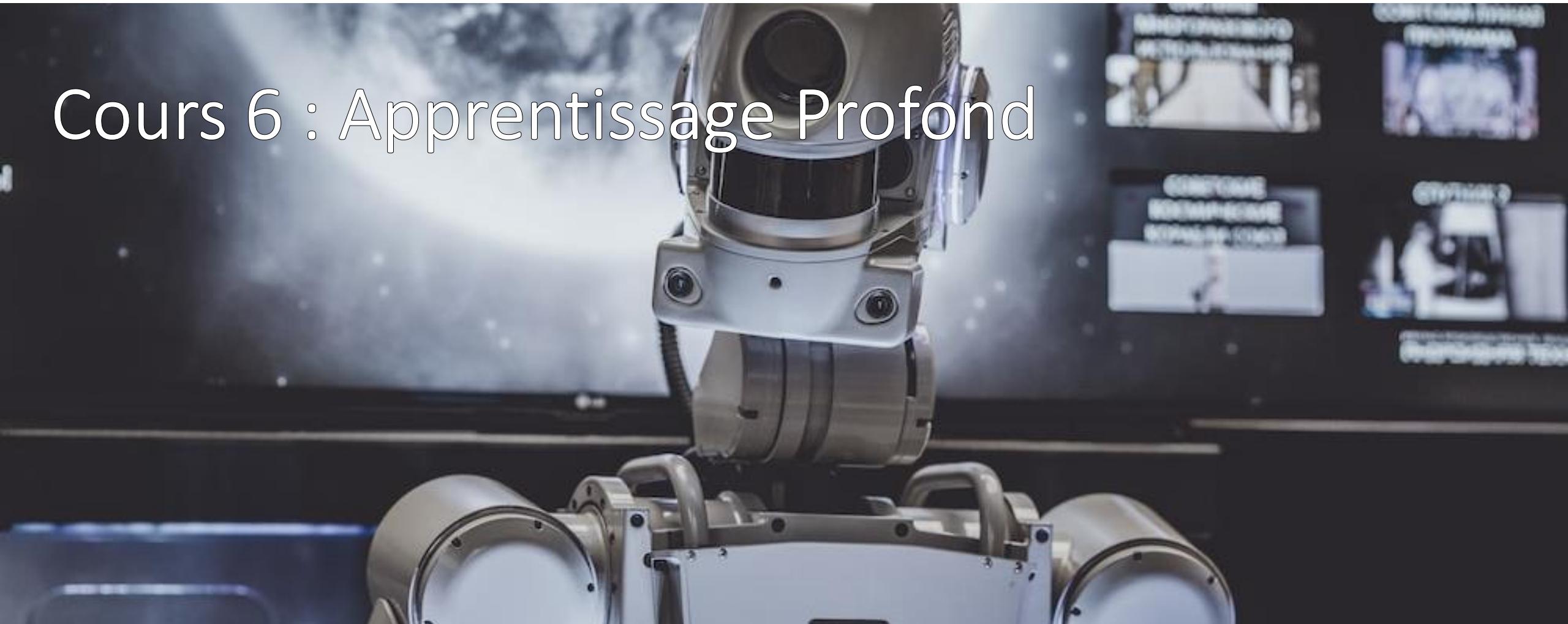


Cours 6 : Apprentissage Profond



Les réseaux de neurones profonds...



2

1

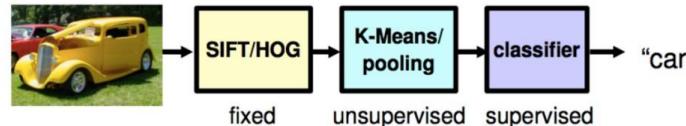
Limites des réseaux entièrement connectés

ML vs DL

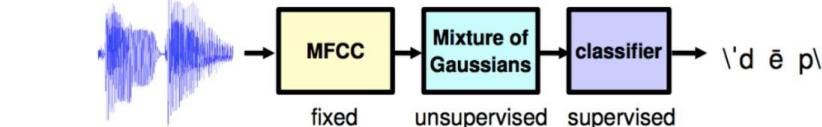
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Apprentissage profond (DL) et reconnaissance des signaux de bas niveau

VISION



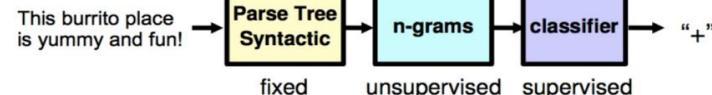
SPEECH



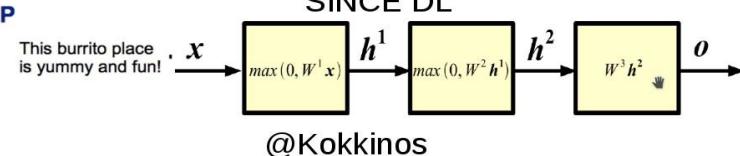
@Kokkinos

- DL : une percée pour la reconnaissance des données de signaux de bas niveau
- Avant la DL : représentations intermédiaires élaborées à la main pour chaque tâche
- Nécessité d'une expertise dans chaque domaine
- Faible niveau de sémantique dans la représentation

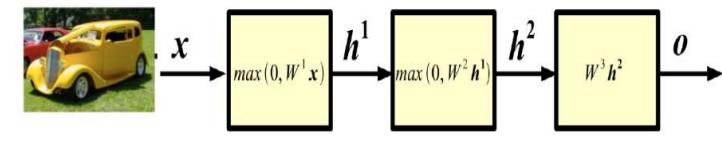
NLP



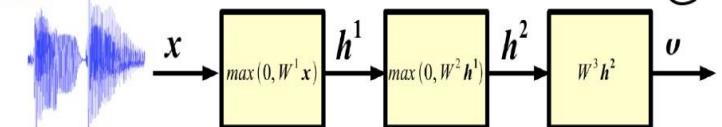
NLP



VISION

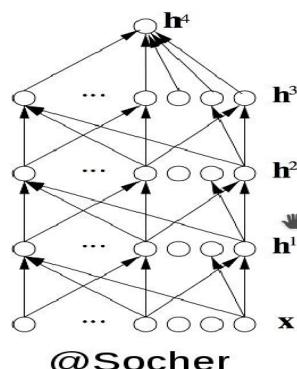


SPEECH



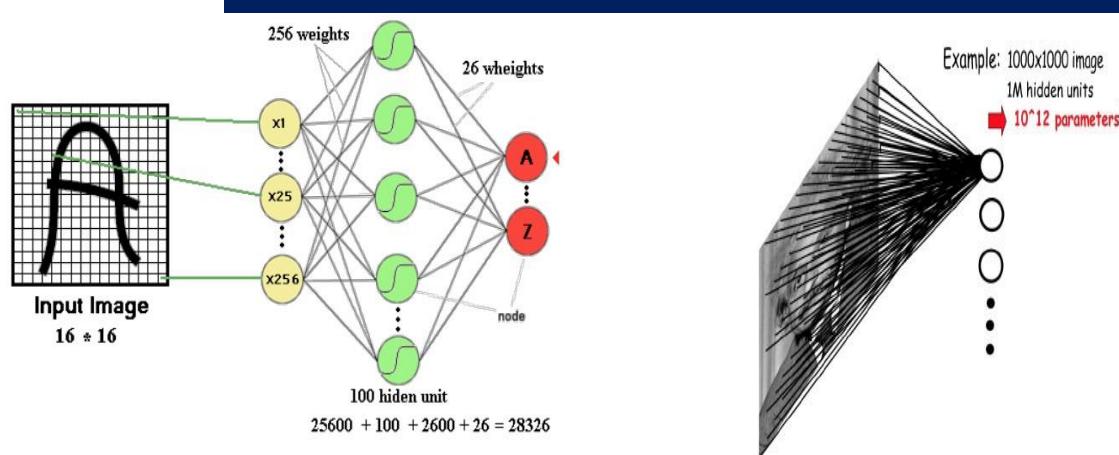
@Kokkinos

- DL : une percée pour la reconnaissance des données de signaux de bas niveau
- Depuis DL : apprentissage automatique des représentations intermédiaires
- Performances expérimentales exceptionnelles >> caractéristiques élaborées à la main Capacité à apprendre des représentations intermédiaires de haut niveau
- Méthodologie d'apprentissage commune ⇒ indépendante du domaine, pas d'expertise



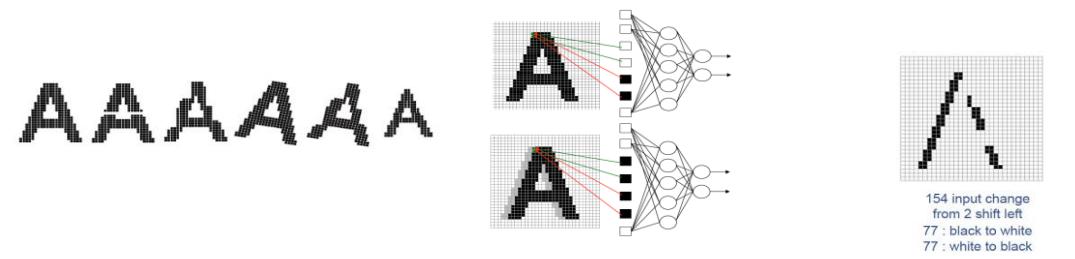
Limites des réseaux entièrement connectés

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

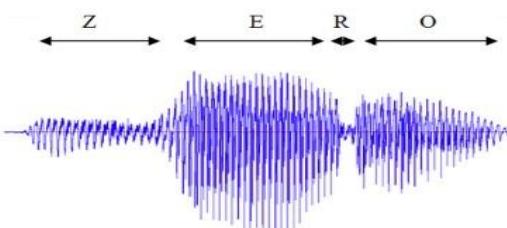


Connaissances préalables sur la structure des données \Rightarrow Exemple : Entraînement MLP pour la reconnaissance de formes (rectangle, triangle, diamant, étoile) à partir d'images en couleur

- Problème de scalabilité avec les réseaux entièrement connectés#
- Explosion des paramètres même pour une seule couche cachée !

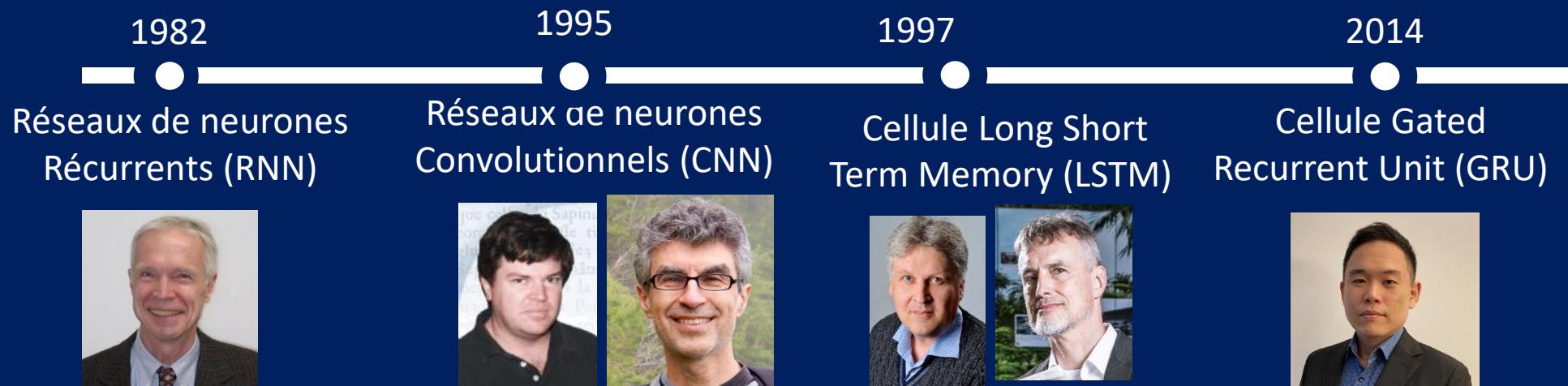


- Données de signaux : importance de la structure locale
- Signaux 1D : structure temporelle locale Données du signal
2D : structure spatiale locale



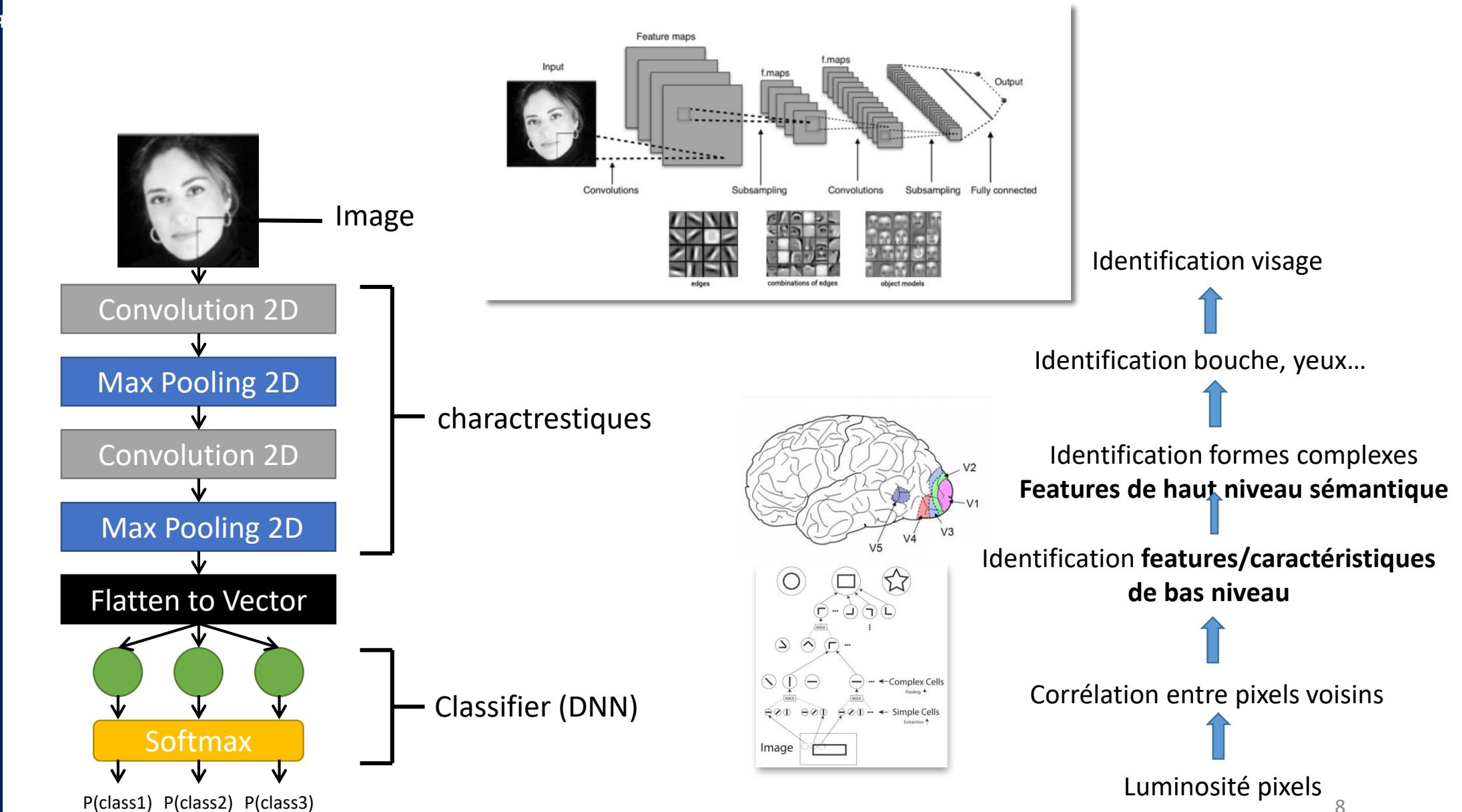
2 Les Réseaux de Neurones Convolutionnels

Recurrent Neural Networks (RNN)



Une approche biologique

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



Convolution D2 et filtrage

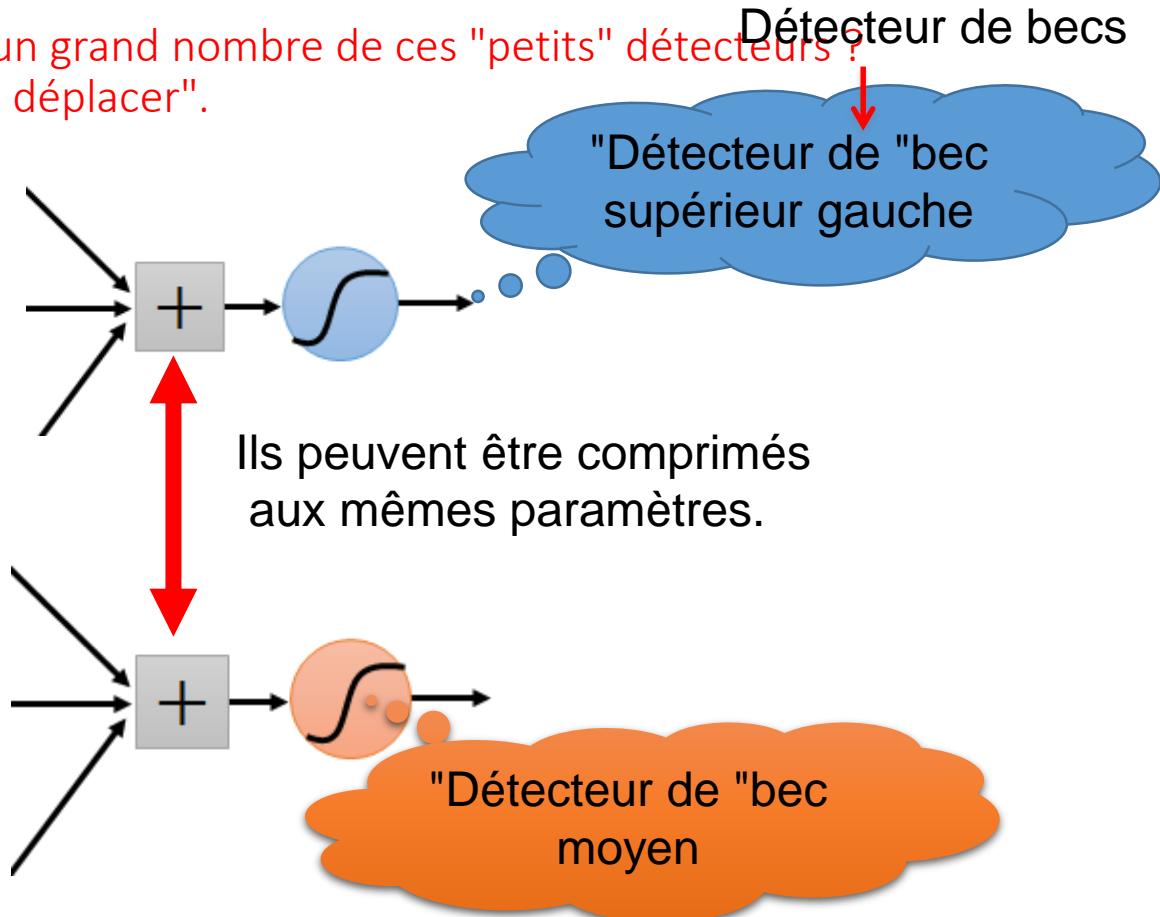
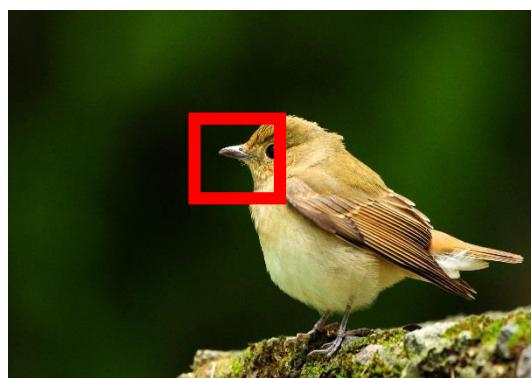
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Un CNN est un réseau neuronal composé de plusieurs couches convolutives. (et quelques autres couches). Une couche convulsive a un nombre de filtres qui effectuent des opérations convolutives.

Le même motif apparaît à différents endroits :

Ils peuvent être comprimés !

Qu'en est-il de l'entraînement d'un grand nombre de ces "petits" détecteurs ? et que chaque détecteur doit "se déplacer".



Convolution 2D (1/4)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

$$O(i, j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I((s \cdot i) + m, (s \cdot j) + n) \cdot K(m, n)$$

s : Stride (enjambement)

H : nombre de lignes dans l'image (hauteur)

W : nombre de colonnes dans l'image (largeur)

M : Nombre de lignes dans le noyau (hauteur)

N : Nombre de colonnes dans le noyau (largeur))

$$i \text{ de } 0 .. \left\lfloor \frac{H - M}{s} \right\rfloor + 1$$
$$j \text{ de } 0 .. \left\lfloor \frac{W - N}{s} \right\rfloor + 1$$

I_{00}	I_{01}	I_{02}	I_{03}	I_{04}
I_{10}	I_{11}	I_{12}	I_{13}	I_{14}
I_{20}	I_{21}	I_{22}	I_{23}	I_{24}
I_{30}	I_{31}	I_{32}	I_{33}	I_{34}

K_{00}	K_{01}	K_{02}
K_{10}	K_{11}	K_{12}
K_{20}	K_{21}	K_{22}

O_{00}	O_{01}	O_{02}
O_{10}	O_{11}	O_{12}

with stride = 1

Convolution 2D (2/4)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Image

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150



0	-1	0
-1	5	-1
0	-1	0

$$(57 \cdot 0) + (59 \cdot -1) + (58 \cdot 0) + \\ (63 \cdot -1) + (66 \cdot 5) + (75 \cdot -1) + \\ (61 \cdot 0) + (69 \cdot -1) + (89 \cdot 0) = 64$$

64		



0	-1	0
-1	5	-1
0	-1	0

$$(59 \cdot 0) + (58 \cdot -1) + (67 \cdot 0) + \\ (66 \cdot -1) + (75 \cdot 5) + (100 \cdot -1) + \\ (69 \cdot 0) + (89 \cdot -1) + (121 \cdot 0) = 62$$

64	62	



0	-1	0
-1	5	-1
0	-1	0

$$(58 \cdot 0) + (67 \cdot -1) + (82 \cdot 0) + \\ (75 \cdot -1) + (100 \cdot 5) + (124 \cdot -1) + \\ (89 \cdot 0) + (121 \cdot -1) + (150 \cdot 0) = 113$$

64	62	113



0	-1	0
-1	5	-1
0	-1	0

$$(58 \cdot 0) + (67 \cdot -1) + (82 \cdot 0) + \\ (75 \cdot -1) + (100 \cdot 5) + (124 \cdot -1) + \\ (89 \cdot 0) + (121 \cdot -1) + (150 \cdot 0) = 113$$

64	62	113

Convolution 2D (3/4)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157



0	-1	0
-1	5	-1
0	-1	0

$$(58 \cdot 0) + (67 \cdot -1) + (82 \cdot 0) + \\ (75 \cdot -1) + (100 \cdot 5) + (124 \cdot -1) + \\ (89 \cdot 0) + (121 \cdot -1) + (150 \cdot 0) = 113$$

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157



0	-1	0
-1	5	-1
0	-1	0

$$(63 \cdot 0) + (66 \cdot -1) + (75 \cdot 0) + \\ (61 \cdot -1) + (69 \cdot 5) + (89 \cdot -1) + \\ (71 \cdot 0) + (96 \cdot -1) + (126 \cdot 0) = 33$$

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157



0	-1	0
-1	5	-1
0	-1	0

$$(66 \cdot 0) + (75 \cdot -1) + (100 \cdot 0) + \\ (69 \cdot -1) + (89 \cdot 5) + (121 \cdot -1) + \\ (96 \cdot 0) + (126 \cdot -1) + (145 \cdot 0) = 54$$

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157



0	-1	0
-1	5	-1
0	-1	0

$$(75 \cdot 0) + (100 \cdot -1) + (124 \cdot 0) + \\ (89 \cdot -1) + (121 \cdot 5) + (150 \cdot -1) + \\ (126 \cdot 0) + (145 \cdot -1) + (157 \cdot 0) = 121$$

Réultat:

64	62	113

64	62	113
33		

64	62	113
33	54	

64	62	113
33	54	121

Convolution 2D (4/4)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

stride = 2

Image

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157

$$(57 \cdot 0) + (59 \cdot -1) + (58 \cdot 0) + \\ (63 \cdot -1) + (66 \cdot 5) + (75 \cdot -1) + \\ (61 \cdot 0) + (69 \cdot -1) + (89 \cdot 0) = 64$$

Résultat

64	
----	--

Image

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157

$$(58 \cdot 0) + (67 \cdot -1) + (82 \cdot 0) + \\ (75 \cdot -1) + (100 \cdot 5) + (124 \cdot -1) + \\ (89 \cdot 0) + (121 \cdot -1) + (150 \cdot 0) = 113$$

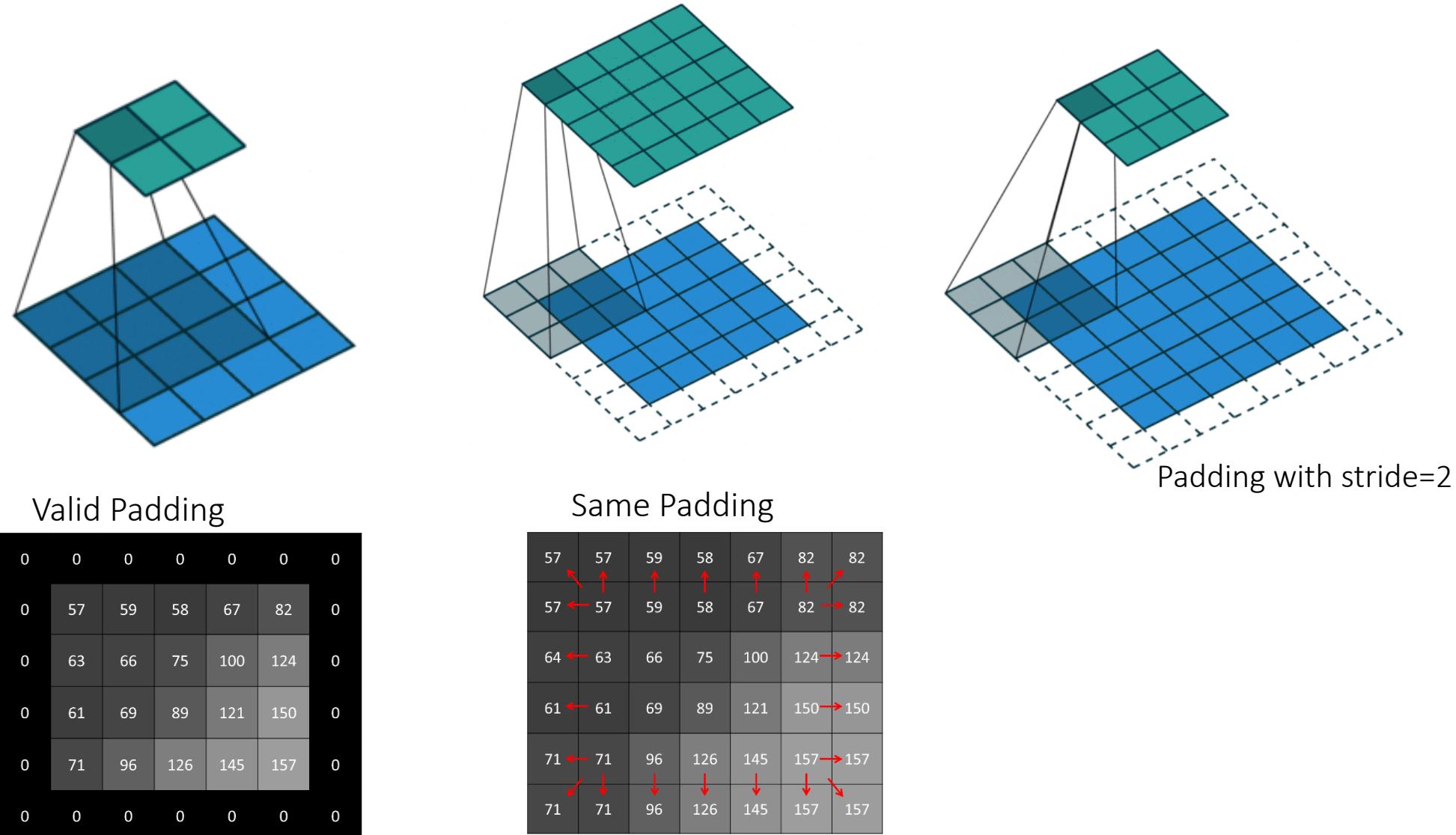
64	113
----	-----

Kernel

0	-1	0
-1	5	-1
0	-1	0

Padding 2D

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



Convolution 2D avec et sans Padding

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D
 - et filtrage
 - Convolution 2D
 - V Padding 2D
 - Conv Avec t sans Padding
 - Autres Réseaux
 - Pax Pooling
 - Flatining
 - Archi des R Profonds
 - Syst de recon de motifs
 - Exemple de RN profonds
 - ANN Vs CNN

Sans padding :

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157

Résultat

stride = 1



0	-1	0
-1	5	-1
0	-1	0

64	62	113
33	54	121

Sans padding :

- Sortie plus petite
- Les informations sur les bordures sont perdues

Avec padding :

0	0	0	0	0	0	0
0	57	59	58	67	82	0
0	63	66	75	100	124	0
0	61	69	89	121	150	0
0	71	96	126	145	157	0
0	0	0	0	0	0	0



0	-1	0
-1	5	-1
0	-1	0

163	114	89	95	219
131	64	62	113	255
102	33	54	121	255
198	214	255	255	255

Avec padding :

- Matrice de sortie plus grande
- Les informations sur les bordures sont conservées

D'autres noyaux de convolution 2D

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Gaussian blur

1/16	1/8	1/16
1/8	1/4	1/8
1/16	1/8	1/16



Sharpen

0	-1	0
-1	5	-1
0	-1	0



Emboss

-2	-1	0
-1	1	1
0	1	2



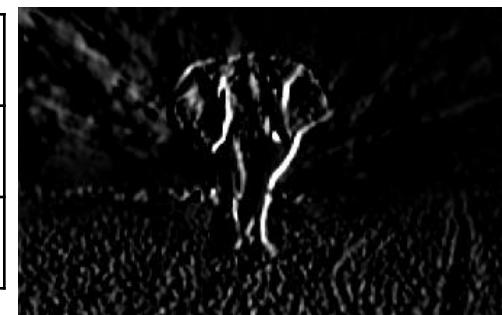
Outline

-1	-1	-1
-1	8	-1
-1	-1	-1



Left Sobel

1	0	-1
2	0	-2
1	0	-1



Top Sobel

1	2	1
0	0	0
-1	-2	-1



Max Pooling (1/3)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Pourquoi Le Pooling ?

$$O(i,j) = \sum_{m=1}^{M/S} \left\lfloor \frac{I_n - P}{S} \right\rfloor + 1$$

- Le sous-échantillonnage des pixels ne modifie pas l'objet oiseau



Sous-échantillonnage

oiseau



Nous pouvons sous-échantillonner les pixels pour réduire la taille de l'image.



moins de paramètres pour caractériser l'image

Max Polling (2/3)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds

Exemple

Fenetre 2x2
Stride: 2

57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157

$$(57 + 59 + 63 + 66) / 4 = 61.25$$



57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157

$$(58 + 67 + 75 + 100) / 4 = 75$$



57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157

$$(61 + 69 + 71 + 96) / 4 = 74.25$$



57	59	58	67	82
63	66	75	100	124
61	69	89	121	150
71	96	126	145	157

$$(89 + 121 + 126 + 145) / 4 = 120.25$$



Average Pooling

61	

61	75

61	75
74	

61	75
74	120

Max Pooling

66	

66	100

66	100
96	

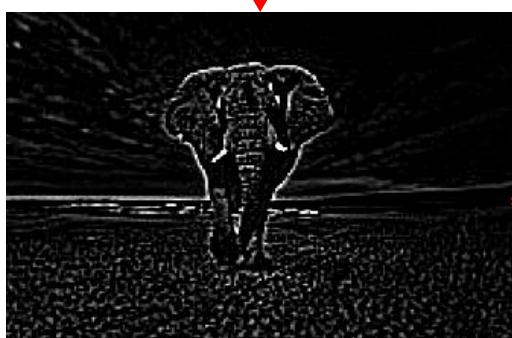
66	100
96	145

Max Polling(2/3)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Exemple

Original image
(200x130)



Filtered image
(198x128)

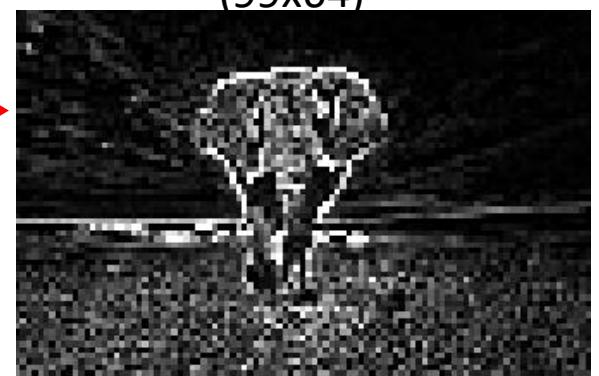
Average
Pooling

Max
Pooling

(99x64)



(99x64)



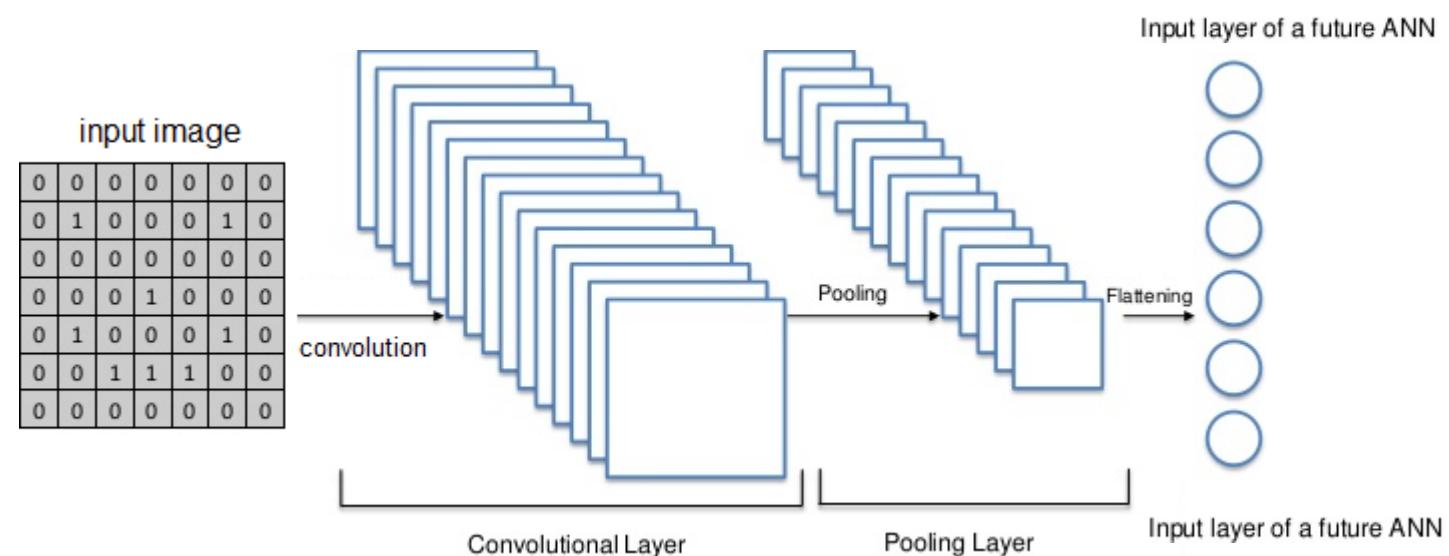
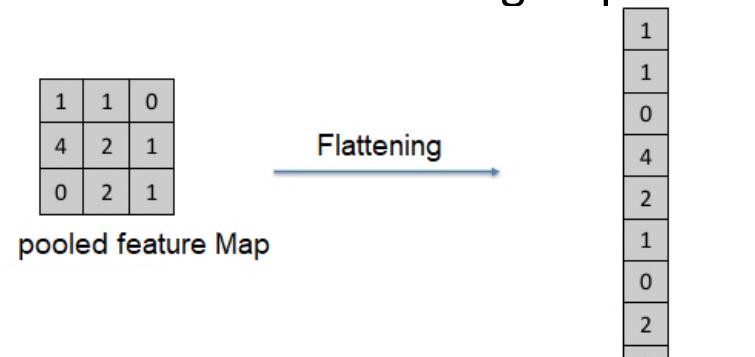
Flattening (1/2)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Cette étape est simple.

Après avoir terminé les deux étapes précédentes, nous sommes censés avoir une carte des fonctionnalités regroupées maintenant. Aplatir notre carte d'entités regroupées dans une colonne comme dans l'image.

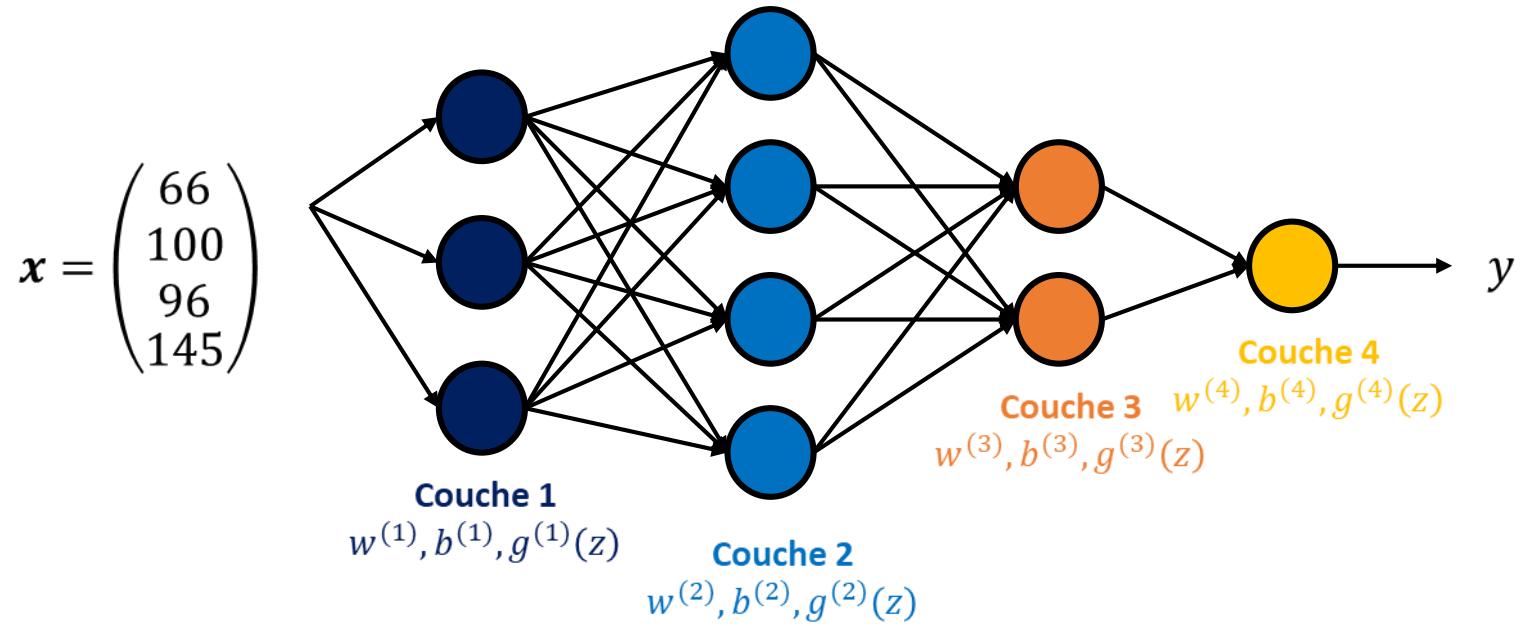
La raison pour laquelle nous faisons cela est que nous allons devoir insérer ces données dans un réseau neuronal artificiel plus tard.



Flattening (2/2)

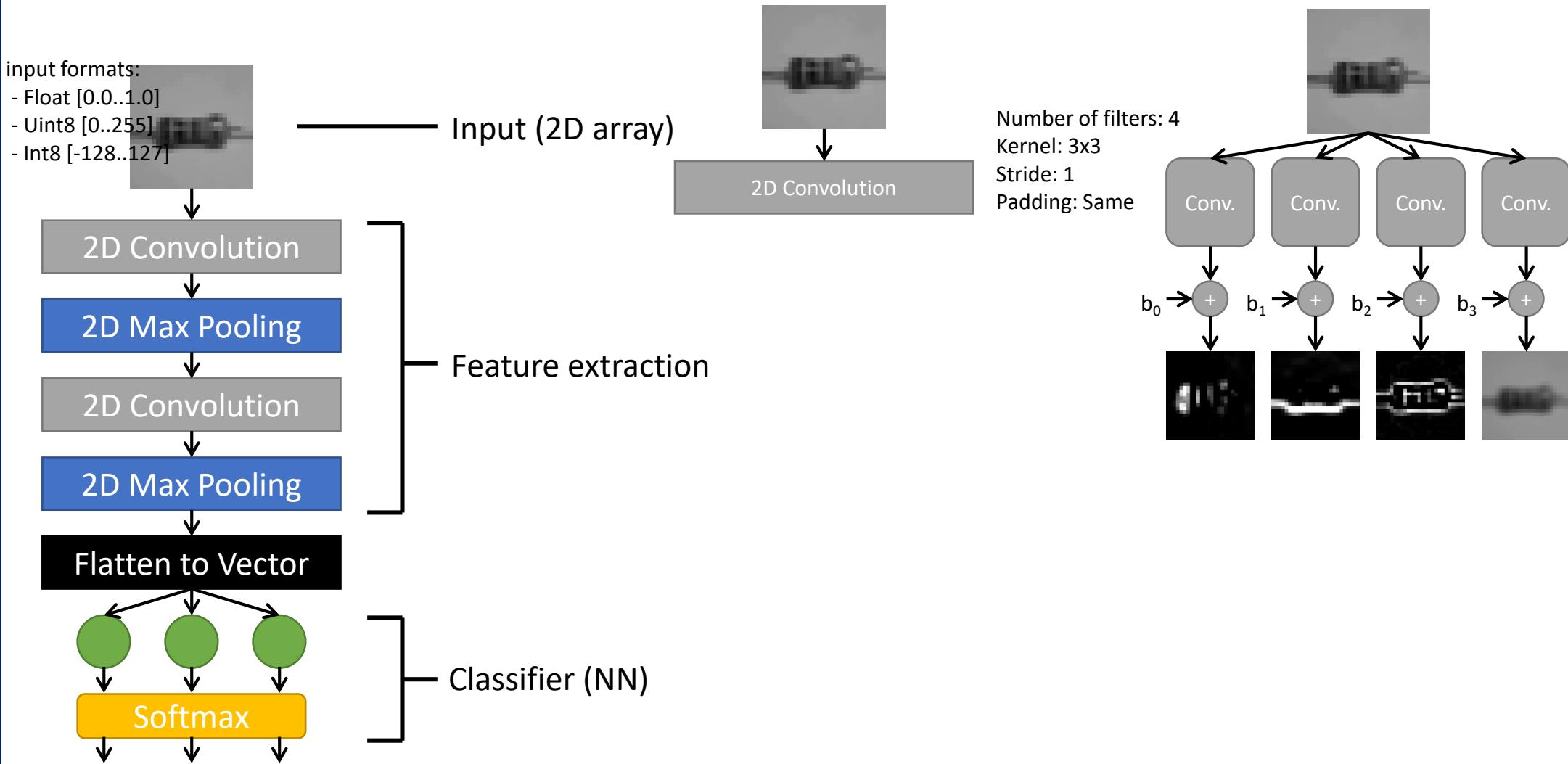
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

66	100
96	145



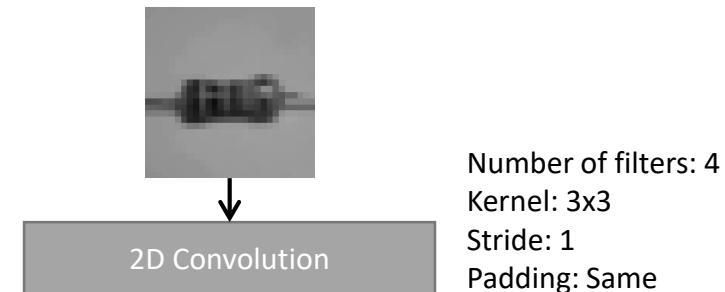
Architecture des réseaux profonds(1/6)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

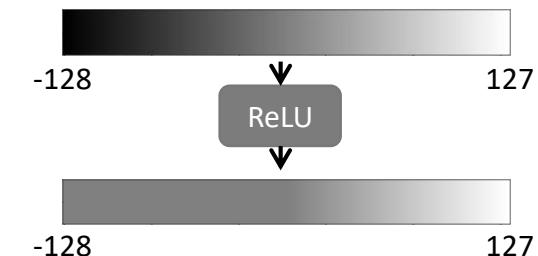
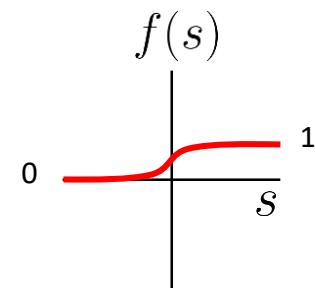


Architecture des réseaux profonds(2/6)

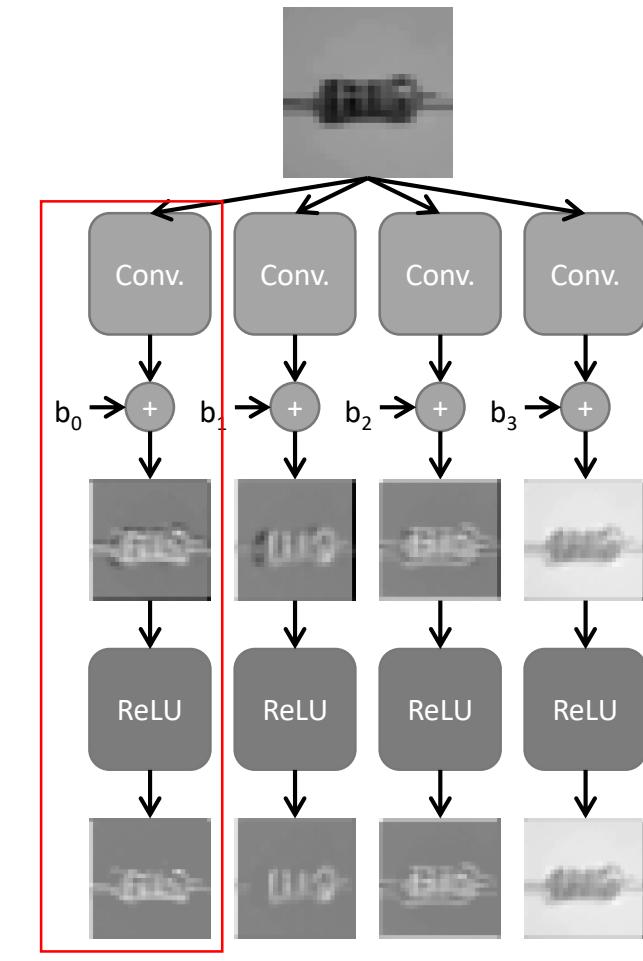
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



Sigmoid function



ReLU function



Architecture des réseaux profonds(3/6)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds

Choix de la fonction d'activation

✓ Le **ReLU** est l'activation par défaut lors du développement de Perceptron multicouche et de réseaux de neurones convolutifs.

Limitations sigmoid et tanh :

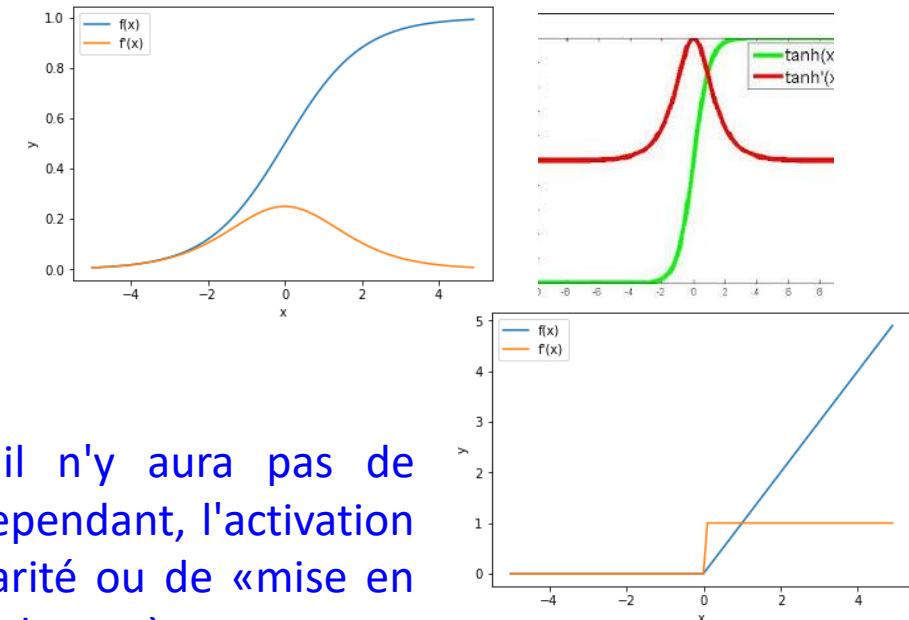
produisent un mauvais apprentissage pour les réseaux profonds à cause d'un problème connu sous le nom Vanishing Gradient Problem ou erreur de chute.

Aussi; demande un calcul coûteux.

Avantages

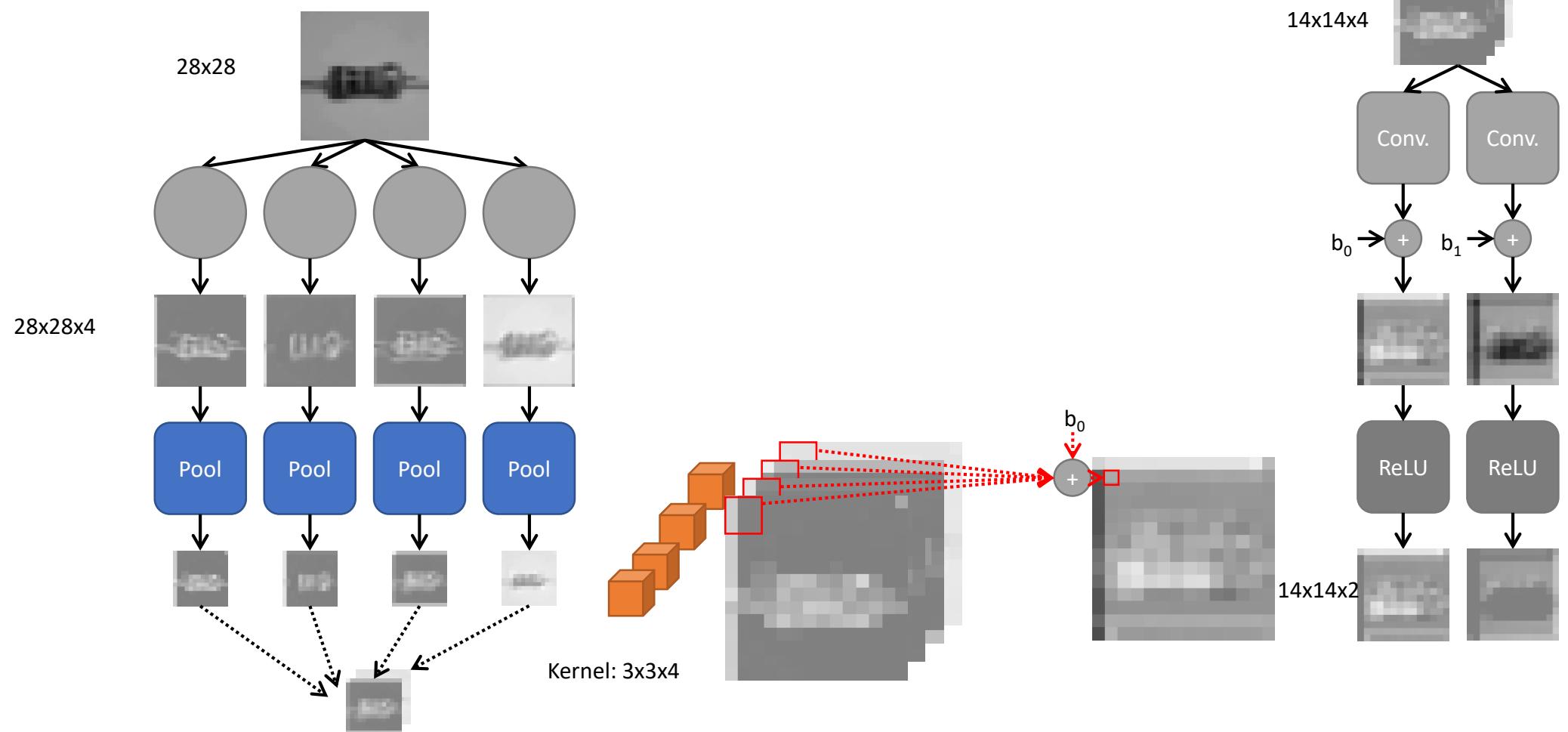
- ✓ N'est couteux en temps de calcul.
- ✓ Presque linéaire.
- ✓ Assure la non-linéarité dans les réseaux profonds.

✓ La dérivée de ReLU (orange) est rétro-propagée, il n'y aura pas de dégradation du signal d'erreur, car $1 \times 1 \times 1 \times 1 \dots = 1$. Cependant, l'activation ReLU conserve toujours une caractéristique de non-linéarité ou de «mise en marche» qui lui permet de se comporter de manière analogue à un neurone biologique.



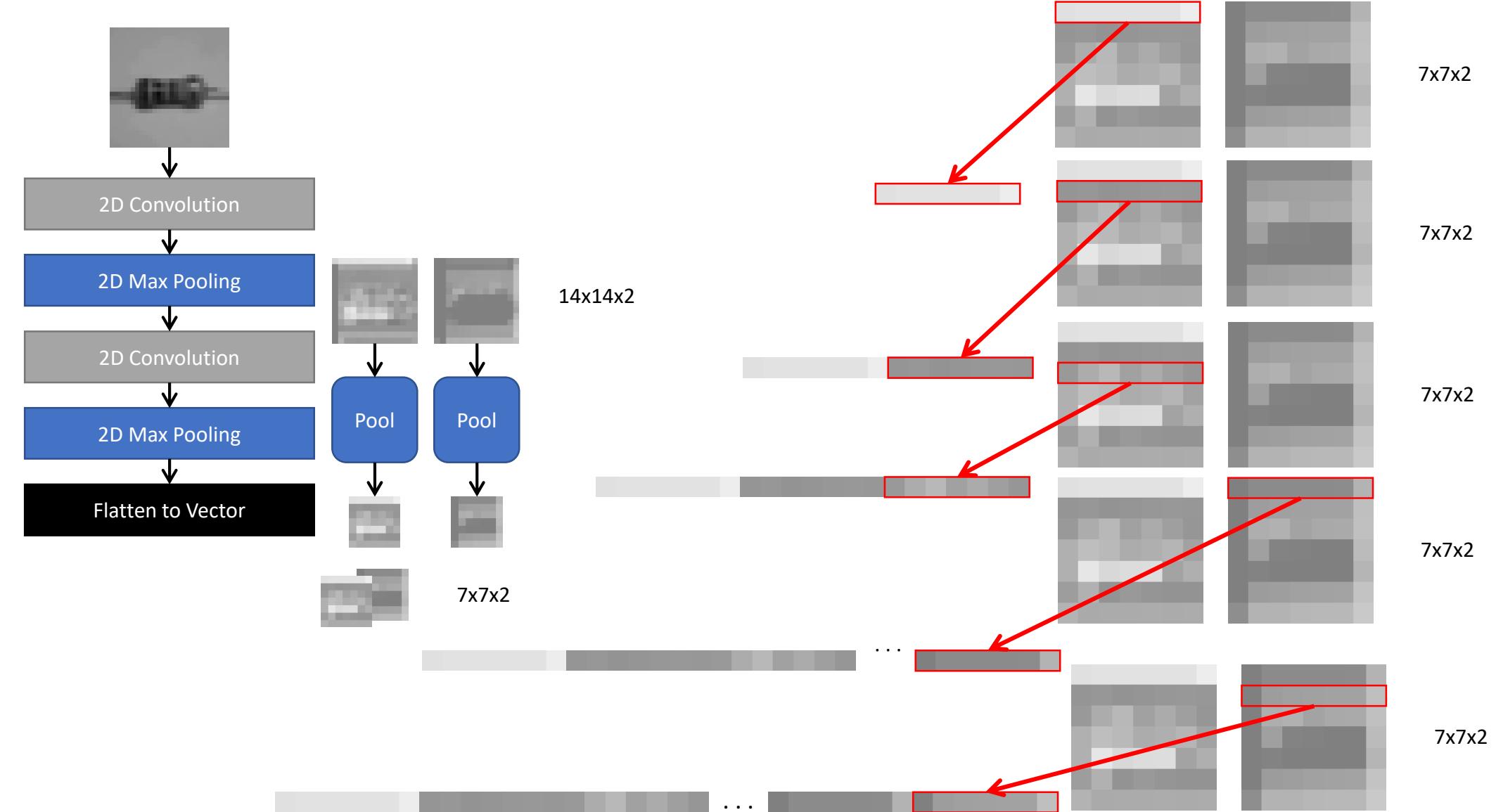
Architecture des réseaux profonds(4/6)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



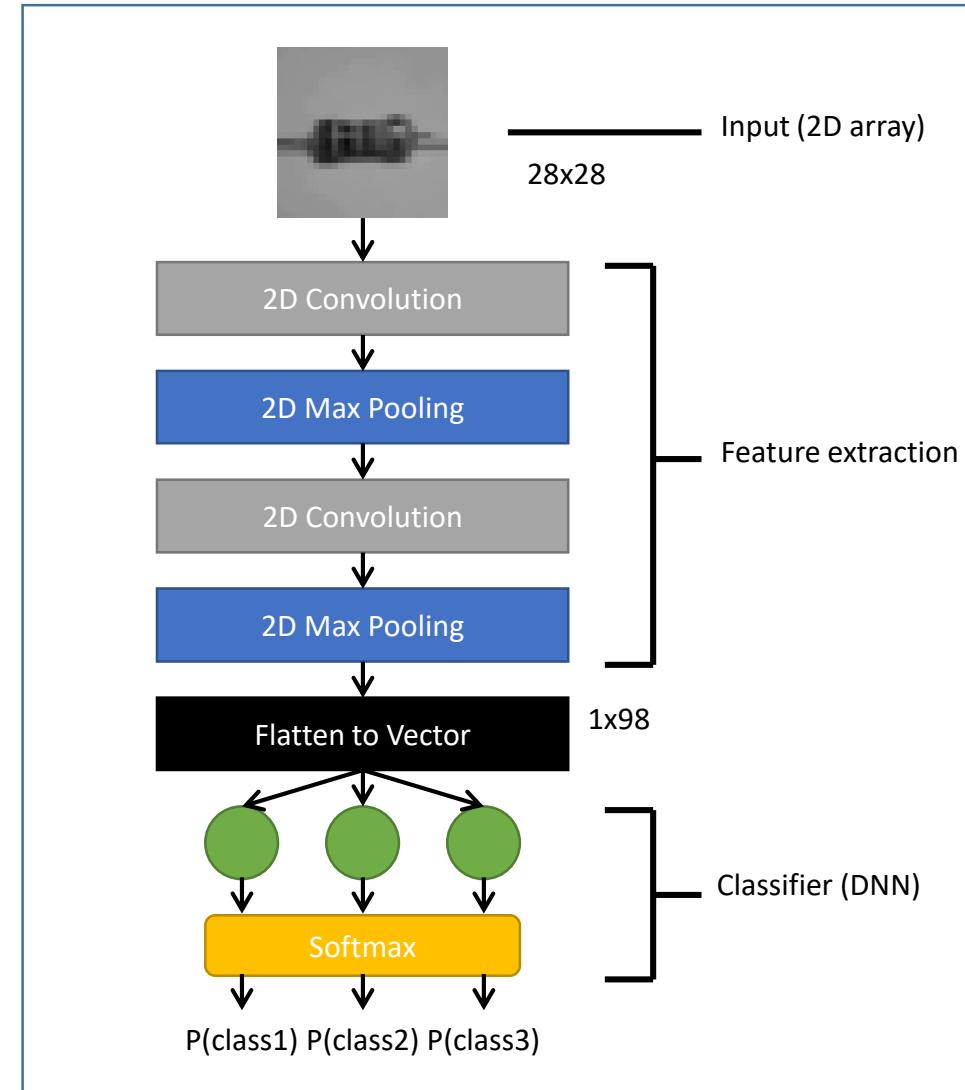
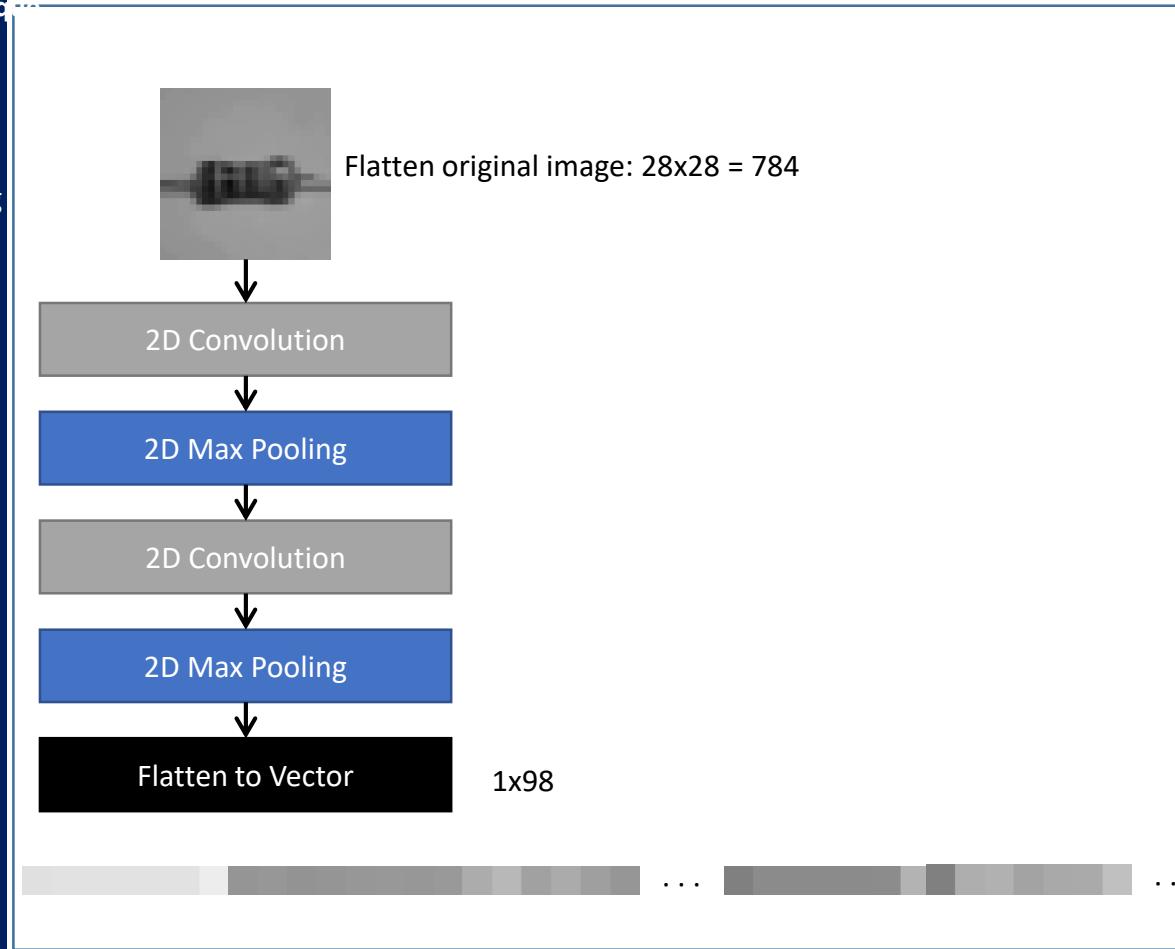
Architecture des réseaux profonds(5/6)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



Architecture des réseaux profonds(6/6)

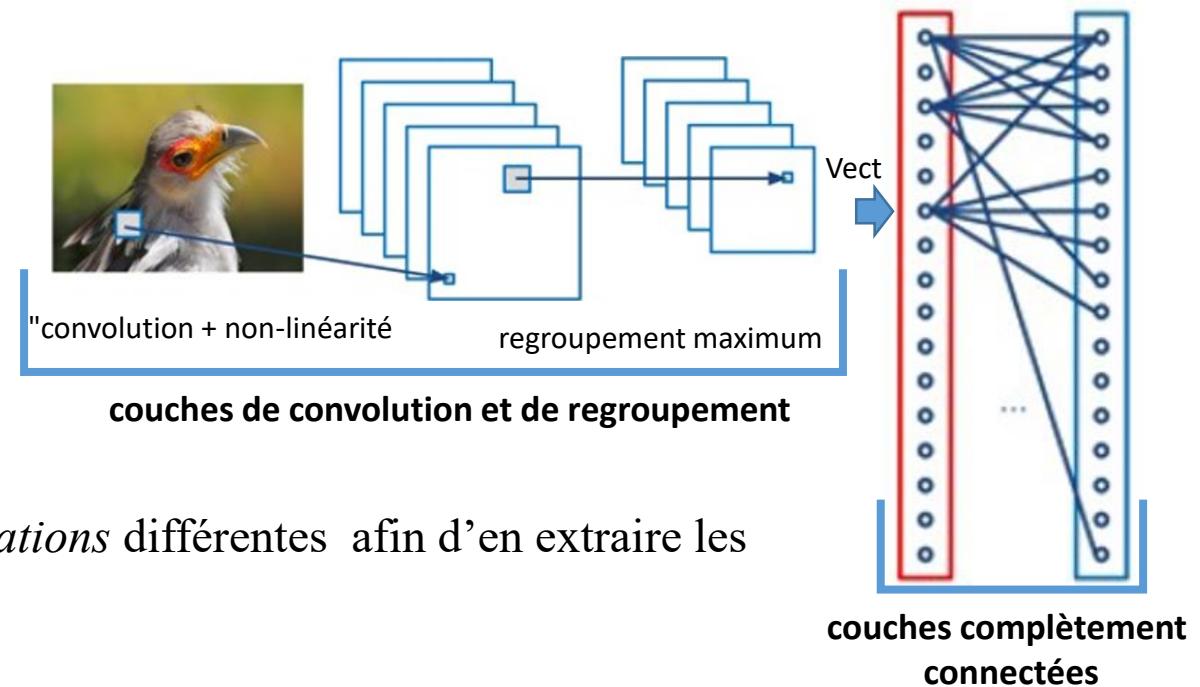
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



Système de reconnaissance de motifs(1/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Les CNN sont conçus pour apprendre des caractéristiques de bas niveau à partir des données en 2D en entrée, en utilisant des couches de convolution et de regroupement pour extraire des motifs et des textures.



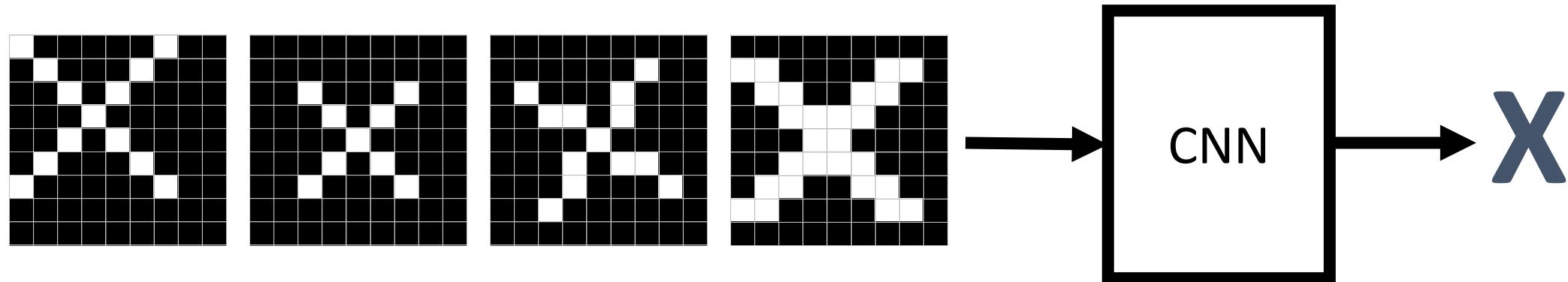
Un CNN applique généralement 3 types d'opérations différentes afin d'en extraire les informations pertinentes.

Ces 3 types d'opérations sont les suivantes :

- La *convolution*
- Le regroupement (*pooling*)
- La fonction d'activation de type ReLU

Système de reconnaissance de motifs(2/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



translation

$$\begin{matrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & \boxed{1} & -1 & -1 & -1 & -1 & \boxed{1} & -1 \\ -1 & -1 & \boxed{1} & -1 & -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & \boxed{1} & -1 & \boxed{1} & -1 & -1 \\ -1 & -1 & -1 & -1 & \boxed{1} & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 \\ -1 & -1 & -1 & 1 & -1 & \boxed{1} & -1 & -1 \\ -1 & -1 & \boxed{1} & -1 & -1 & -1 & \boxed{1} & -1 \\ -1 & \boxed{1} & -1 & -1 & -1 & -1 & \boxed{1} & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & \boxed{1} \end{matrix}$$

scaling

$$\begin{matrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & \boxed{1} & -1 & -1 & -1 & -1 & \boxed{1} & -1 \\ -1 & -1 & \boxed{1} & -1 & -1 & -1 & 1 & -1 \\ -1 & \boxed{1} & -1 & -1 & -1 & \boxed{1} & -1 & -1 \\ -1 & -1 & \boxed{1} & 1 & -1 & 1 & -1 & -1 \\ -1 & -1 & -1 & -1 & \boxed{1} & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & \boxed{1} & 1 & -1 \\ -1 & -1 & -1 & -1 & -1 & 1 & -1 & \boxed{1} \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{matrix}$$

?

rotation

$$\begin{matrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{matrix}$$

weight

$$\begin{matrix} = & = & = & = & = & = & = & = \\ = & \textcolor{red}{\#} & = & = & = & = & \textcolor{red}{\#} & \textcolor{red}{\#} \\ = & \textcolor{red}{\#} & \textcolor{red}{\#} & = & = & = & \textcolor{red}{\#} & \textcolor{red}{\#} \\ = & = & \textcolor{red}{\#} & = & = & = & = & = \\ = & = & = & = & = & = & = & = \\ = & = & = & = & = & = & \textcolor{red}{\#} & = \\ = & = & = & = & = & = & = & \textcolor{red}{\#} \\ = & \textcolor{red}{\#} & \textcolor{red}{\#} & = & = & = & = & \textcolor{red}{\#} \\ = & = & = & = & = & = & = & = \end{matrix}$$

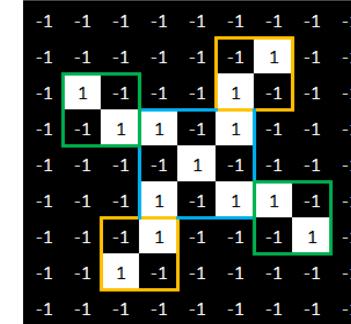
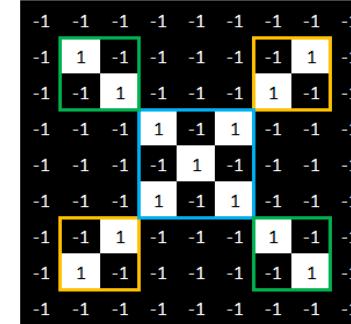
En comparant *pixel par pixel*, nous voyons qu'il y a des pixels différents entre les 2 images.

Pourtant, certains *morceaux de l'image* sont identiques entre les 2 images.

Système de reconnaissance de motifs(3/12)

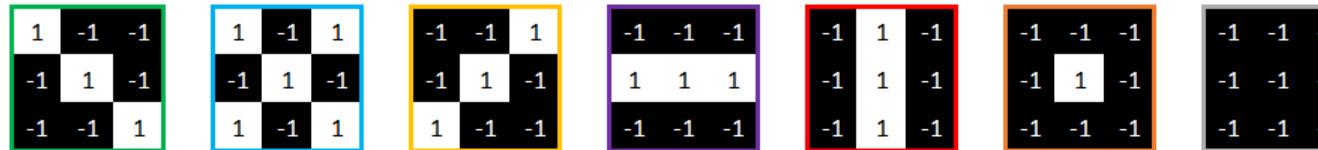
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Ces *morceaux* sont au nombre de 3.

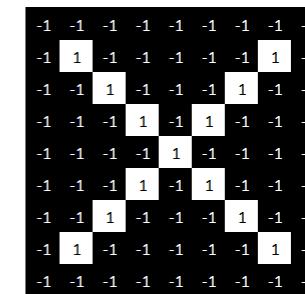


Nous appellerons désormais ces *morceaux* des **features**.

Considérons désormais des features de taille 3×3 , nous allons voir comment la machine peut les trouver au moyen de calculs très simples.
Voici donc les nouvelles features que nous souhaitons que la machine trouve.



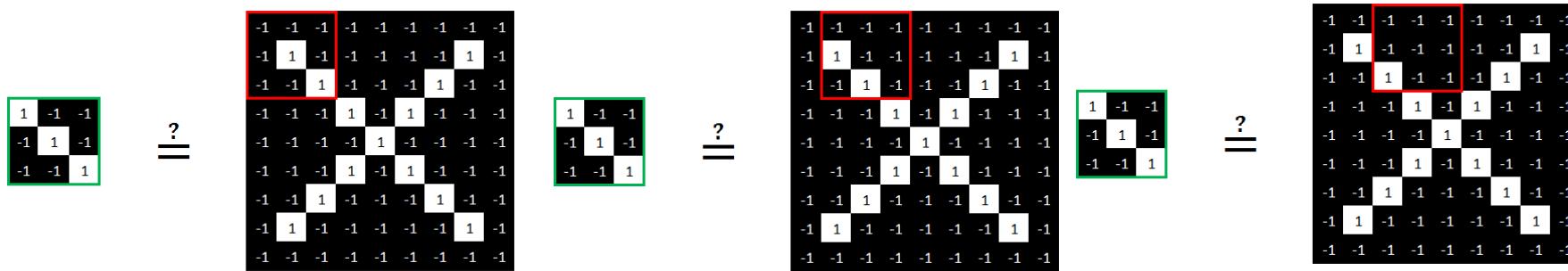
Nous allons commencer par rechercher cette feature sur cette image.



Pour cela, nous allons comparer cette feature en la faisant « glisser » sur l'image comme suit :

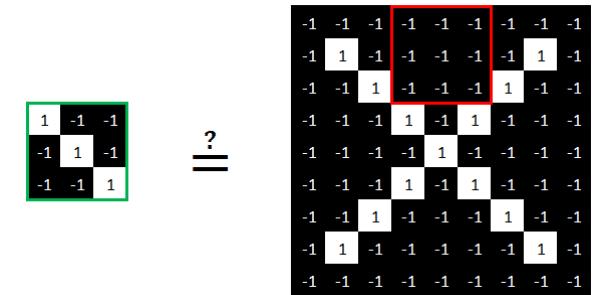
Système de reconnaissance de motifs(4/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



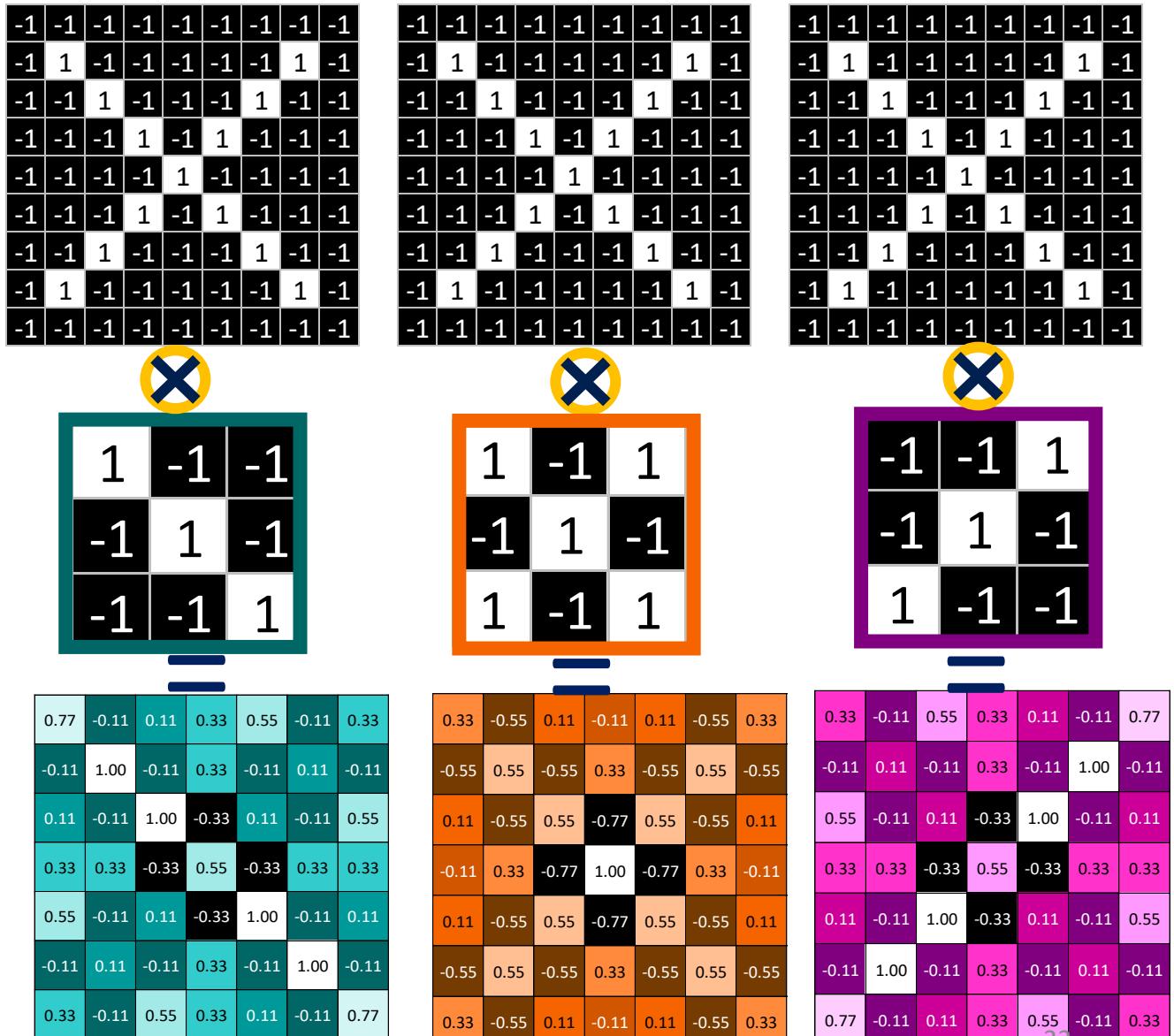
Concrètement, pour comparer cette feature avec une partie de l'image, nous allons :

- 1/ multiplier les 9 valeurs des pixels de la caractéristique avec les 9 valeurs des pixels du morceau de l'image à trouver,
- 2/ additionner ces 9 résultats
- 3/ diviser par le nombre de pixels (ici 9)
- 4/ conclure (si le résultat est égal à 1, alors la feature a été identifiée dans l'image)



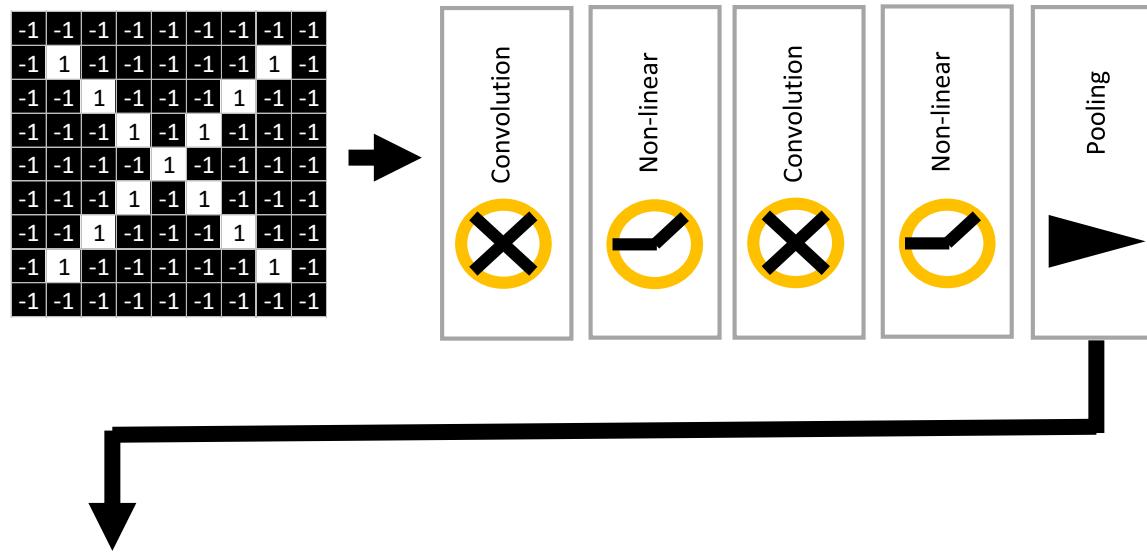
Système de reconnaissance de motifs(5/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
Convolution 2D
V Padding 2D
Conv Avec t sans Padding
Autres Réseaux
Pax Pooling
Flatinning
Archi des R Profonds
Syst de recon de motifs
Exemple de RN profonds
ANN Vs CNN



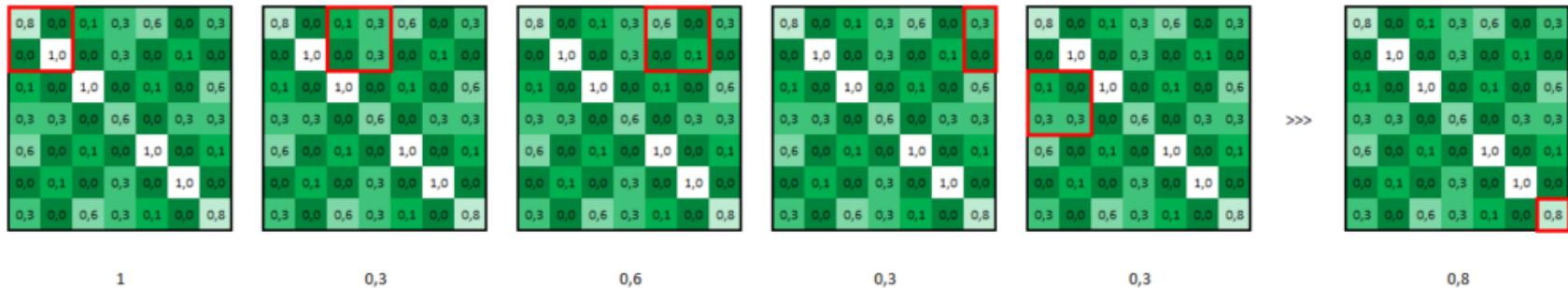
Système de reconnaissance de motifs(6/12)

- I. ML vs DL
 - II. Limites REC
 - III. Une approche biologique
 - IV. Conv 2D
 - et filtrage
 - Convolution 2D
 - V Padding 2D
 - Conv Avec t sans Padding
 - Autres Réseaux
 - Pax Pooling
 - Flatining
 - Archi des R Profonds
 - Syst de recon de motifs
 - Exemple de RN profonds
 - ANN Vs CNN



Le pooling

Le max-pooling prend la valeur maximale de chaque « feature de l'image ».

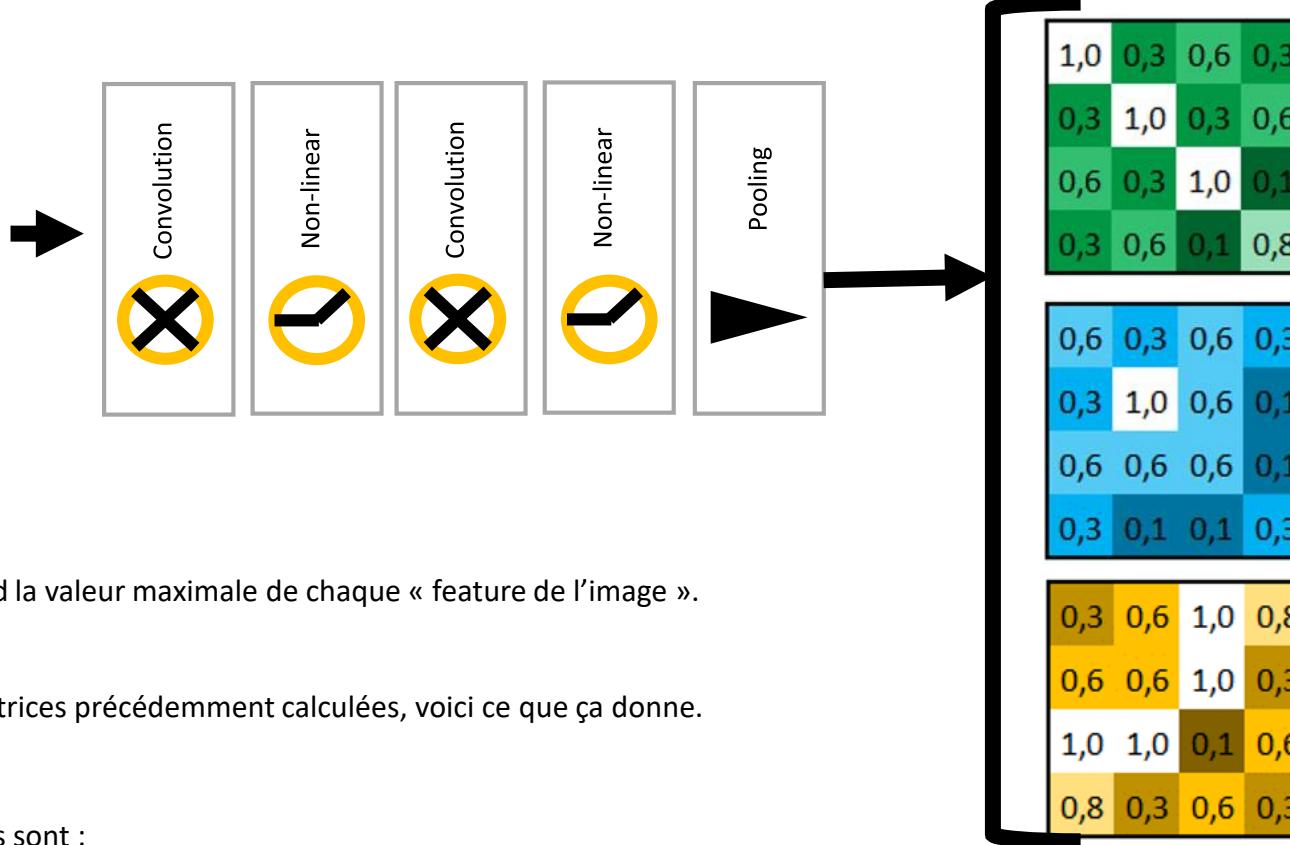


Appliquée à nos matrices précédemment calculées, voici ce que ça donne.

Système de reconnaissance de motifs(7/12)

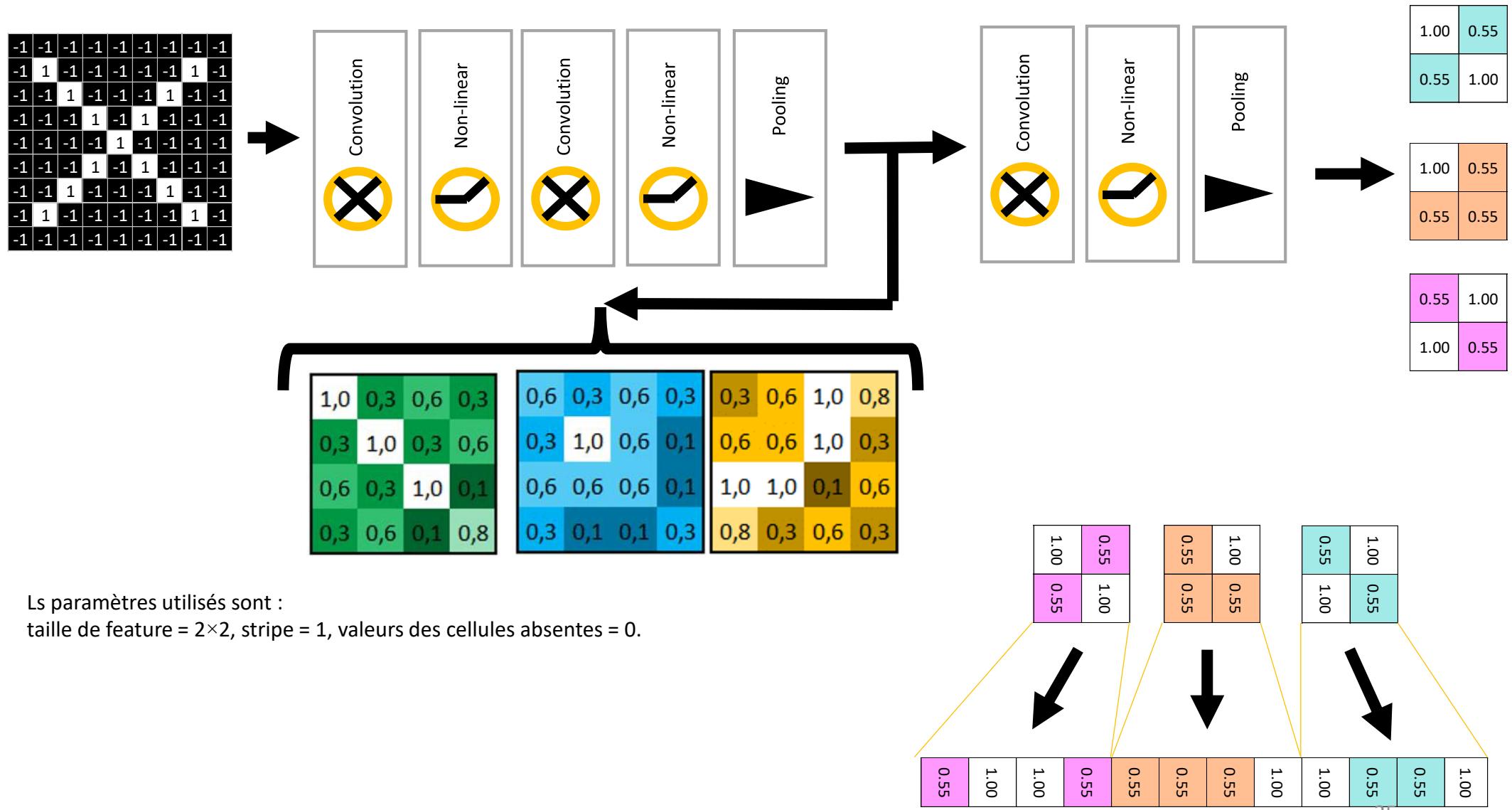
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	1	-1	-1	-1	-1	-1	1	-1
-1	-1	1	-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	1	-1	-1	-1
-1	-1	-1	-1	1	-1	-1	-1	-1
-1	-1	-1	1	-1	1	-1	-1	-1
-1	-1	1	-1	-1	-1	1	-1	-1
-1	1	-1	-1	-1	-1	1	-1	-1
-1	-1	-1	-1	-1	-1	-1	1	-1



Système de reconnaissance de motifs(8/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



Système de reconnaissance de motifs(9/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Apprentissage d'un CNN

```
#include<Arduino.h>
#define kernelSize 3
#define poolSize 2
const int NN = 6;
const int MM = 6;
void convolve2D(int input[NN][MM], int
output[NN-2][MM-2], int
kernel[kernelSize][kernelSize]) {
    for (int i = 0; i < NN-2; i++) {
        for (int j = 0; j < MM-2; j++) {
            int sum = 0;
            for (int k = 0; k < kernelSize; k++) {
                for (int l = 0; l < kernelSize; l++) {
                    sum += input[i + k][j + l] *
kernel[k][l];
                }
            }
            output[i][j] = sum;
        }
    }
}
```

Système de reconnaissance de motifs(10/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Apprentissage d'un CNN

```
#include<Arduino.h>
#define kernelSize 3
#define poolSize 2
const int NN = 6;
const int MM = 6;
void maxPool(int input[NN-2][MM-2], int output[NN-2-2][MM-2-2]) {
    for (int i = 0; i < NN-4; i++) {
        for (int j = 0; j < MM-4; j++) {
            int maxValue = 0;
            for (int k = 0; k < poolSize; k++) {
                for (int l = 0; l < poolSize; l++) {
                    maxValue = max(maxValue, input[i * poolSize + k][j * poolSize + l]);
                }
            }
            output[i][j] = maxValue;
        }
    }
}
```

```
void flatten() {
    int idx = 0;
    for(int i = 0; i < OUTPUT_SIZE/POOL_SIZE; i++)
    {
        for(int j = 0; j < OUTPUT_SIZE/POOL_SIZE; j++)
        {
            flattened[idx++] = output[i][j];
        }
    }
}
```

Système de reconnaissance de motifs(11/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Apprentissage d'un CNN

```
void setup() {  
    Serial.begin(115200);  
    int input[6][6] = {{1, 2, 3, 2, 1, 0},{0, 1, 2, 3, 2, 1},  
                      {1, 2, 3, 2, 1, 0},{0, 1, 2, 3, 2, 1},  
                      {1, 2, 3, 2, 1, 0},{0, 1, 2, 3, 2, 1}};  
    int kernel[kernelSize][kernelSize] = {{0, 1, 0},{0, 1, 0},{-1, 0, -1}};  
    int convOutput[4][4];  
    convolve2D(input, convOutput, kernel);  
    int poolOutput[2][2];  
    maxPool(convOutput, poolOutput);  
    for(int i=0;i<2;i++)  
    {  
        for(int j=0;j<2;j++)  
        {  
            Serial.print("poolOutput:[\t"); Serial.print(i); Serial.print(",\t");  
            Serial.print(j);Serial.print("]:= \t");Serial.println(poolOutput[i][j],3);  
        }  
    }  
}  
void loop() {  
    // inserer votre code  
}
```

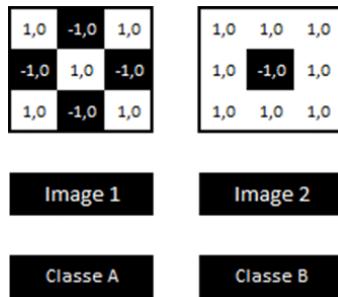
Système de reconnaissance de motifs(12/12)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Apprentissage d'un CNN

Notre dataset

Nous pouvons utiliser 2 images ou plus (respectivement une croix et un rond), mettre en place un CNN, puis l'entraîner afin qu'il minimise la fonction coût.



Variables à ajuster

Dans un CNN, 3 types de variables seront ajustés à chaque itération afin de minimiser notre fonction coût, à savoir :

- Les poids des couches du réseau de neurones fully connected
- Les biais des couches du réseau de neurones fully connected
- Les valeurs des pixels de notre filtre.

-0,5	-0,9	0,4
-0,8	0,7	0,2
0,8	0,3	0,4

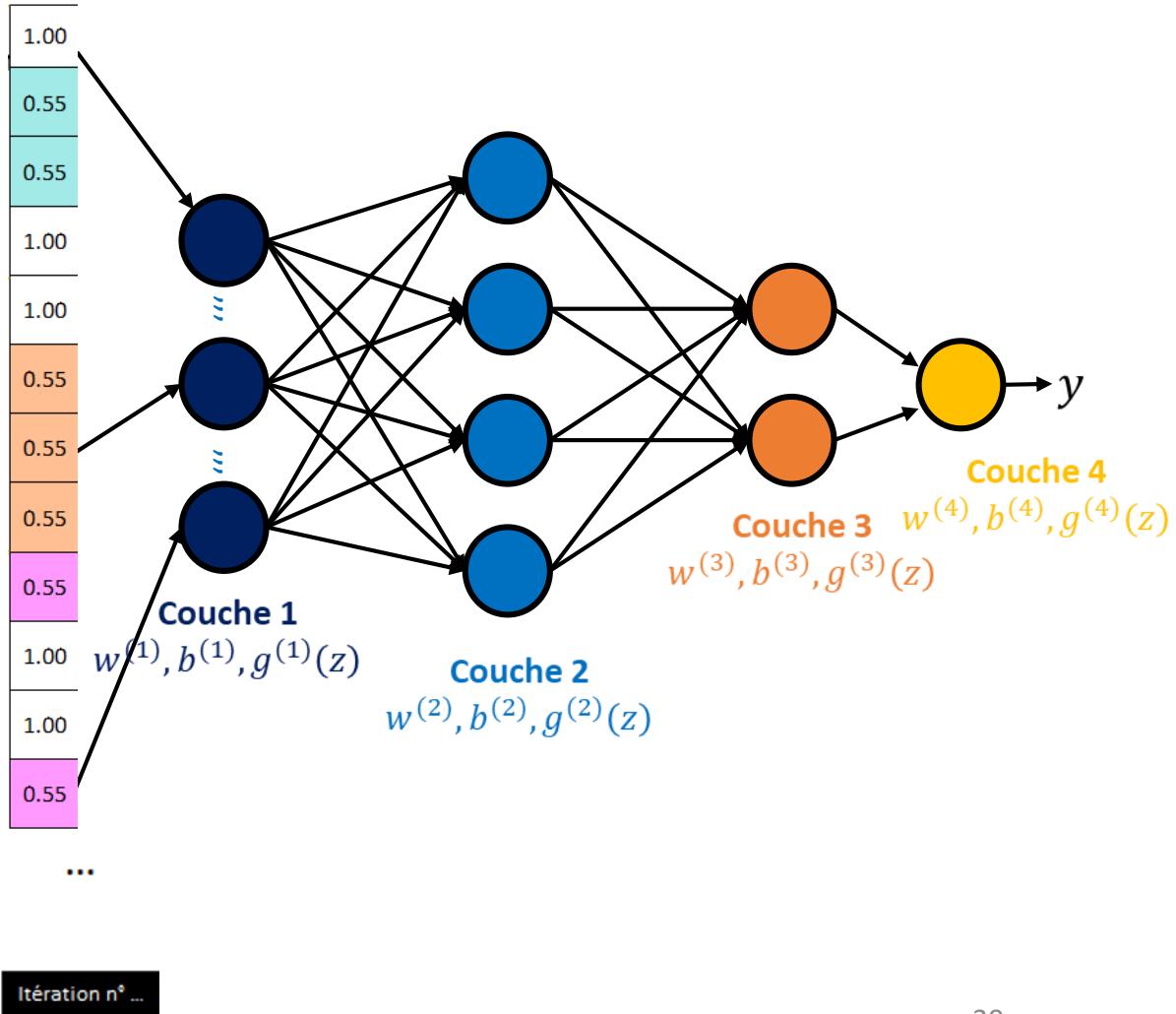
Itération n° 1

-0,8	-0,9	0,4
-0,1	0,6	0,5
0,7	0,4	0,2

Itération n° 2

-0,9	-0,7	0,3
-0,5	0,8	0,4
0,7	0,3	0,2

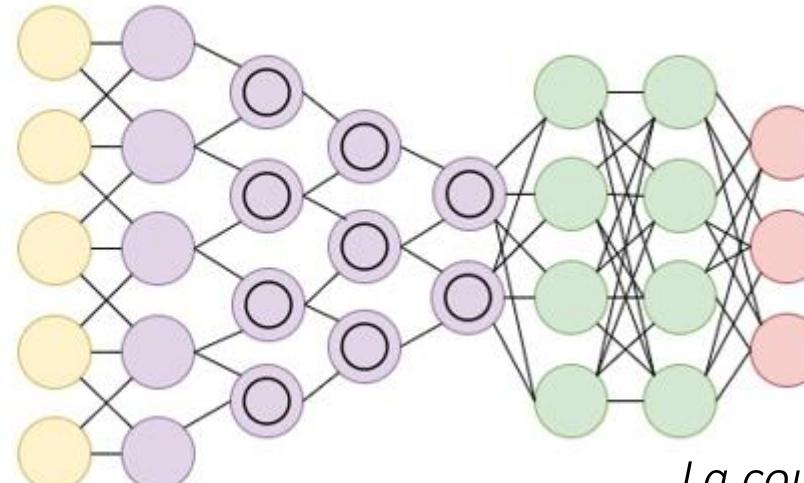
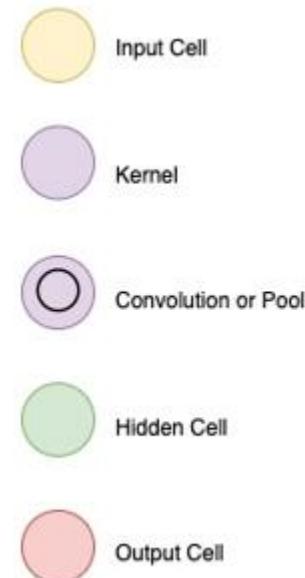
Itération n° 3



Le softmax(1/2)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

La ■ Z_i sont les éléments du vecteur d'entrée et peuvent prendre n'importe quelle valeur réelle
■ K est le nombre de classes dans le classifieur multi-classes.



$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

Dans ce cas K=2

Le softmax(2/2)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatining
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Fonction Softmax:

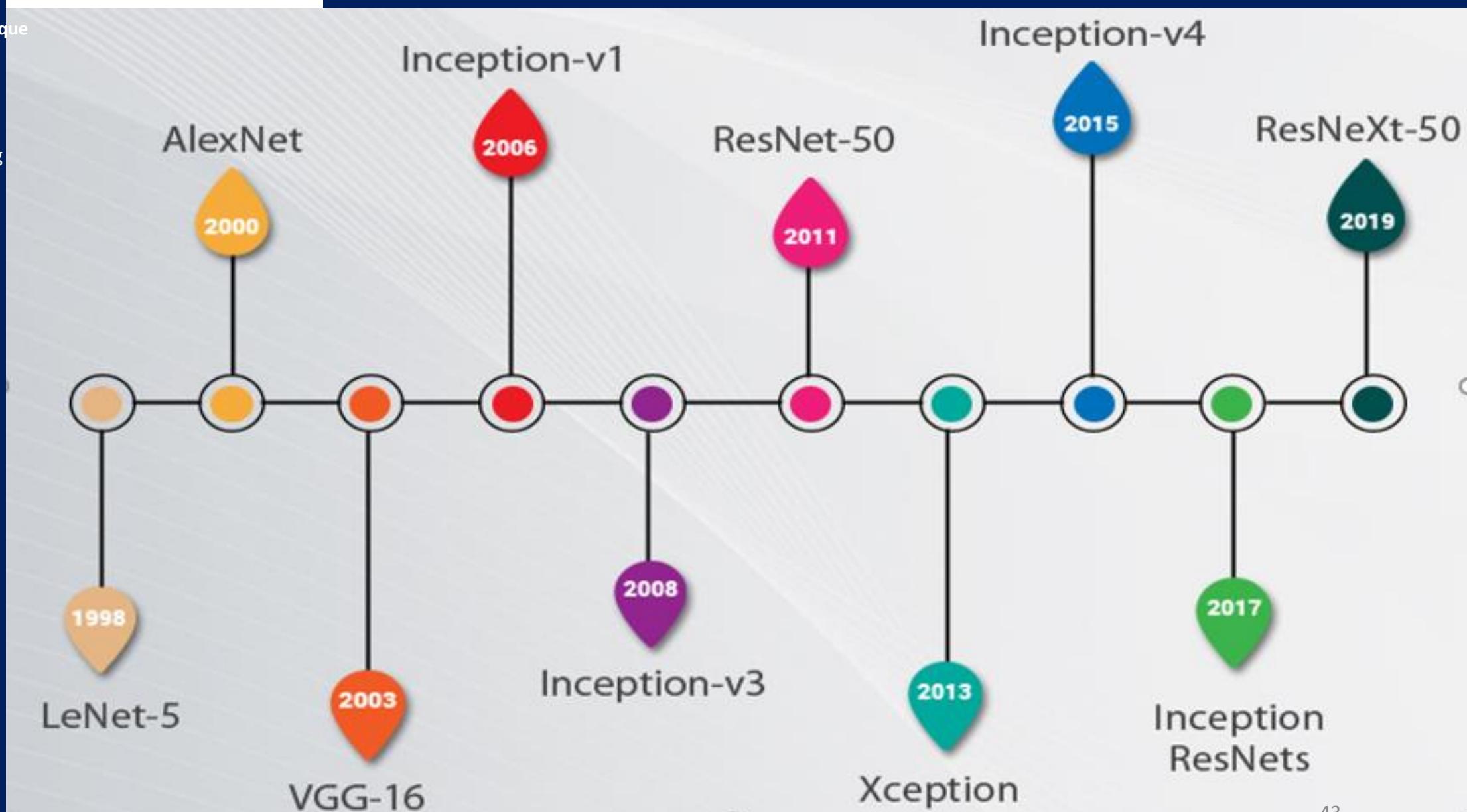
```
#include<Arduino.h>
#include<math.h>
// Define the softmax function
void softmax(float *x, int len) {
    float max_val = -INFINITY;
    for (int i = 0; i < len; i++) {
        max_val = max(max_val, x[i]);
    }
    float exp_sum = 0;
    for (int i = 0; i < len; i++) {
        x[i] = exp(x[i] - max_val);
        exp_sum += x[i];
    }
    for (int i = 0; i < len; i++) {
        x[i] /= exp_sum;
    }
}
///////////////////////////////
void setup() {
    float x[] = {1.0, 2.0, 3.0};
    int len = 3;
    softmax(x, len);
}

void loop() {
    // }
```

3 Exemple des réseaux de neurones profonds

Exemple des réseaux de neurones profonds(1/7)

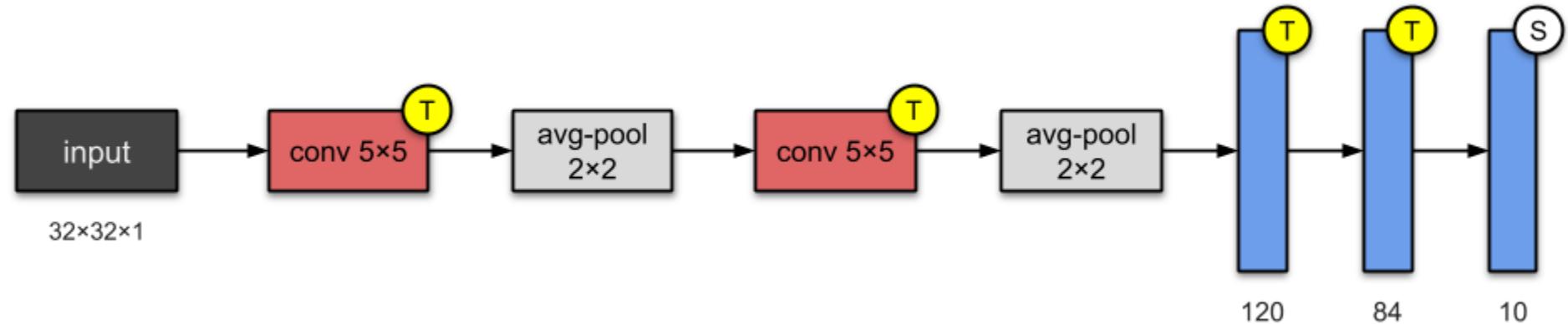
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN



Exemple des réseaux de neurones profonds(2/7)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

LeNet-5



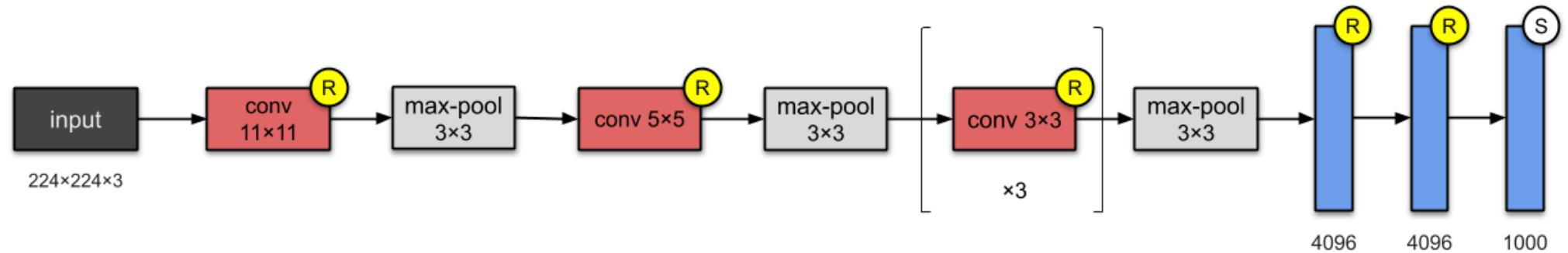
L'architecture est devenue le "modèle" standard :

- (1) empilage de convolutions et mise en commun de couches (7 niveaux)
- (2) terminer le réseau par une ou plusieurs couches entièrement connectées

Exemple des réseaux de neurones profonds(3/7)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

AlexNet

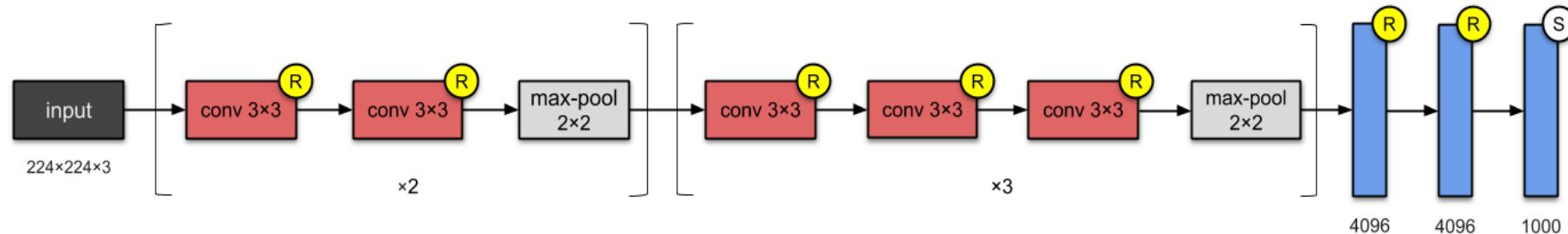


- (1) plus profond et avec plus de filtres que LeNet
 - (2) a utilisé des couches convolutives empilées
 - (3) a été le premier à mettre en œuvre des unités linéaires rectifiées (ReLUs) comme fonctions d'activation
- en 2012 a surpassé tous les concurrents précédents et a remporté le défi ImageNet

Exemple des réseaux de neurones profonds(4/7)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

VGG-16

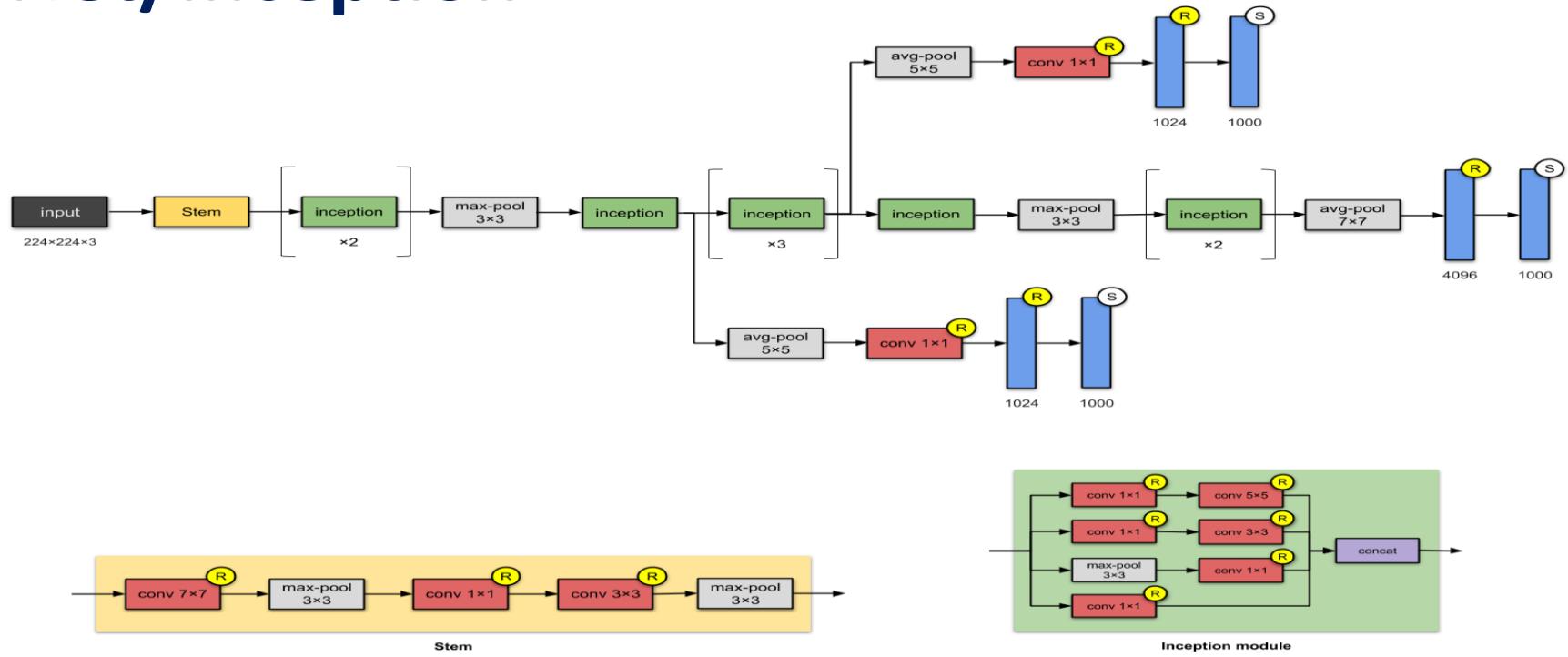


- (1) réseau beaucoup plus profond que les précédents (16 à 19 couches)
 - (2) des filtres plus nombreux et plus petits (3x3)
- est arrivé deuxième au concours ImageNet 2014

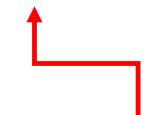
Exemple des réseaux de neurones profonds(5/7)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

GoogleNet/Inception



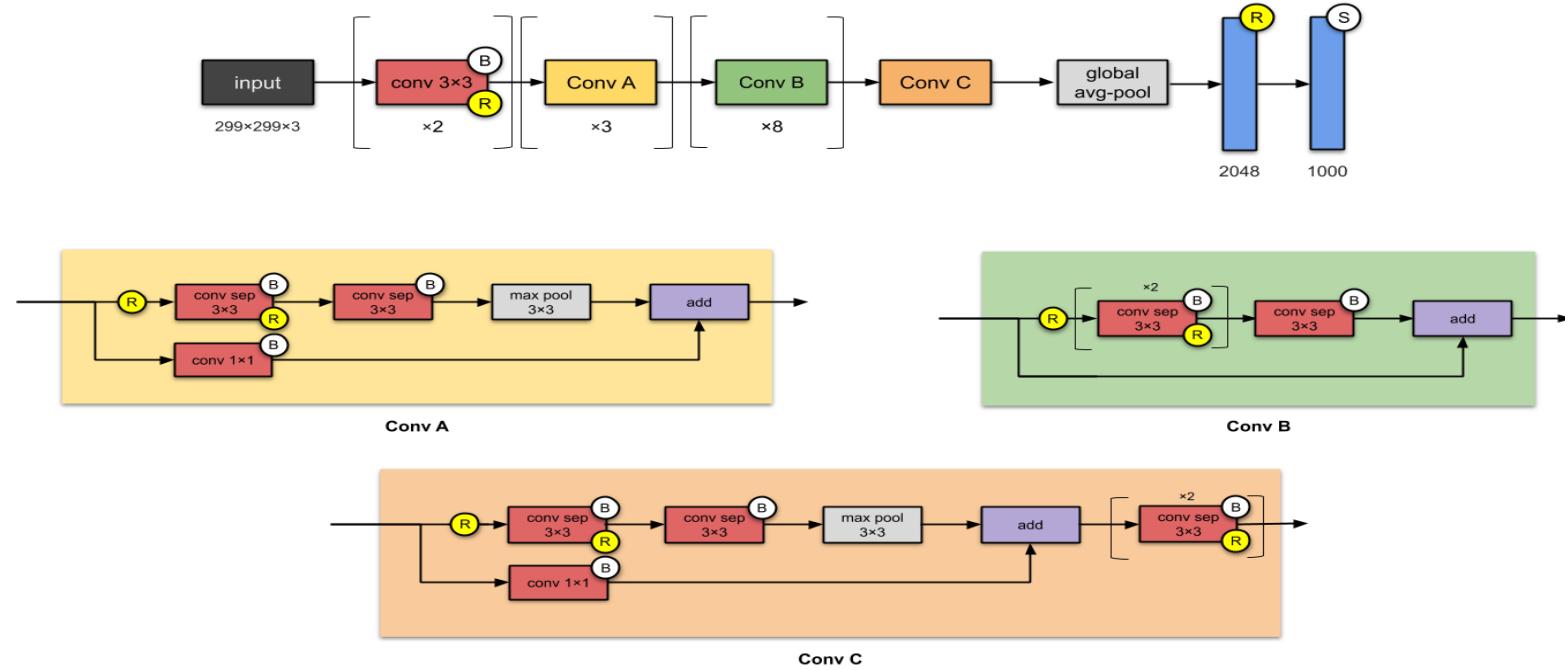
- (1) réseau encore plus profond (22 couches)
 - (2) mise en œuvre du module de démarrage pour réduire considérablement le nombre de paramètres
 - (3) normalisation par lots
- a remporté le défi ImageNet en 2014



Exemple des réseaux de neurones profonds(6/7)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Xception (2016)

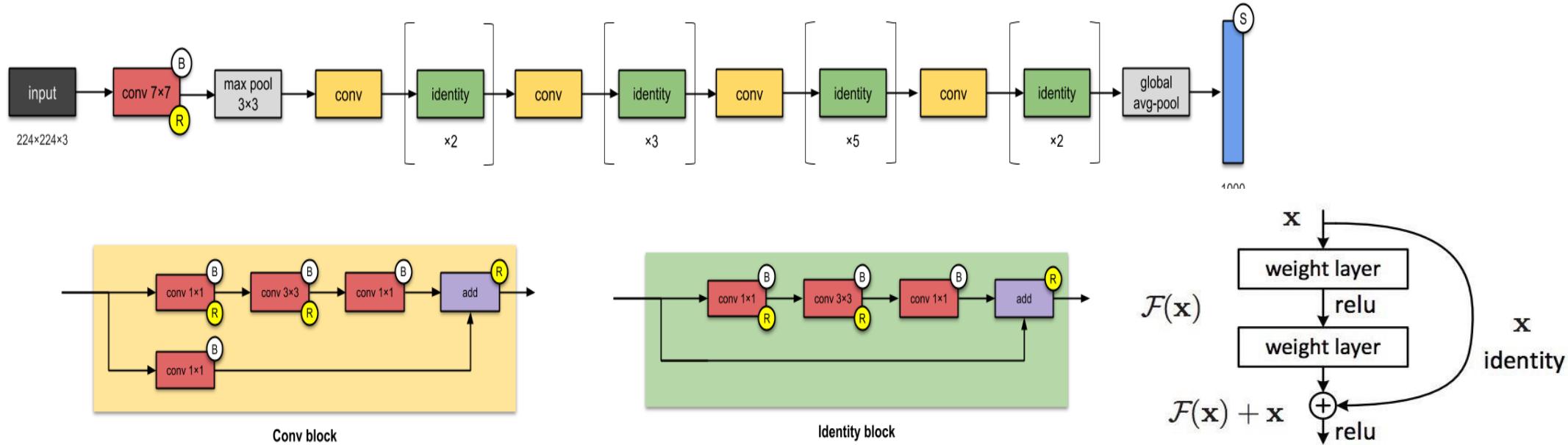


- (1) les modules d'initiation sont remplacés par des convolutions séparables en profondeur
 - (2) même nombre de paramètres qu'à l'origine
- surpassé légèrement Inception v3 sur l'ensemble de données ImageNet, et largement sur un ensemble de données de classification d'images plus vaste comprenant 17 000 classes.

Exemple des réseaux de neurones profonds(7/7)

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Resnet (2015)



- (1) au lieu d'apprendre une correspondance à partir dex to $H(x)$, apprendre la différence $F(x) = H(x) - x$
→ crée des sauts de connexion
- (2) connexions de saut empilées l'une sur l'autre
→ permet des architectures plus profondes sans perte de performance

ANN vs CNN

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

ANN	CNN
<ul style="list-style-type: none">• Utilise des couches entièrement connectées. ANN ne peut être utilisé que pour les petites images.	<ul style="list-style-type: none">• Utilise des couches partiellement connectées. La taille de la connexion dépend de la taille du filtre. Le CNN peut être utilisé pour n'importe quelle image.
<ul style="list-style-type: none">• Nombre de paramètres très élevé, Pour 28 * 28 image et la couche a 32 neurones; le nombre de paramètres pour une couche sera $28 * 29 * 32 = 25088$. Par conséquent, le temps de calcul dans l'ANN est très couteux.	<ul style="list-style-type: none">• Le nombre de paramètres dans le CNN sera très inférieur par rapport à ANN. Pour une image 28 * 28 et si le calque a 32 filtres de taille (5 * 5), le nombre de paramètres pour ce calque sera $(5 * 5 * 1 * 1) * 32 = 832$. Par conséquent, le temps de calcul du CNN est nettement inférieur à celui de l'ANN.
<ul style="list-style-type: none">• Pas très efficace pour les images malgré leur coût.	<ul style="list-style-type: none">• Très efficace pour les images et moins couteux comparé à l'ANN.

ANN vs CNN

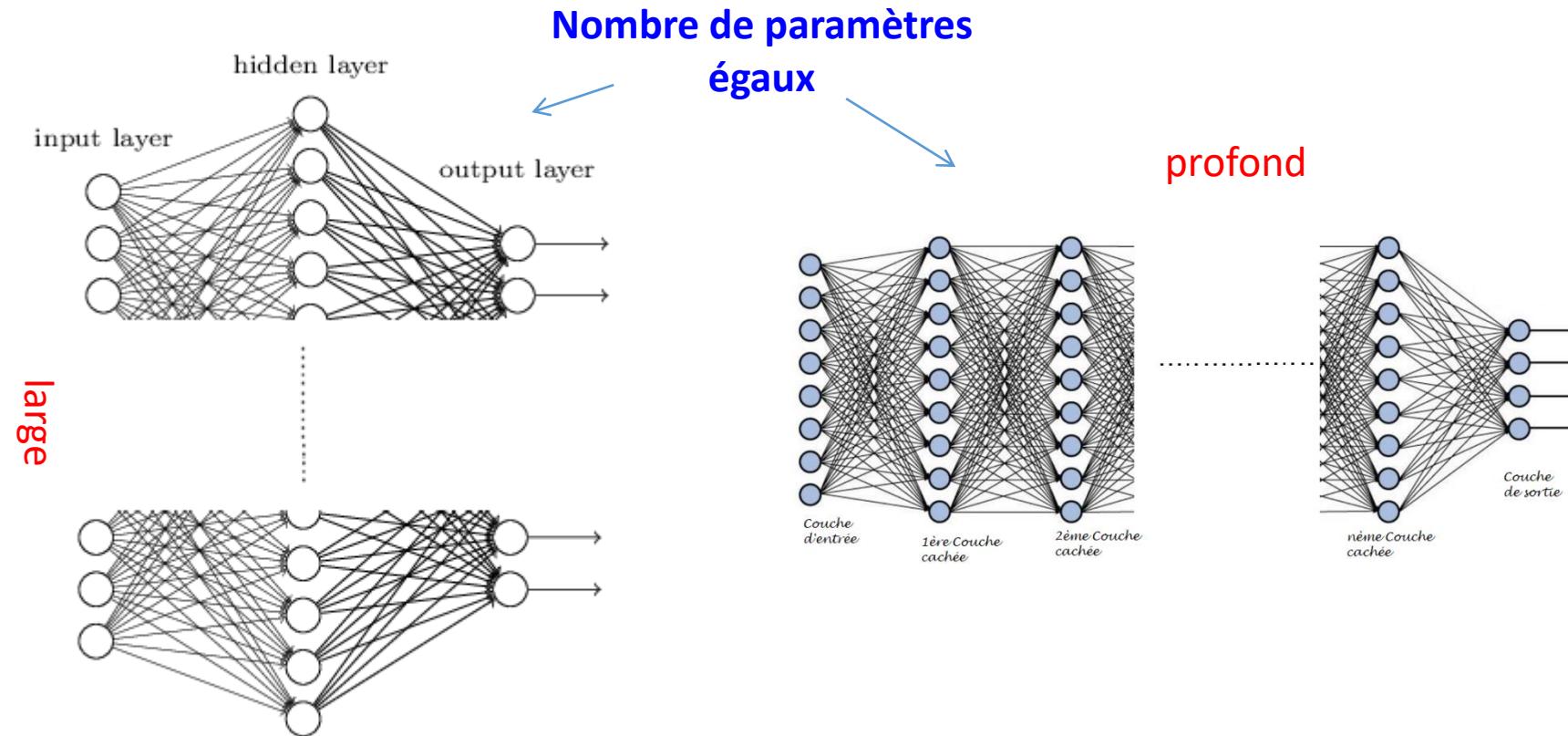
- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN vs CNN

ANN	CNN
<ul style="list-style-type: none">• Pas d'apprentissage du modèle dans la hiérarchie spatiale, c'est-à-dire que les couches supérieures de l'ANN ne combinent pas les sorties des couches inférieures.	<ul style="list-style-type: none">• Apprentissage hiérarchique spatiale des patterns, c'est-à-dire que les couches supérieures de CNN sont formées en combinant des couches inférieures. Cela aide à identifier les modèles plus efficacement que l'ANN. Les filtres de convolutions différentes tailles (3x3 et 5x5) pour capturer l'espace informations à différentes échelles.La caractéristique intéressante de CNN est sa capacité à exploiter la corrélation spatiale ou temporelle des données. La topologie de CNN est divisée en plusieurs étapes d'apprentissage composées d'une combinaison des couches convolutives, des unités de traitement non linéaires et des couches de sous-échantillonnage.
<ul style="list-style-type: none">• N'est pas invariant par translation, une fois qu'un ANN a appris à reconnaître un modèle à un emplacement, il ne peut le reconnaître qu'à cet emplacement particulier. En bref, nous devons entraîner à nouveau l'image (ne peut pas réutiliser les poids précédents) si elle est pivotée ou décalée.	<ul style="list-style-type: none">• Invariant par translation. Une fois que le CNN a appris à reconnaître un motif à un emplacement, il peut le reconnaître à n'importe quel autre emplacement. En bref, l'apprentissage (poids) peut être réutilisé même si l'image est pivotée ou décalée. De plus, le CNN est invariant aux petites déformations.

ANN vs CNN

- I. ML vs DL
- II. Limites REC
- III. Une approche biologique
- IV. Conv 2D et filtrage
- Convolution 2D
- V Padding 2D
- Conv Avec t sans Padding
- Autres Réseaux
- Pax Pooling
- Flatinning
- Archi des R Profonds
- Syst de recon de motifs
- Exemple de RN profonds
- ANN Vs CNN

Réseaux larges ou réseaux profonds ?



Pourquoi est-ce qu'un nombre de paramètres égale à des réseaux fins est profond s ont-ils des performances meilleures que les réseaux large et peu profond ? d'où vient la déraisonnable efficacité de la profondeur ?