Institut
des algorithmes
d'apprentissage
de Montréal
MILA

Bref survol d'architectures neuronales avancées:
réseaux à convolution
réseaux récurrents

Pascal Vincent
(avec quelques transparents d'Aaron Courville)

Réseaux à convolution

Plan

- Couche complètement connectée
- Couche partiellement connectée
- Notion de champ récepteur local
- Notion de partage de poids
- champ récepteur local + poids partagés => convolution
- convolution 1D, 2D
- Pooling et subsampling

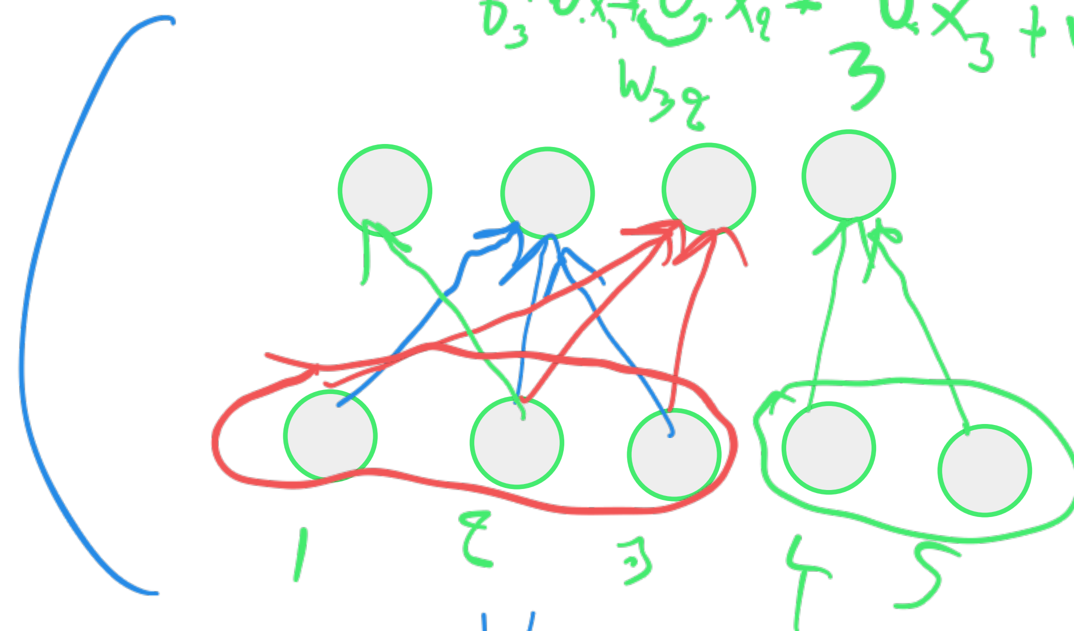
Couche complètement connectée

Couche partiellement connectée

Neurone 3 : $b_3 + 0 x_1 + \underbrace{0}_{w_{32}} x_2 + 0 x_3 + w_{34} x_4 + w_{35} x_5$

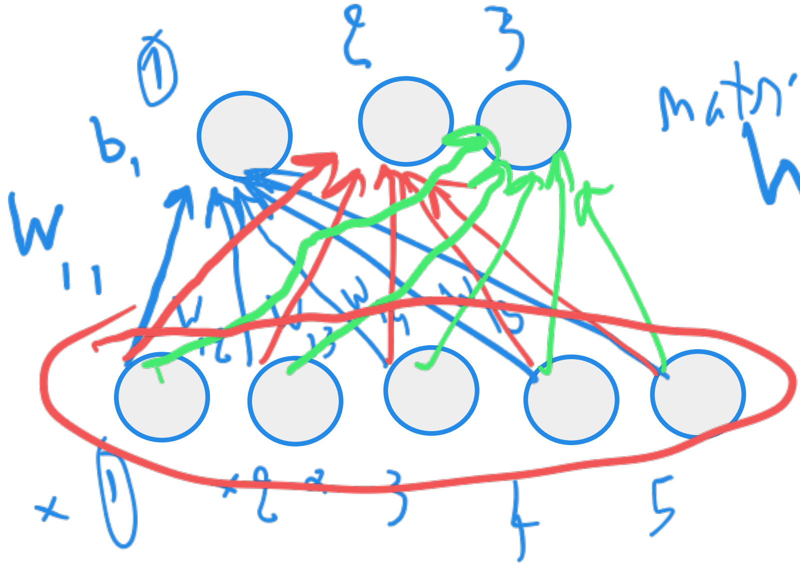
matrice creuse sparse

matrice W pour chaque connexion manquante $w_{ij} = 0$



Neurone 1 : $w_{11} x_1 + w_{12} x_2 + \dots + w_{15} x_5 + b_1$

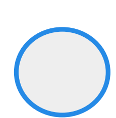
vecteur de poids $w_1 = (w_{11}, \dots, w_{15})$ et biais b_1



matrice W pleine

Champ récepteur
Réceptive field

L'ensemble des neurones d'une couche précédente qui envoient de l'information à un neurone



couche à convolution (discrète)

champ récepteur local + partage de poids =

Partage de poids

$w_{11} = w_{22} = w_{33}$

$w_{12} = w_{23} = w_{34}$

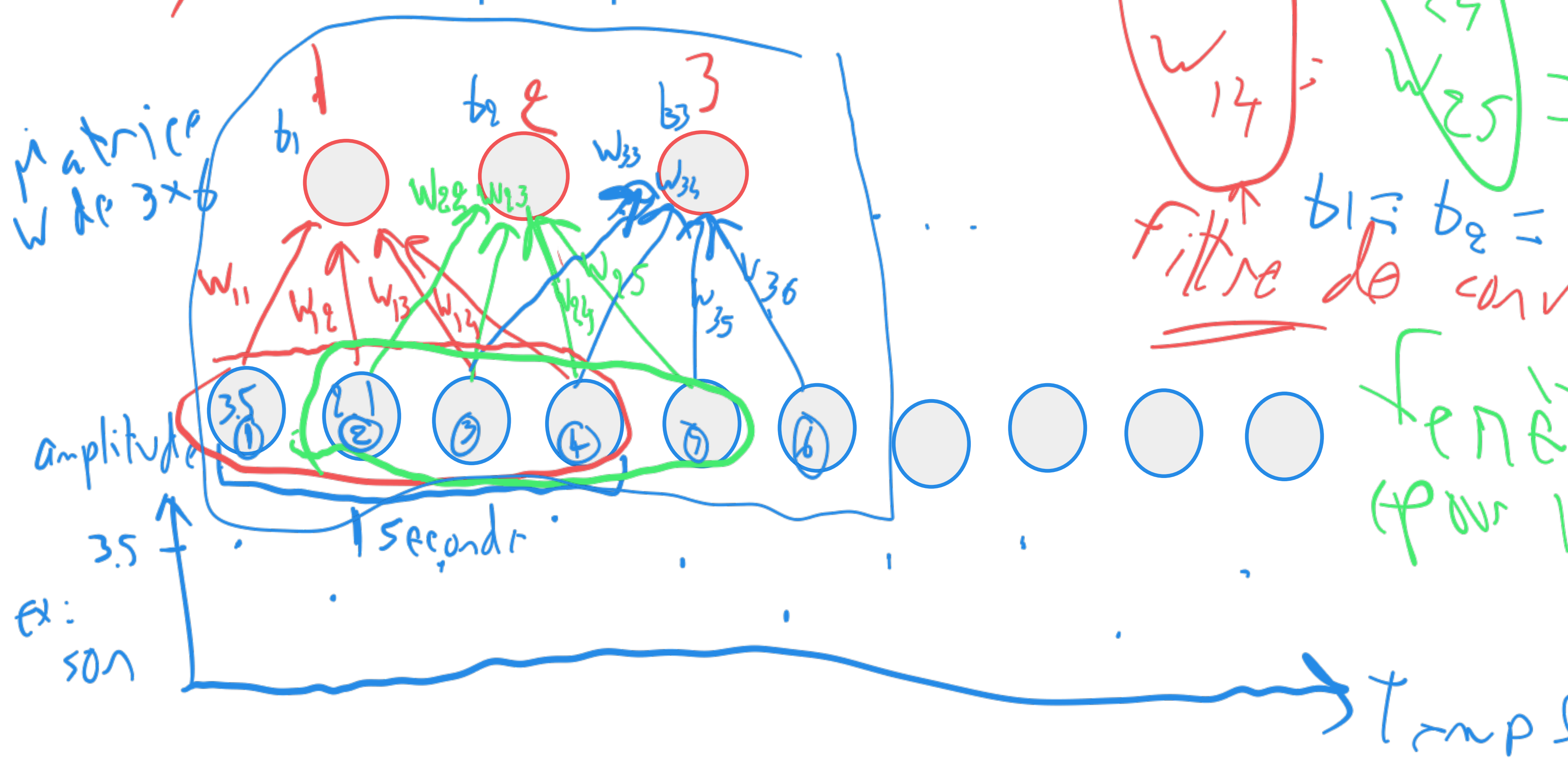
$w_{13} = w_{24} = w_{35}$

$w_{14} = w_{25} = w_{36}$

$b_1 = b_2 = b_3$

filtre de convolution

seulement 4 paramètres à apprendre pour les poids + 1 biais



fenêtre glissante (pour le champ récepteur)

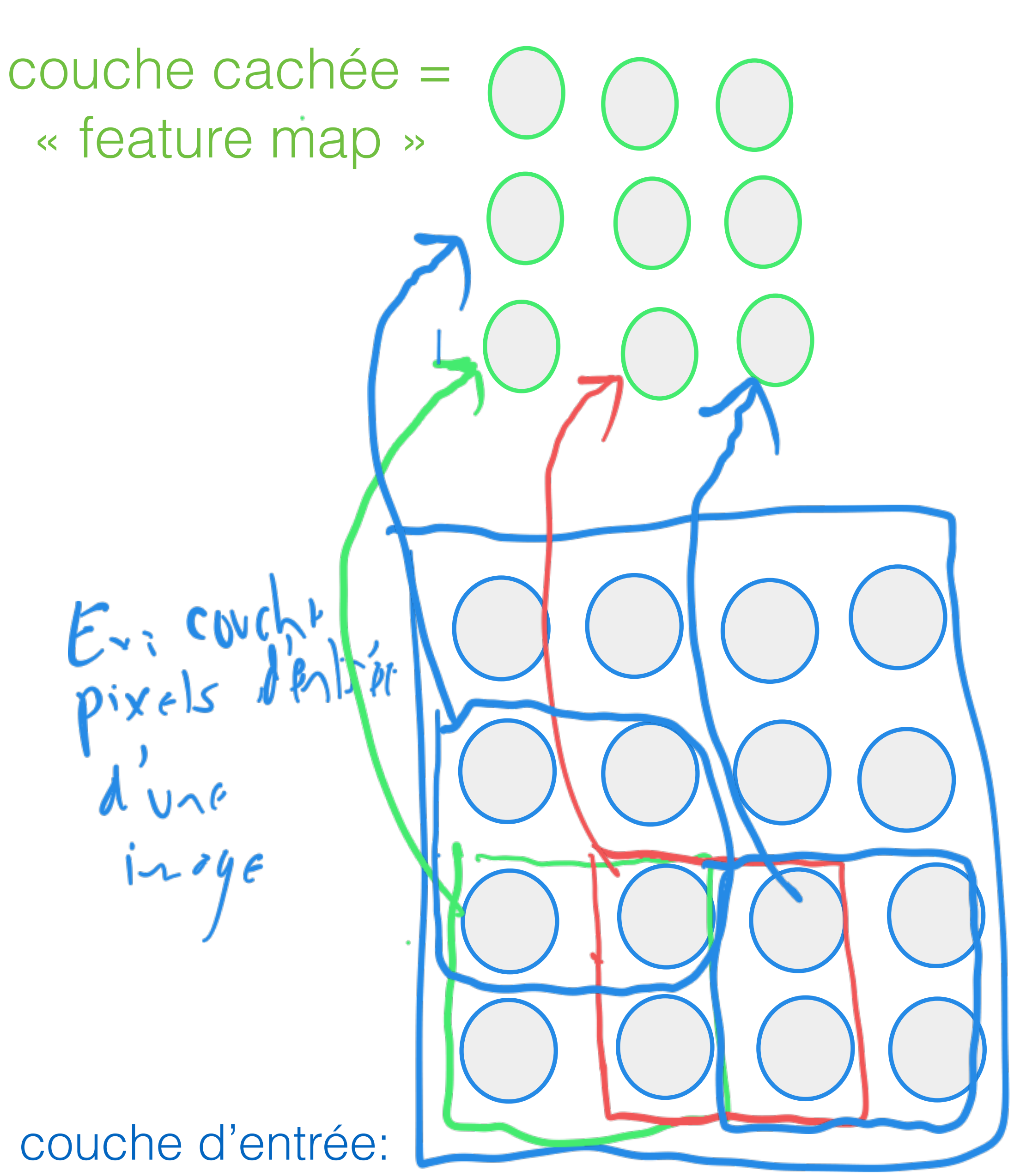
Ex précédent: couche convolution 1D. Historiquement TDNN=Time Delay Neural Network

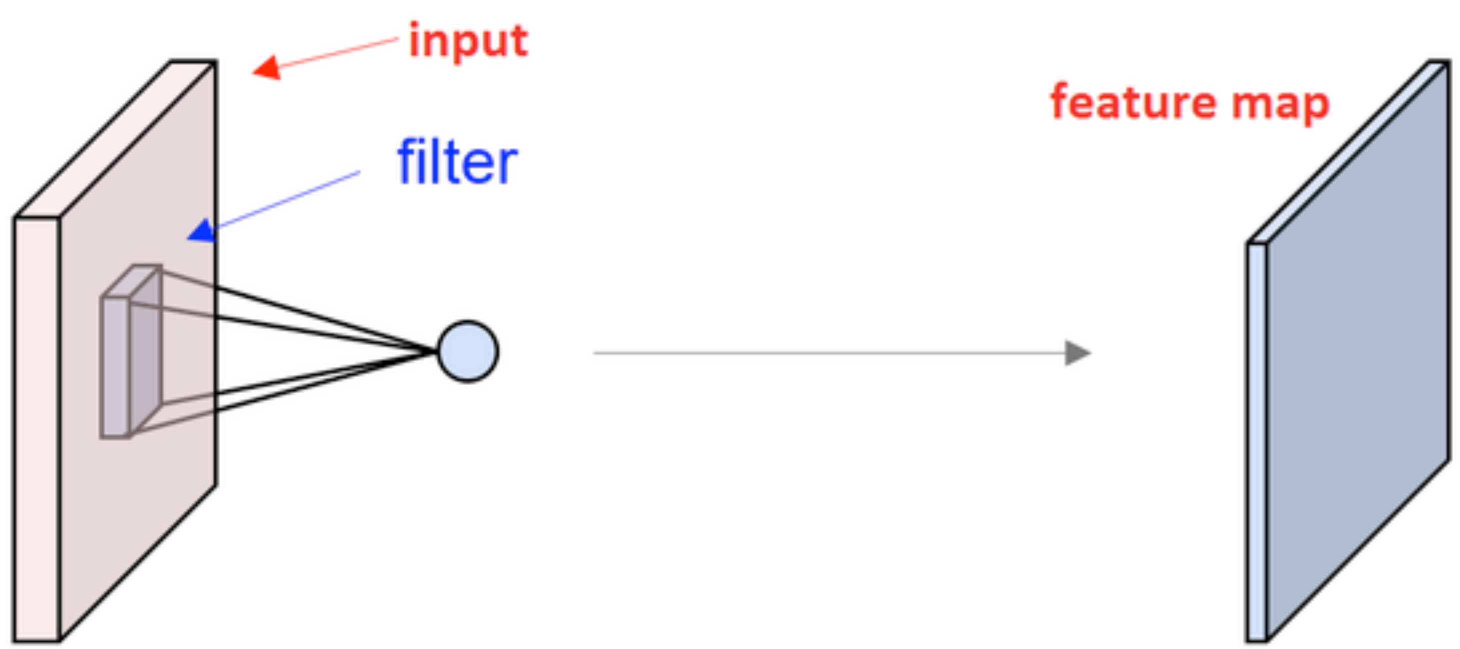
Se généralise à 1 filtre de convolution 2D

Terminologie: masque=filtre=noyau=kernel

correspond à une convolution avec $\boxed{\text{un}}$ MASQUE (filtre) de convolution de dimension 2×2 qui contient 2×2 = 4 poids à apprendre

fenêtre glissante 2D pour les champs récepteurs

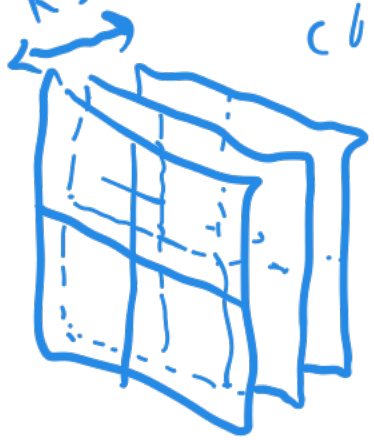



input
filter
feature map

Pour 1 couche on va utiliser

Plusieurs (k) filtres de convolution 2×2

k:3



contient $2\times2\times3$ paramètres différents à apprendre

ex : si je l'applique (les filtres de $2\times2\times3$) à une image 100×100 en entrée, ma 1ère couche cachée va avoir $99\times99\times3$ neurones ?

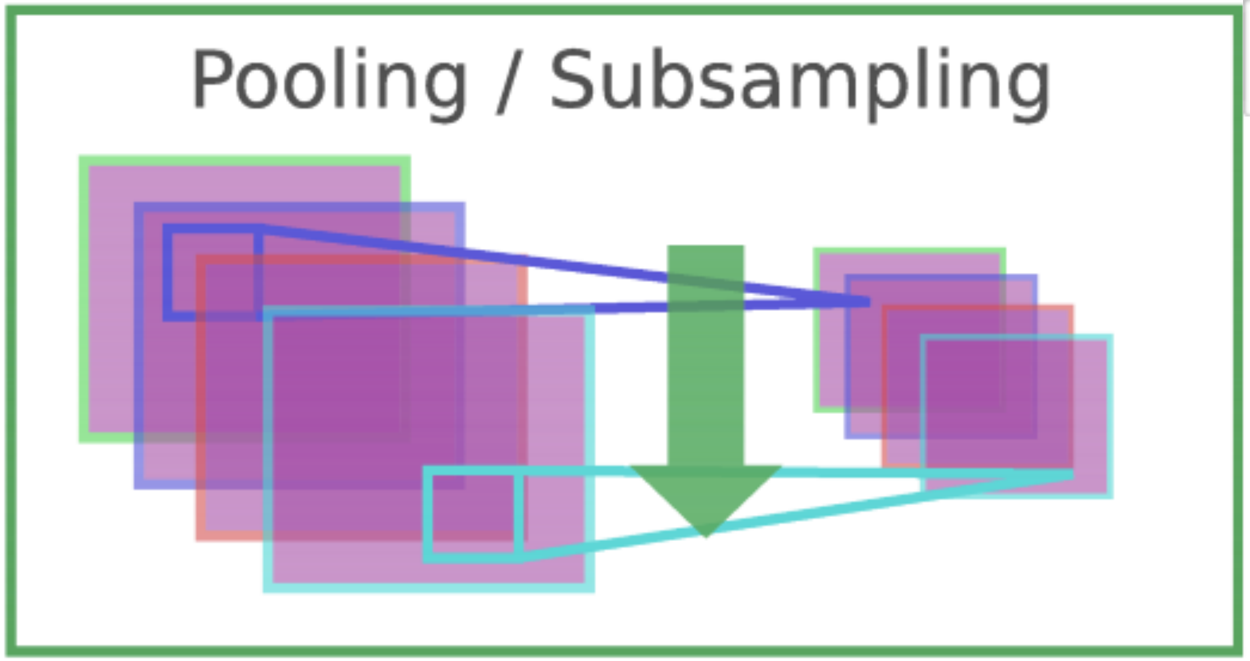
Opération de Pooling. Pour réduire la dimensionalité.

Réduire plusieurs valeurs présentes dans un champ récepteur à ne seule.

Ex: opération somme ou moyenne ou max.

Pooling and subsampling

- Third idea: pool hidden units in same neighborhood
  - pooling is performed in non-overlapping neighborhoods (subsampling)



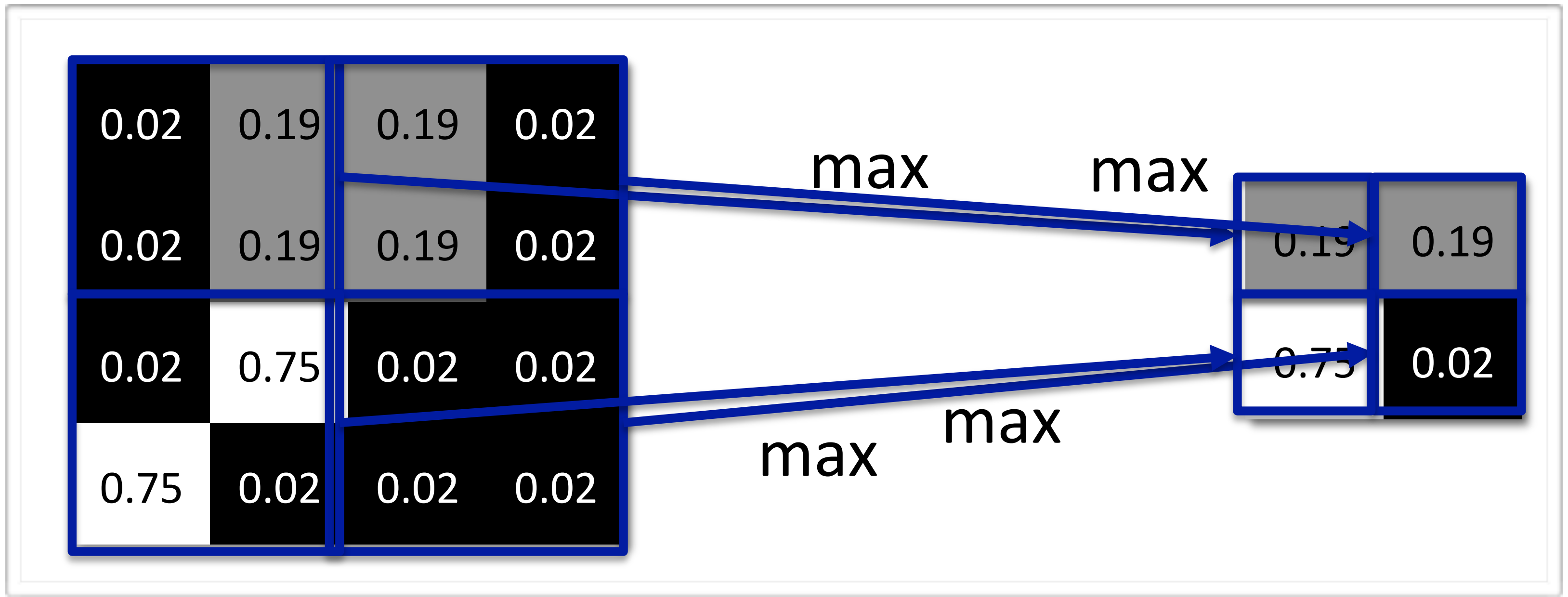
$$y_{ijk} = \max_{p,q} x_{i,j+p,k+q}$$

Jarret et al. 2009

- $x_{i,j,k}$ is value of the i^th feature map at position j,k
- p is vertical index in local neighborhood
- q is horizontal index in local neighborhood
- y_{ijk} is pooled and subsampled layer

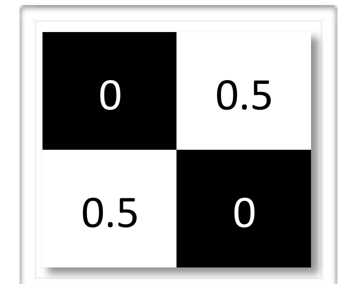
Pooling and subsampling

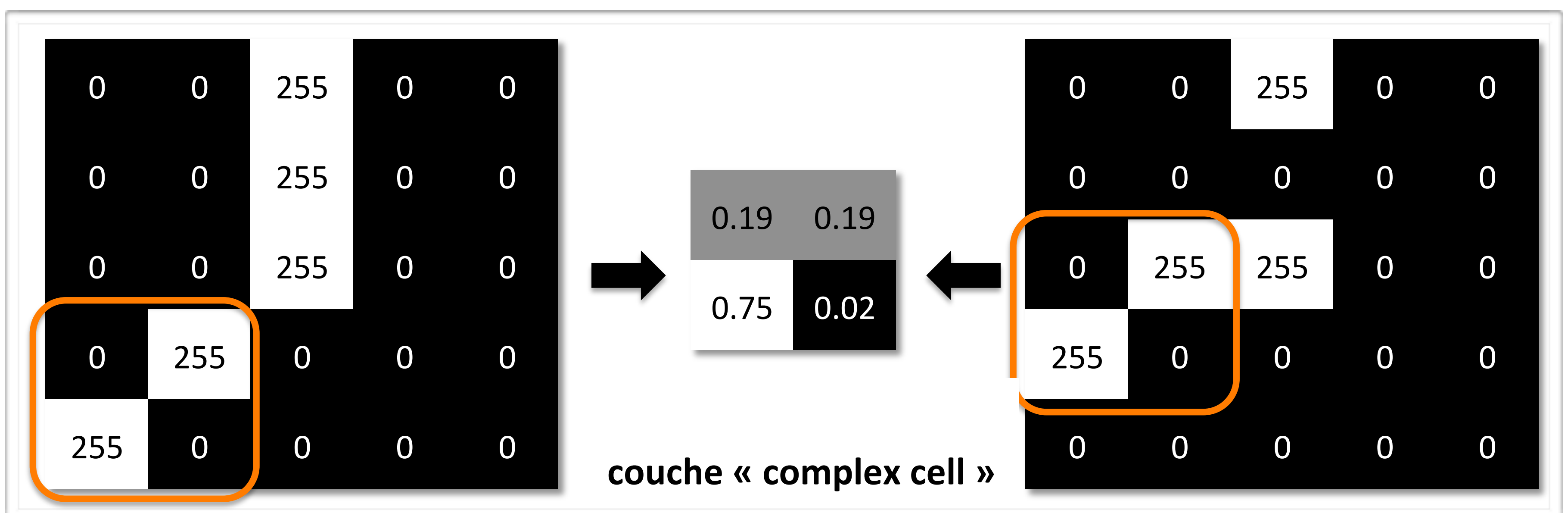
- Third idea: pool hidden units in same neighborhood
  - pooling is performed in (mostly) non-overlapping neighborhoods (subsampling)



Pooling and subsampling

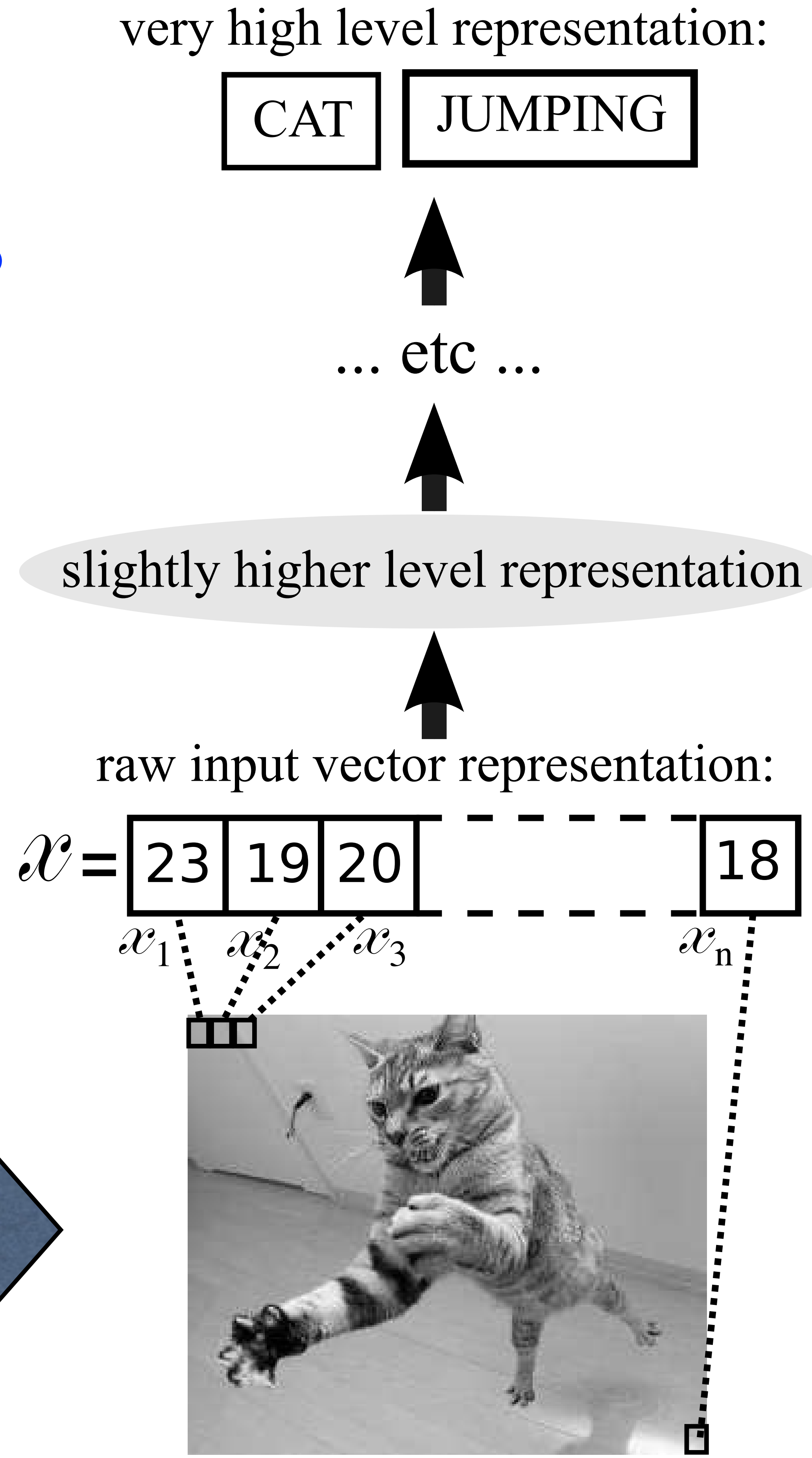
- Illustration of local translation invariance



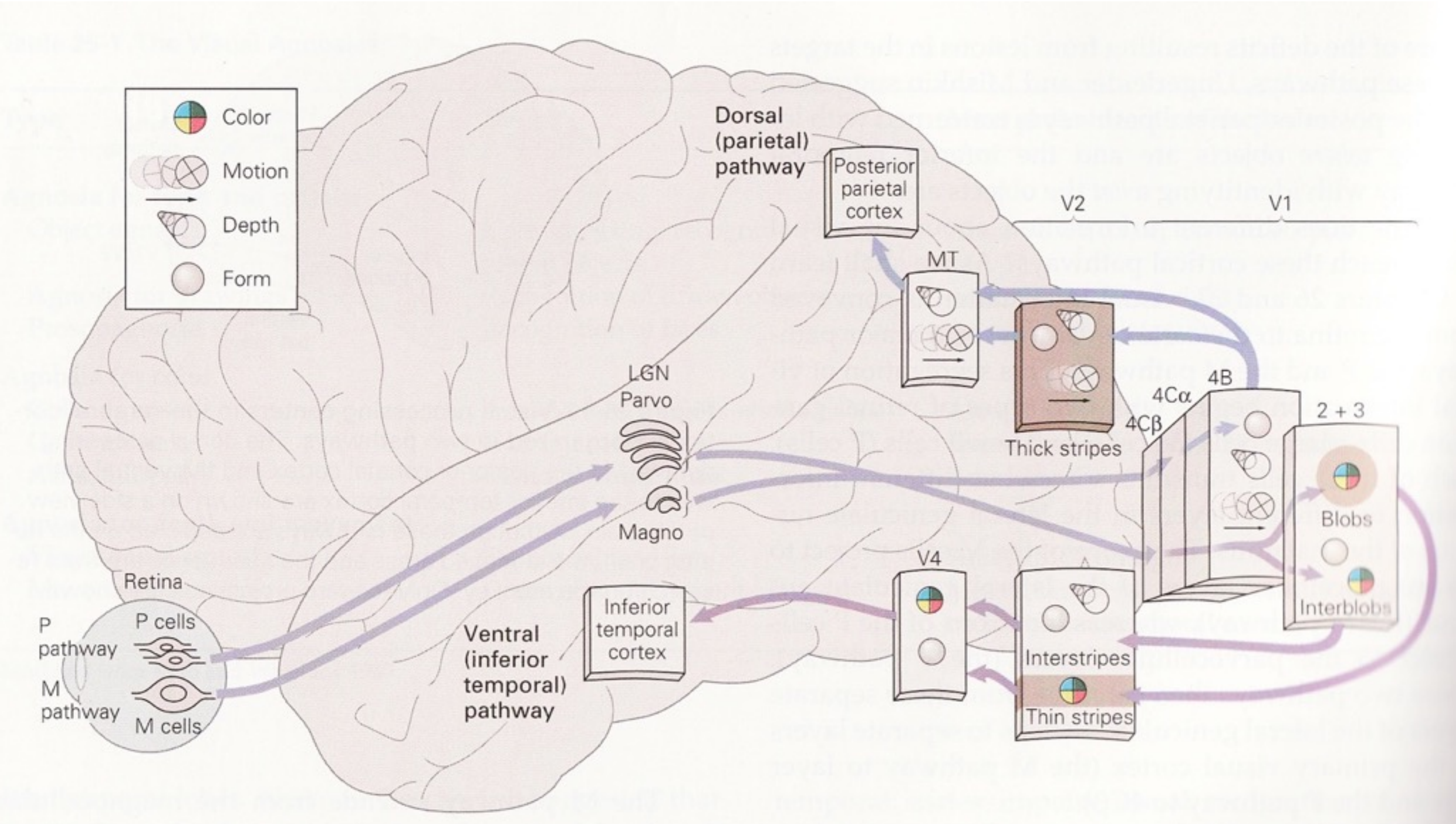


Réseaux multi-couches profonds

La notion de niveau de représentation



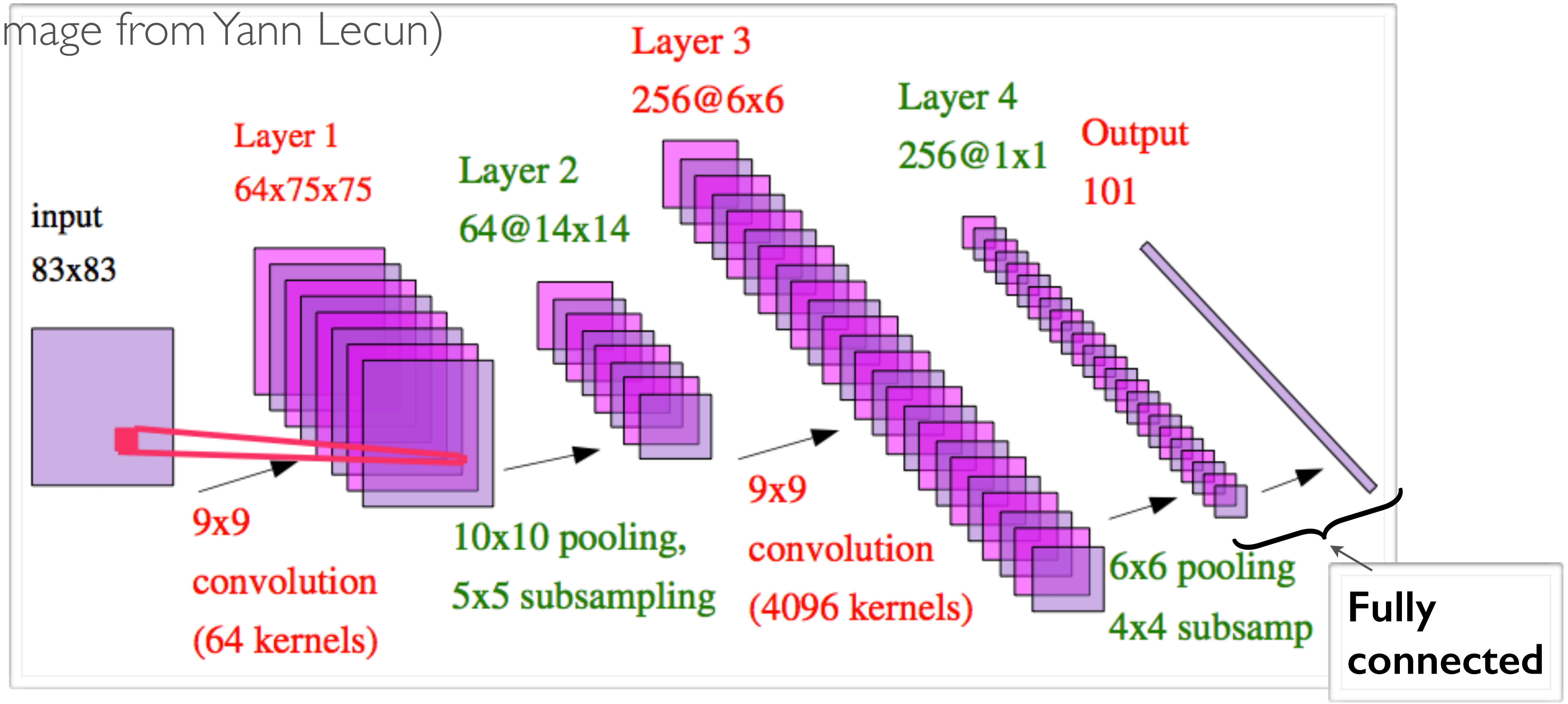
Example of a deep architecture made of multiple layers, solving complex problems...



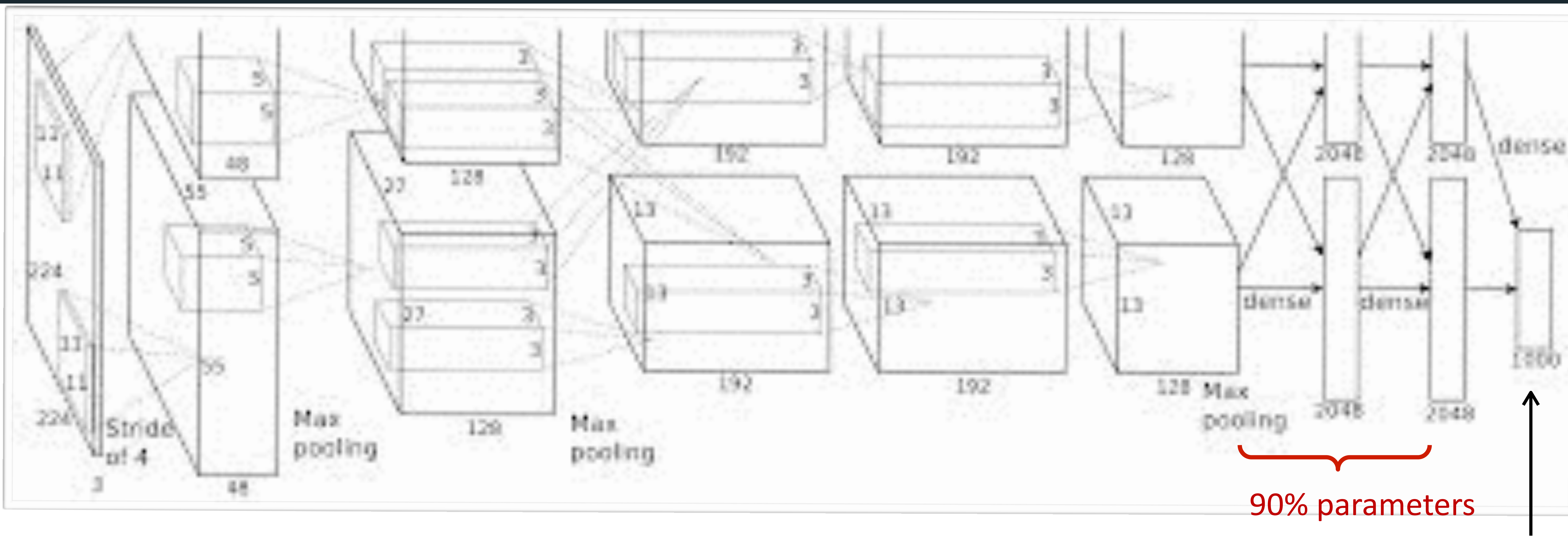
Convolutional neural network

- Convolutional neural network alternates between the convolutional and pooling layers

(image from Yann Lecun)



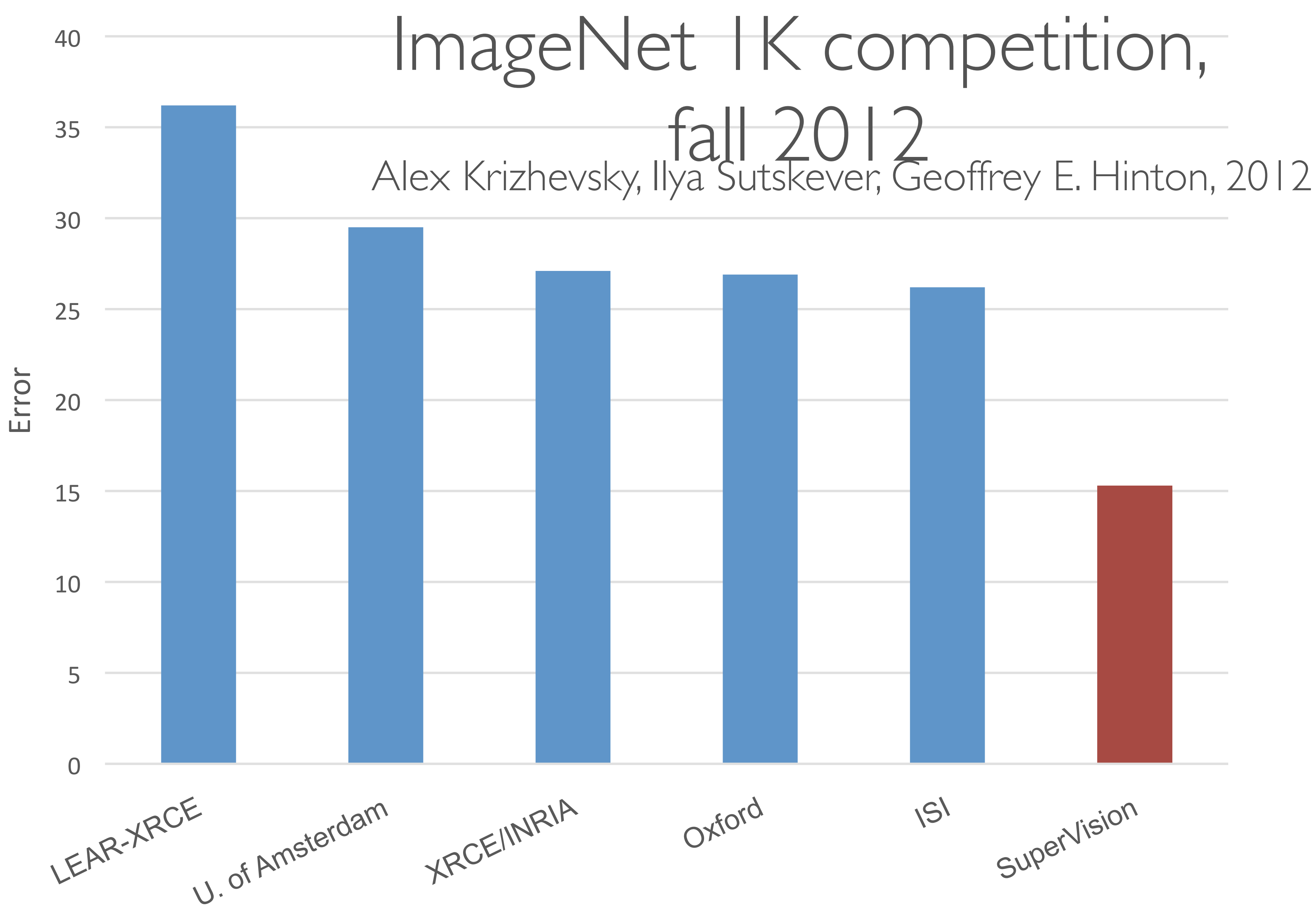
Modern CNNs



Input image

Output class

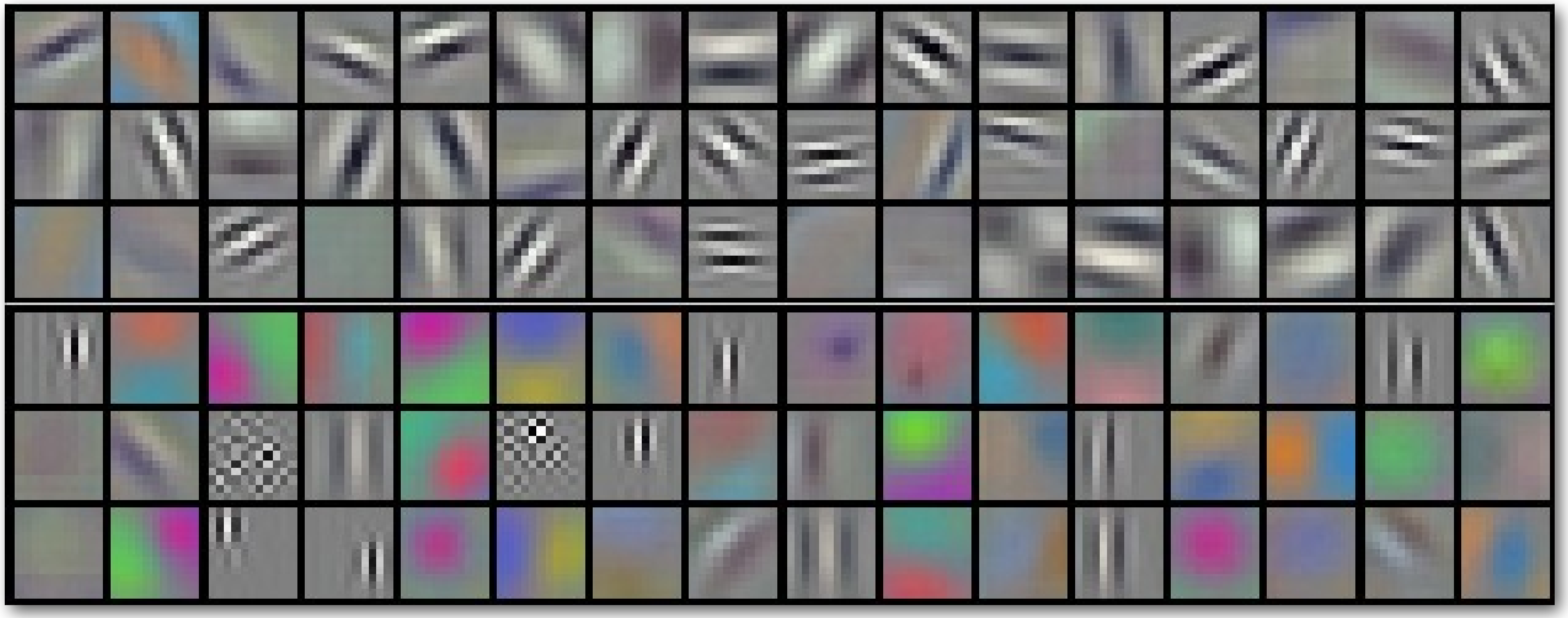
Modern CNNs


ImageNet 1K competition, fall 2012
Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, 2012
Error
40
35
30
25
20
15
10
5
0
LEAR-XRCE
U. of Amsterdam
XRCE/INRIA
Oxford
ISI
SuperVision

Modern CNNs

Filtres appris de la 1ère couche

96 low-level learned features:



CNNs: one year later

ImageNet 1K competition, fall 2013

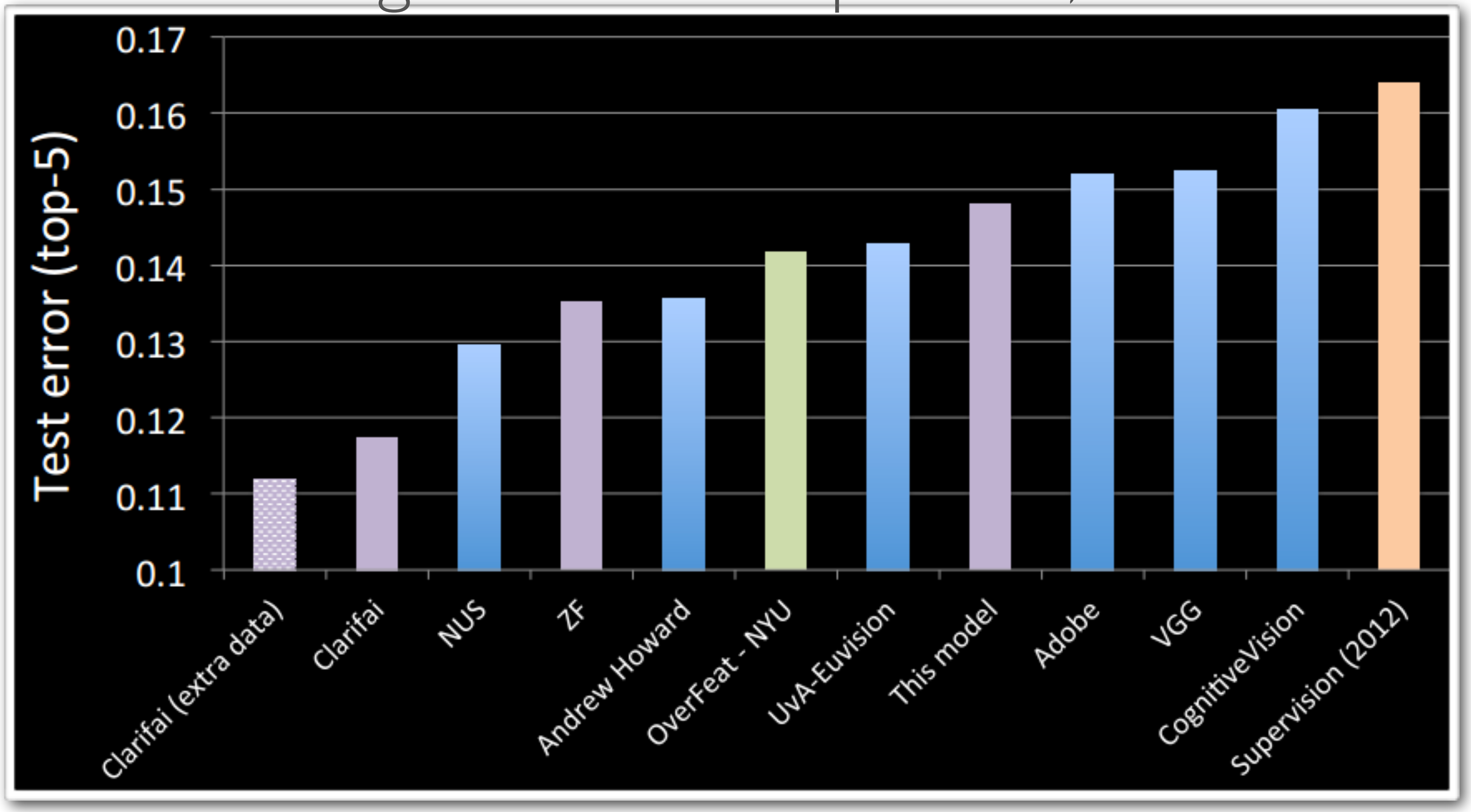


image from Li Deng, 2014

Understanding CNNs

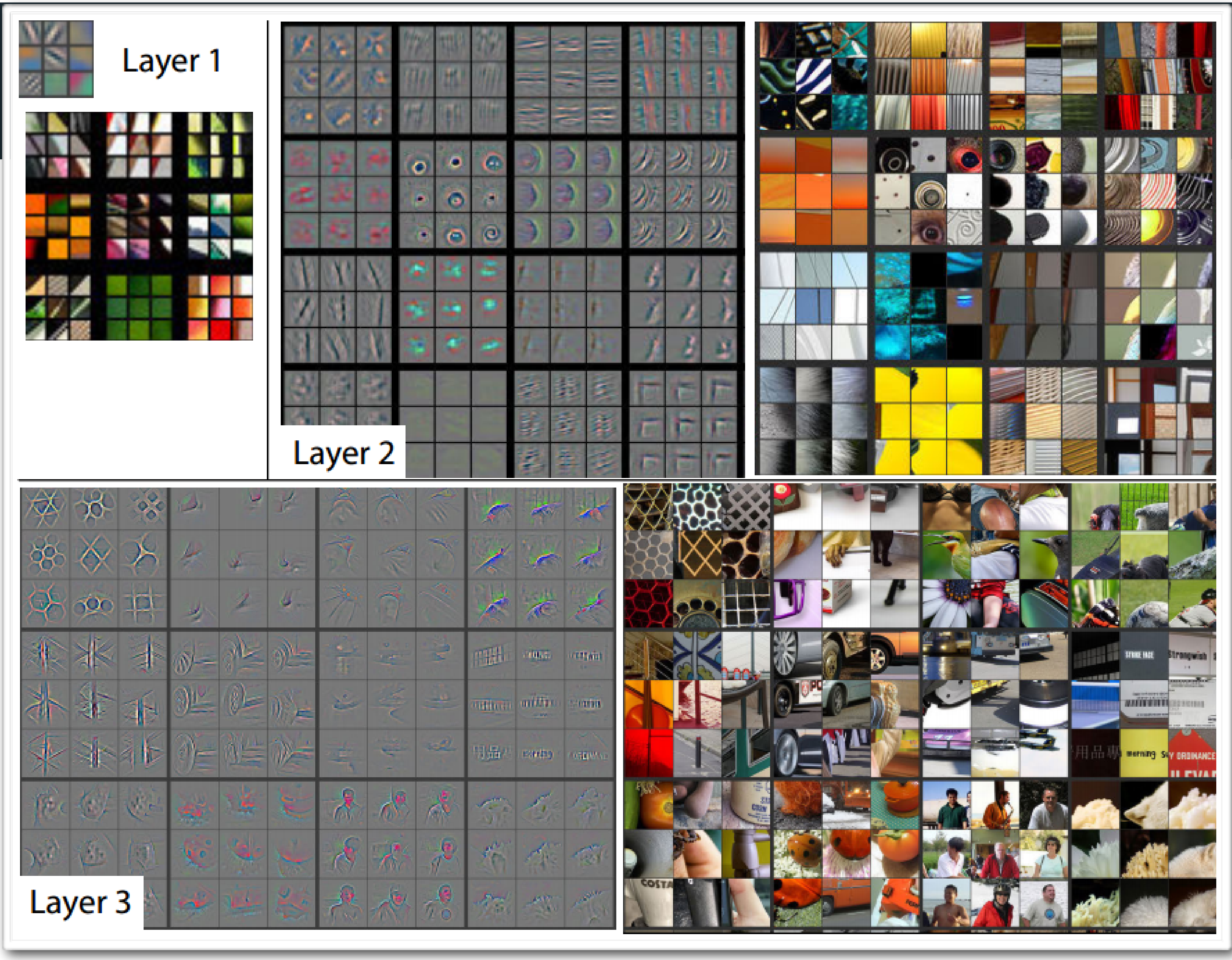
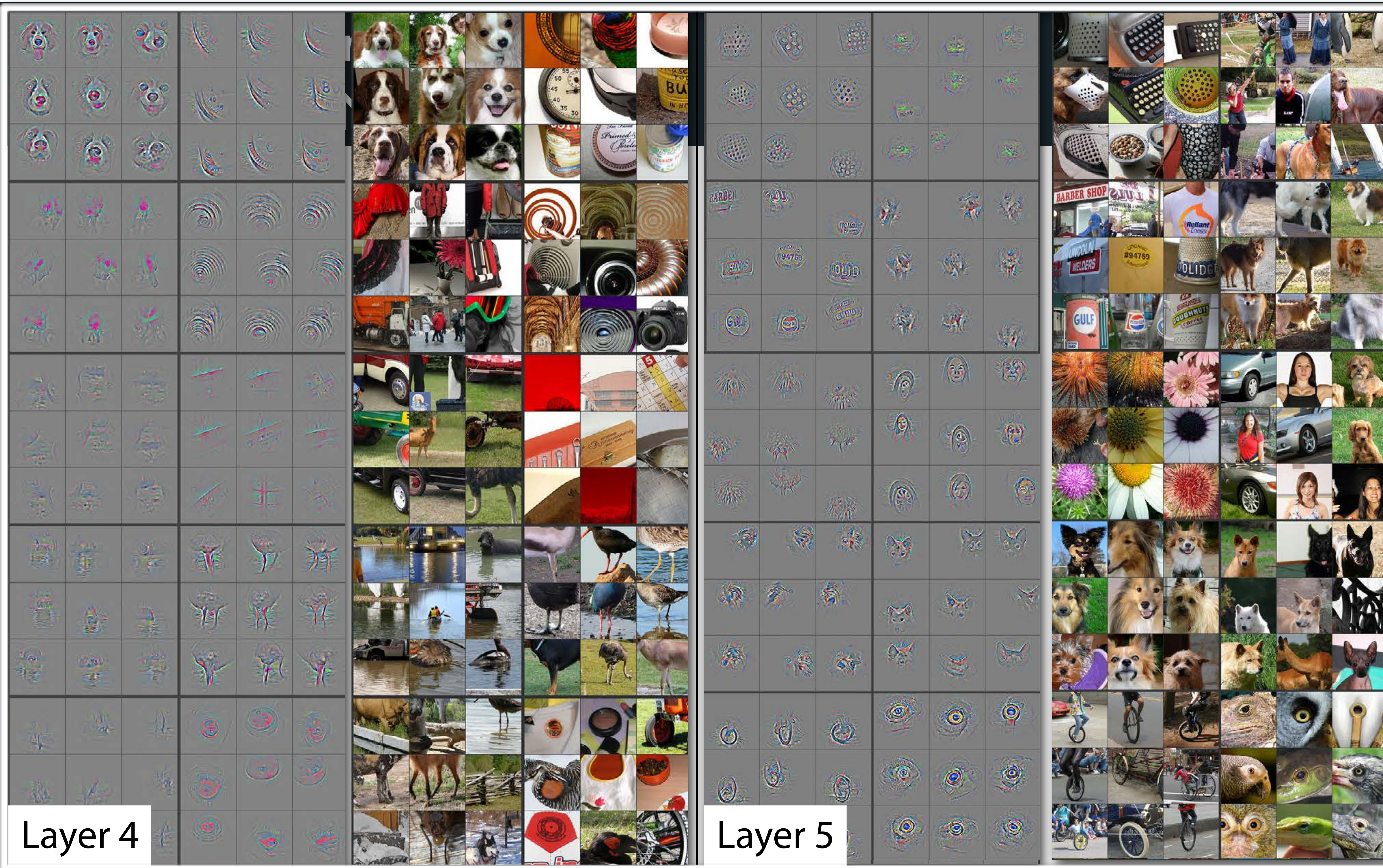


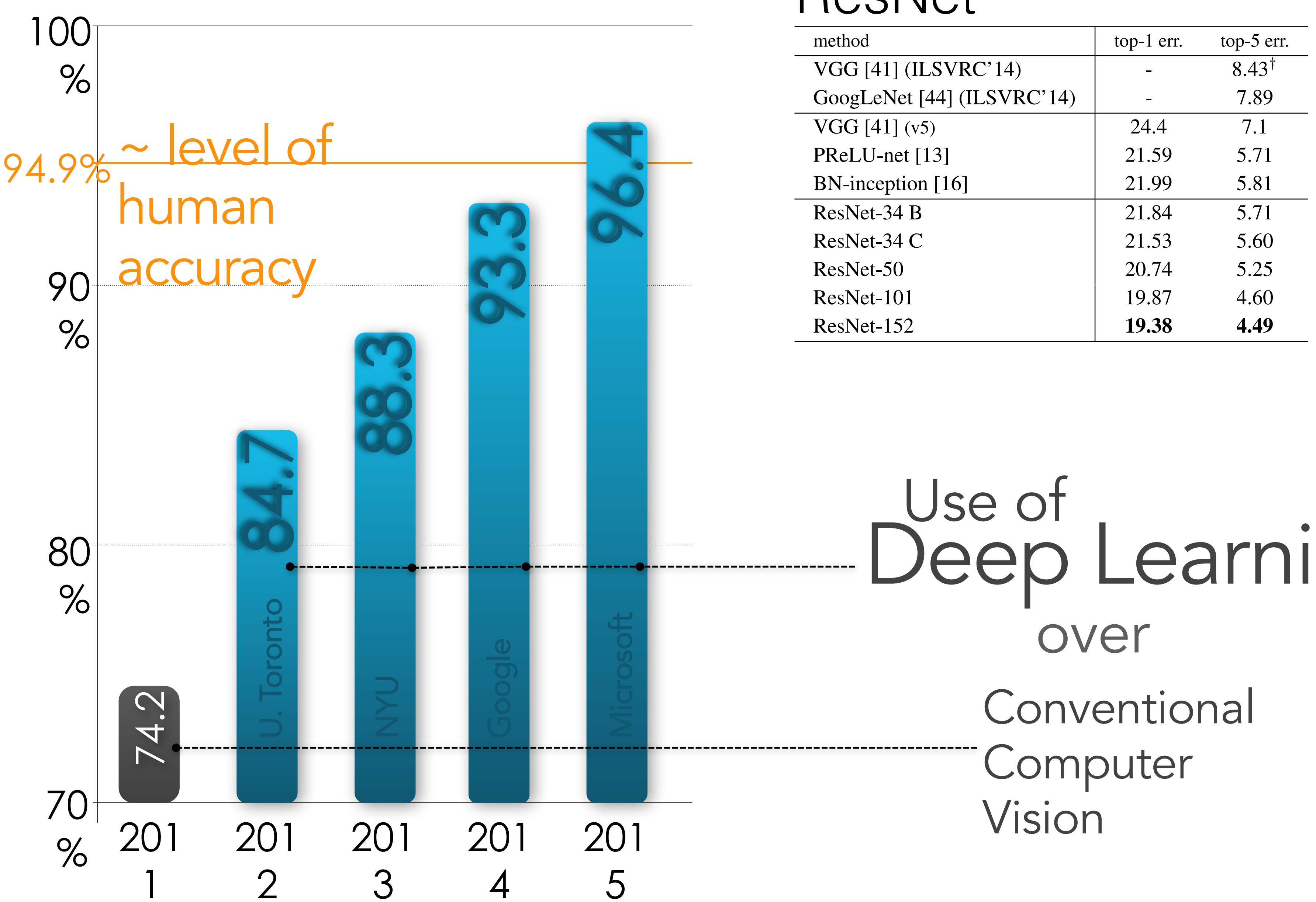
Image from Zeiler and Fergus, 2013



Zeiler and Fergus, 2013

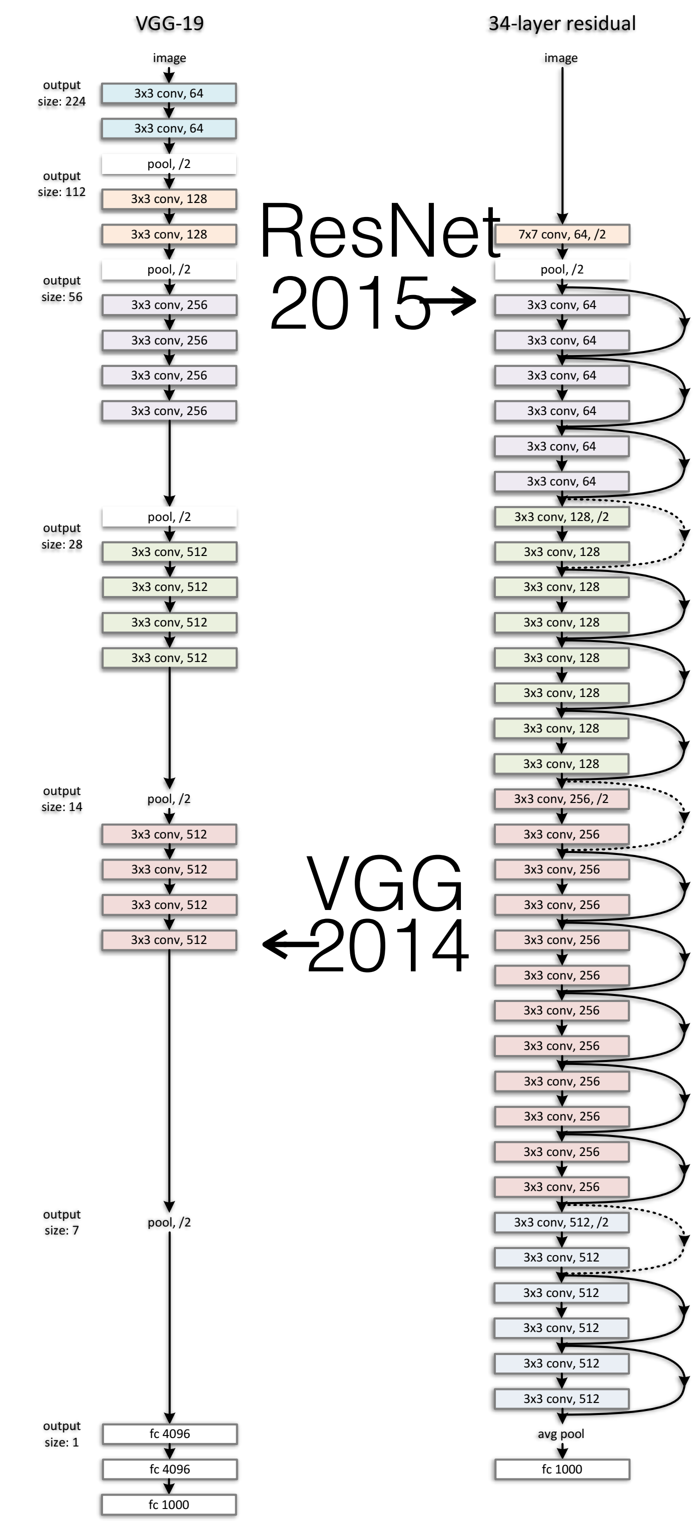
ImageNet Accuracy Still Impro

Top-5 Classification task

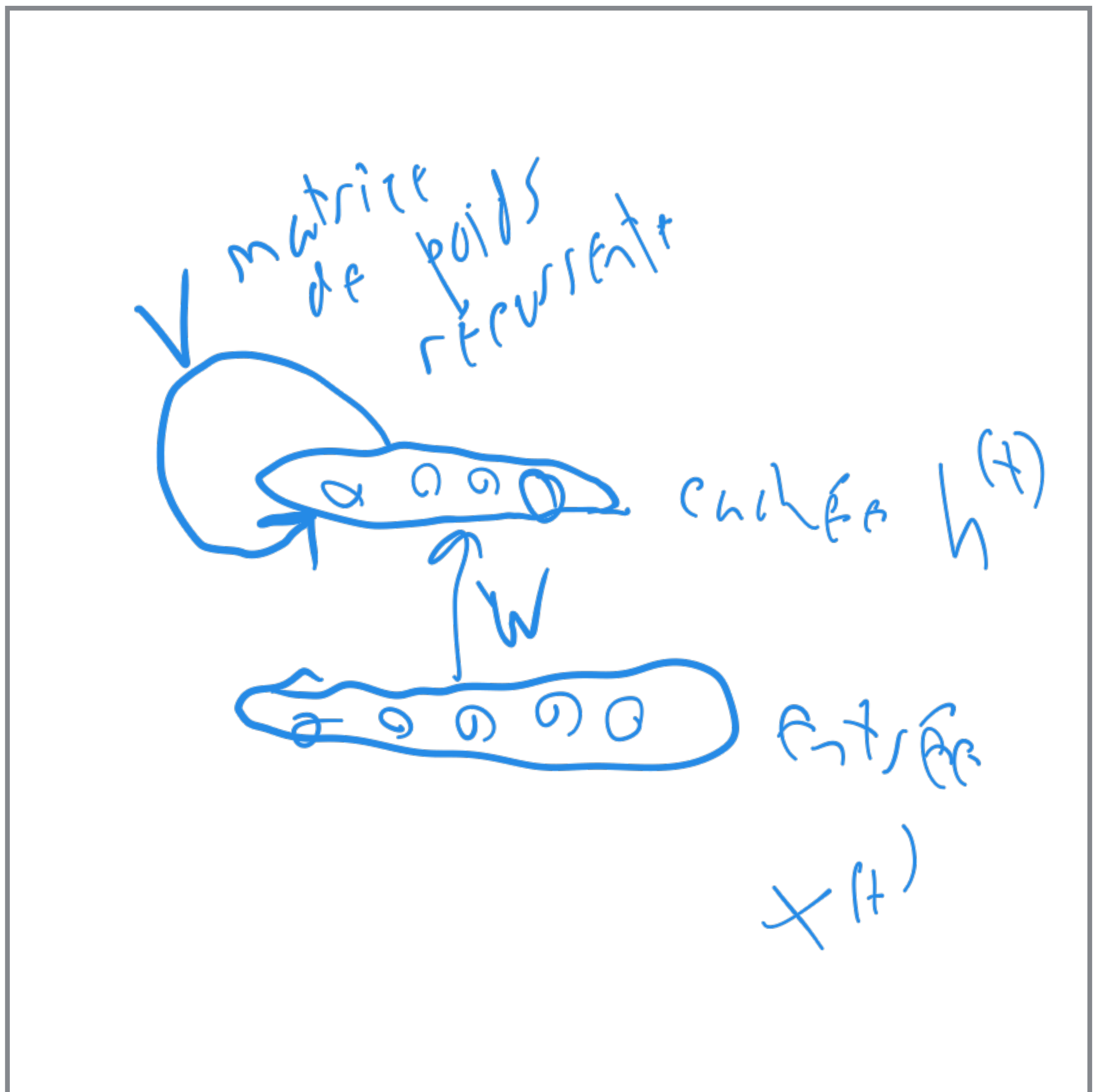


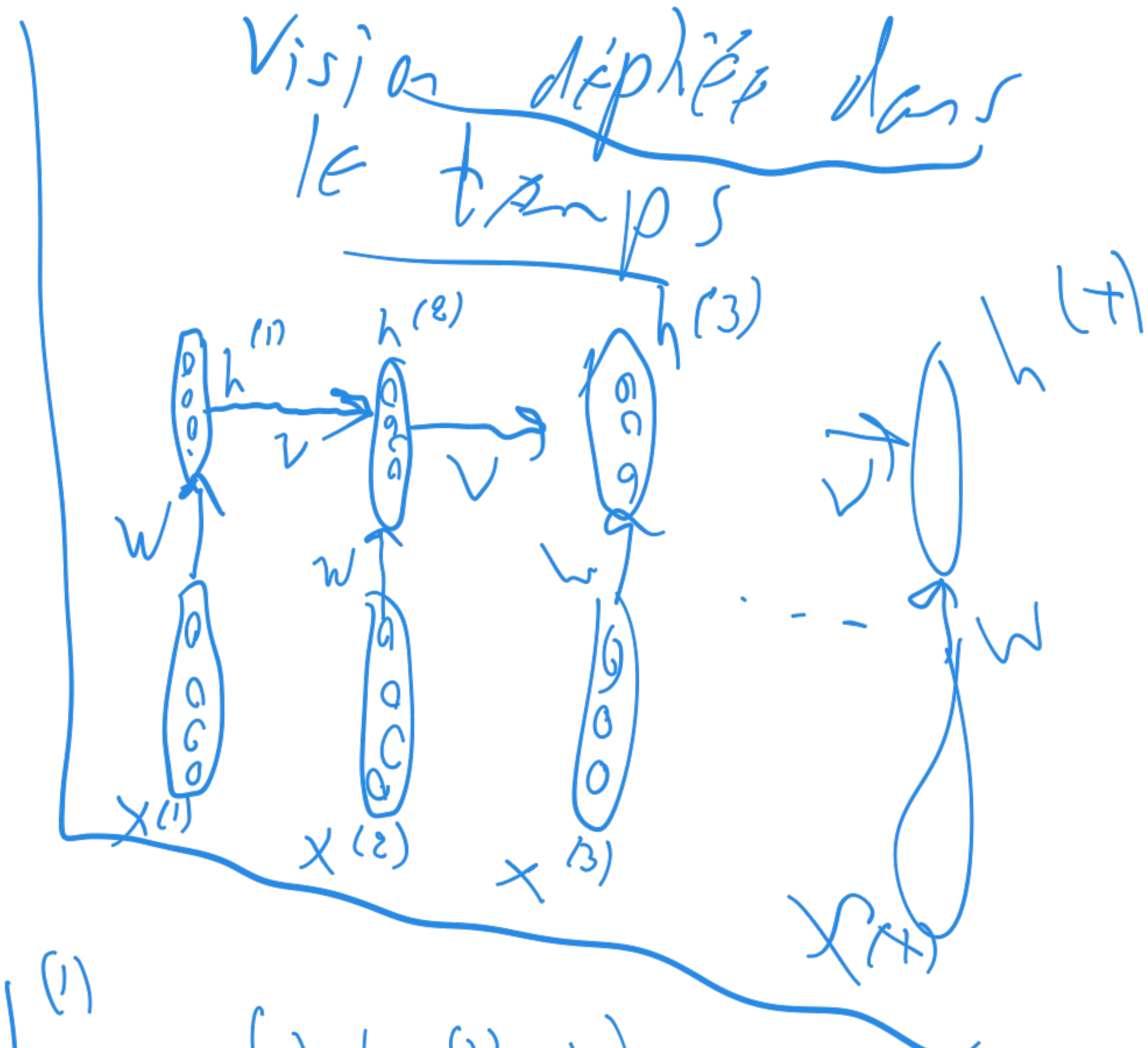
ResNet

method	top-1 err.	top-5 err.
VGG [41] (ILSVRC'14)	-	8.43†
GoogLeNet [44] (ILSVRC'14)	-	7.89
VGG [41] (v5)	24.4	7.1
PReLU-net [13]	21.59	5.71
BN-inception [16]	21.99	5.81
ResNet-34 B	21.84	5.71
ResNet-34 C	21.53	5.60
ResNet-50	20.74	5.25
ResNet-101	19.87	4.60
ResNet-152	**19.38**	**4.49**



Réseau récurrent





$$h^{(1)} = s(W x^{(1)} + b)$$

$$h^{(2)} = s(W x^{(2)} + V h^{(1)} + b)$$

$$h^{(t)} = s(W x^{(t)} + V h^{(t-1)} + b)$$