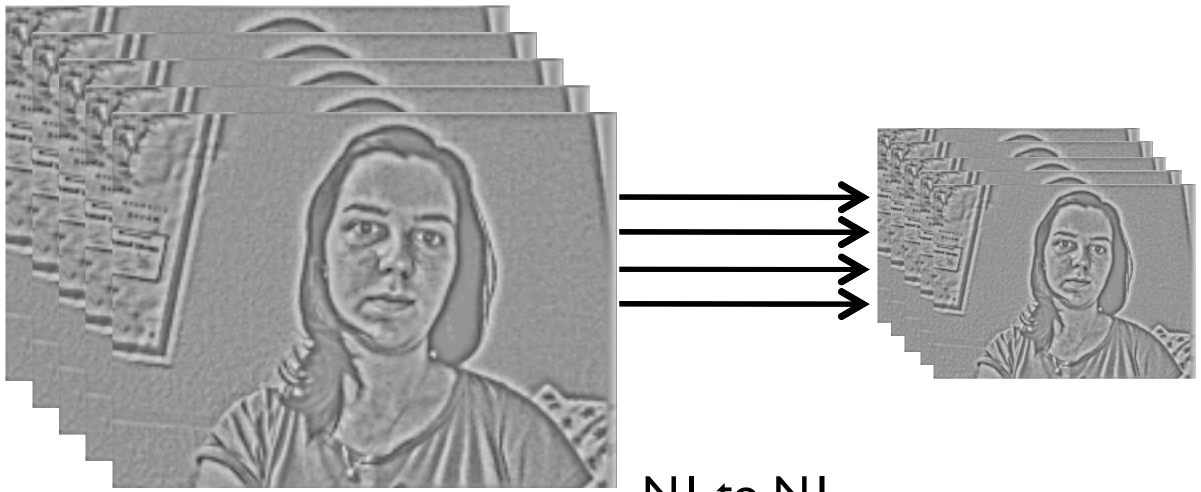
connections!



Unsupervised / Clustering Learning results:

- 1 to 1: not used
- N to M: random choices: poor
- SLAC: good compromise
- 1 to ALL (fully connected): currently best!

pooling



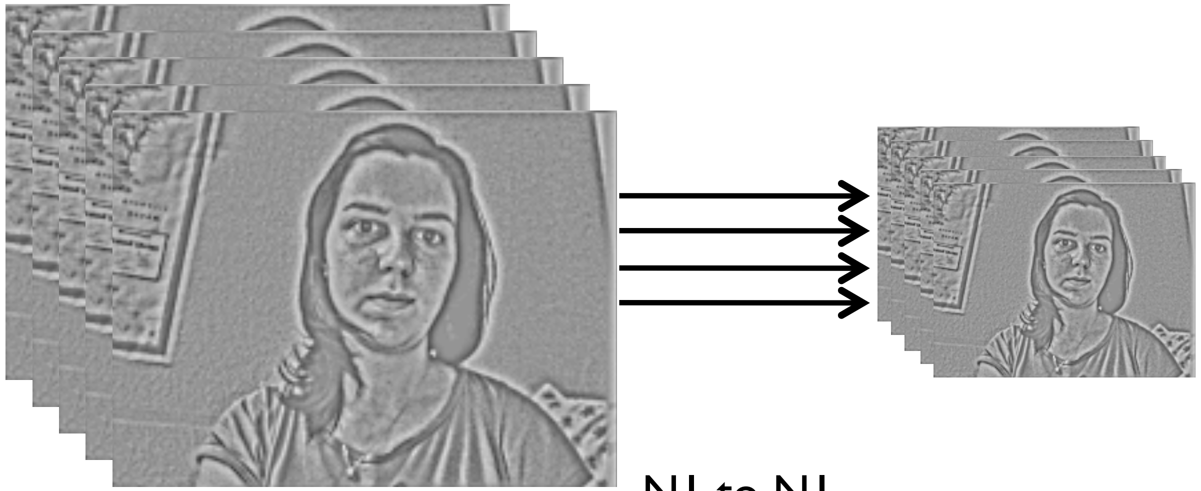
NI to NI
subsampling
or pooling
or L norm pool

Pooling Layers:

•Spatial (image 2D) or feature-wise (3rd dimension)

- average pooling: spatial
- L1, L2, Ln pooling
- Max pooling (L-inf pooling)

pooling



Unsupervised Clustering Learning results:

Spatial:

- average pooling: decent results, classic conv-net
- L1, L2, Ln pooling: Yann LeCun group suggests L2-pooling is one of the best
- Max pooling (L-inf pooling): a great choice with small filters (5x5 or below). L2-pool is best with larger filters.

Feature-wise pooling can be achieved by connection matrix between layers. Usually it is a of the average type. Max pooling is another great choice (Coates/Ng) and one I prefer also. Results are good when coupled to SpatialMaxPooling

Read further:

just some suggestions: read them and all their references and also find all newer publications by these groups on their ideas and tricks

Supervised deep-nets architectures and best choices:
Yann LeCun:
Convolutional Networks and Applications in Vision,Proc. International Symposium on Circuits and Systems (ISCAS'10), IEEE, 2010
Jürgen Schmidhuber: http://www.idsia.ch/~juergen/
Yoshua Bengio: http://www.iro.umontreal.ca/~bengioy/yoshua_en/index.html

Unsupervised deep-nets architectures and best choices:
Adam Coates, Andrew Ng:
Learning Feature Representations with K-means, Adam Coates and Andrew Y. Ng. In Neural Networks: Tricks of the Trade, Reloaded, Springer LNCS, 2012.

Geoffrey Hinton: http://www.cs.toronto.edu/~hinton/