

Apprentissage de descripteurs non-supervisé

Ioan Marius Bilasco
FOX, CRIStAL

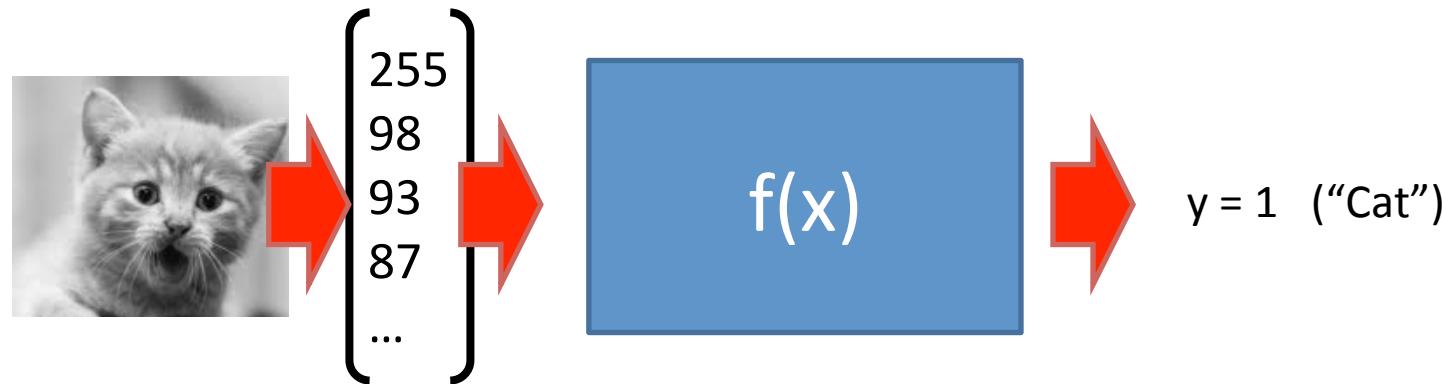
supports inspirés par les travaux d'Adam Coates
<http://www.apcoates.com/>

Apprentissage supervisé

- Un ensemble d'apprentissage :

$$\mathcal{X} = \{(x^{(i)}, y^{(i)}) : i = 1, \dots, m\}$$

- Par exemple: $x^{(i)}$ = intensités des pixels
 $y^{(i)}$ = id de la classe de l'objet



- Objectif: trouver $f(x)$ pour prédire y à partir de x sur la base de l'ensemble d'apprentissage
 - avec l'espoir de fonctionner également sur d'autres données

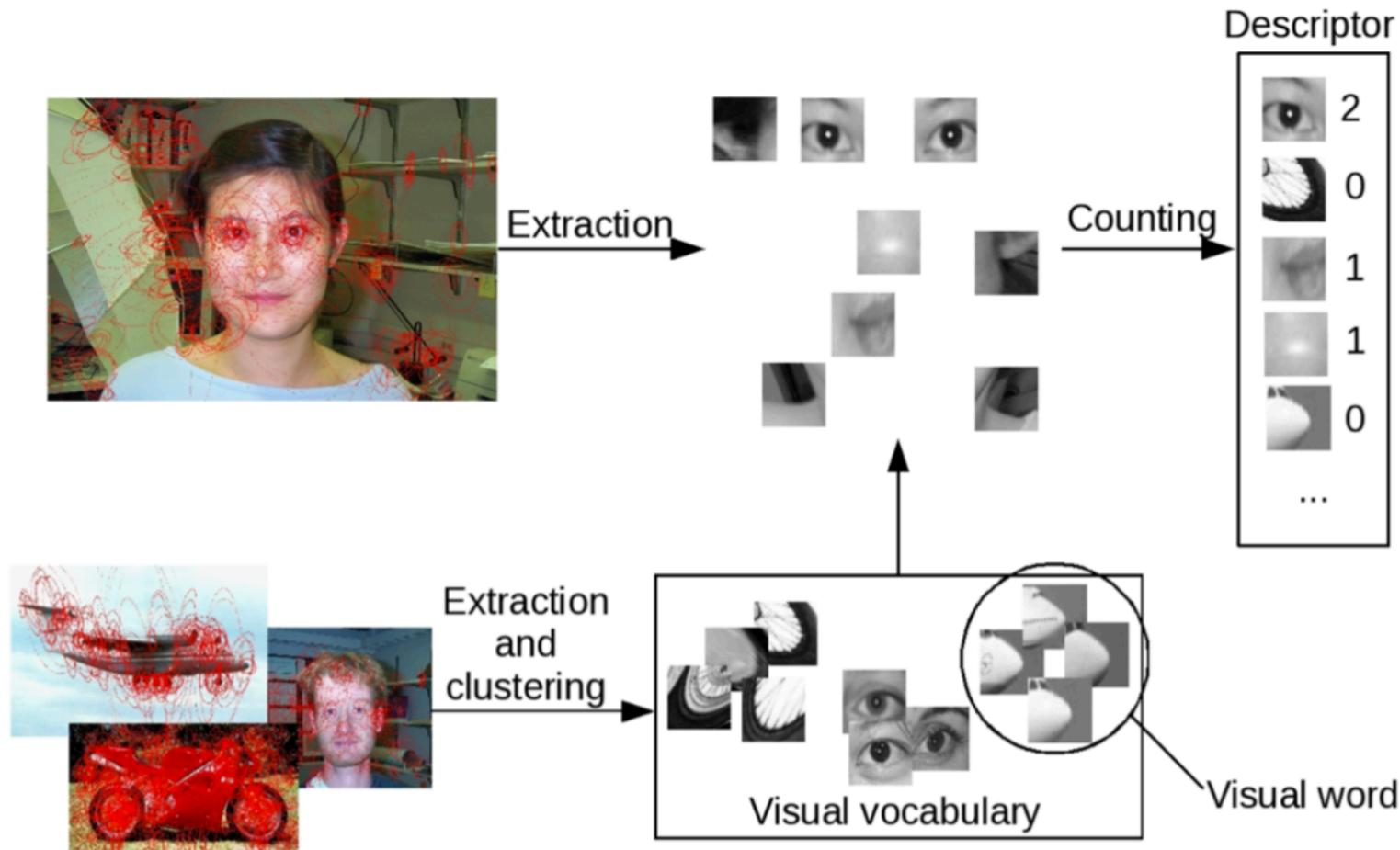
Descripteurs

- Jusque là calculés de manière explicite
 - SIFT, histogrammes, Bag Of Words, VLAD
 - Formellement, une fonction qui projette une entrée brute dans une représentation de plus haut niveau
$$\Phi(x) : \Re^n \rightarrow \Re^K$$
- Choisir la bonne représentation parmi celles disponibles ou en construire des nouvelles
- Faciliter l'apprentissage

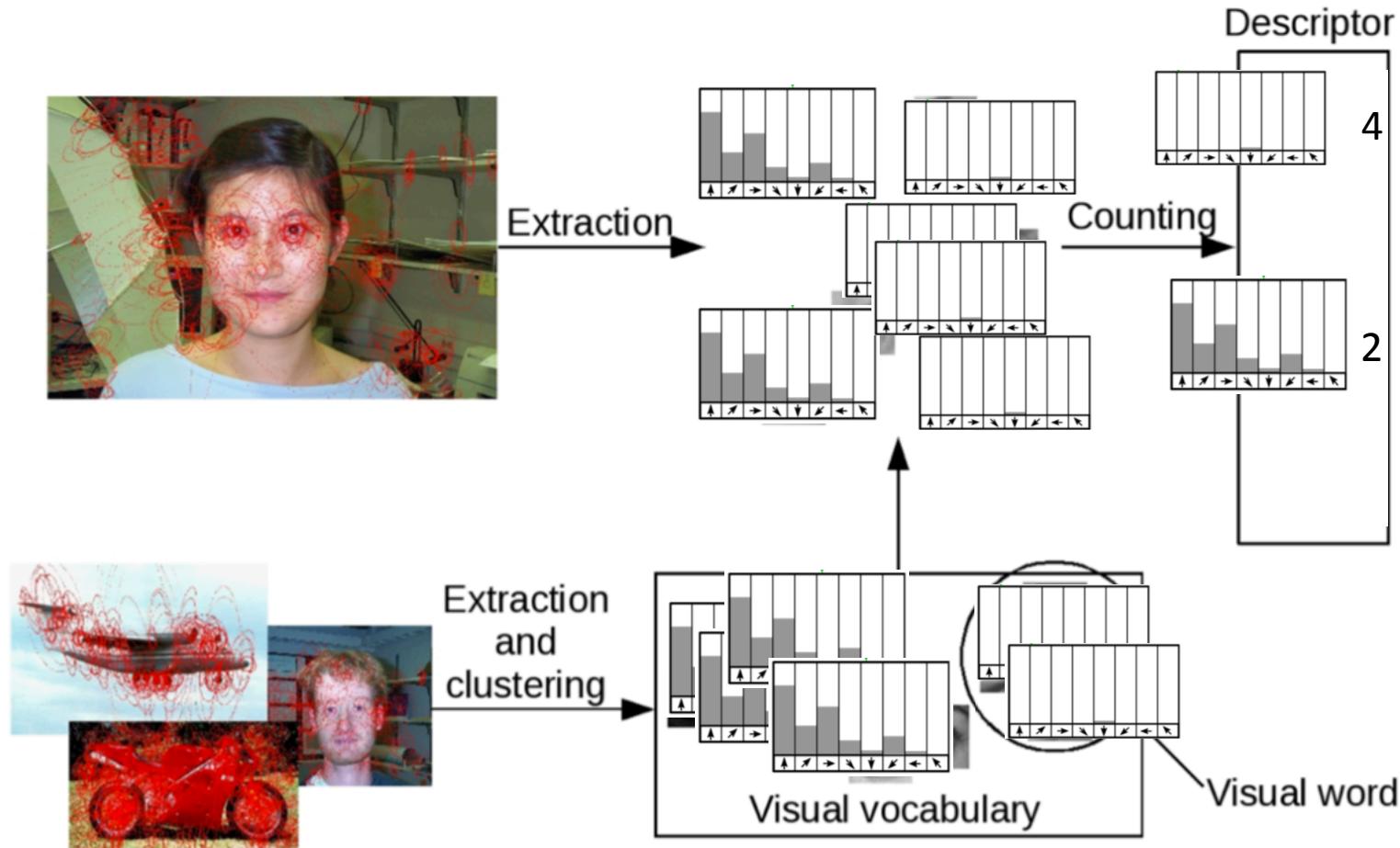
Apprendre des nouveaux descripteurs

- Couvrant un large spectre de patrons : si le descripteur est un vecteur à K dimensions, il est possible de représenter 2^k patrons différents
- Invariance : robustesse aux changements (position, taille)
- Offrir différents angles d'analyse : caractériser différents concepts (e.g., couleur, contours, orientations) dans différents dimensions du vecteur de caractéristiques.

Revenons sur les BoW



Revenons sur les BoW



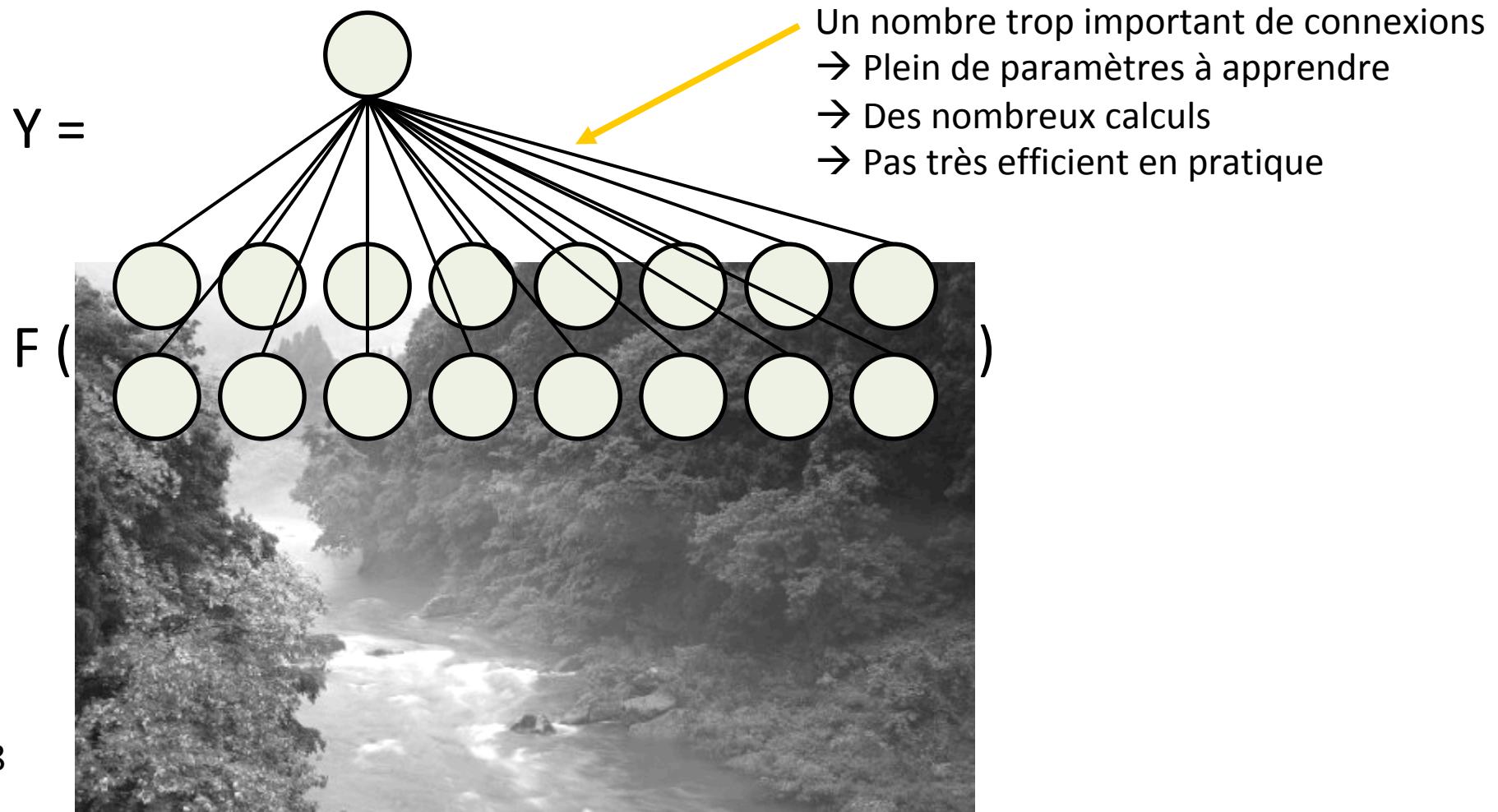
Les mots visuels sont calculés à partir des descripteurs génériques (ici SIFT) conçus sans tenir compte des spécificités des données traitées

... et si on se passait des SIFTs

- Travailler sur l'intégralité de l'image
- Identifier des petites régions (unités d'analyse) dans l'image qui apparaissent de manière récurrente au sein des images des données spécifiques

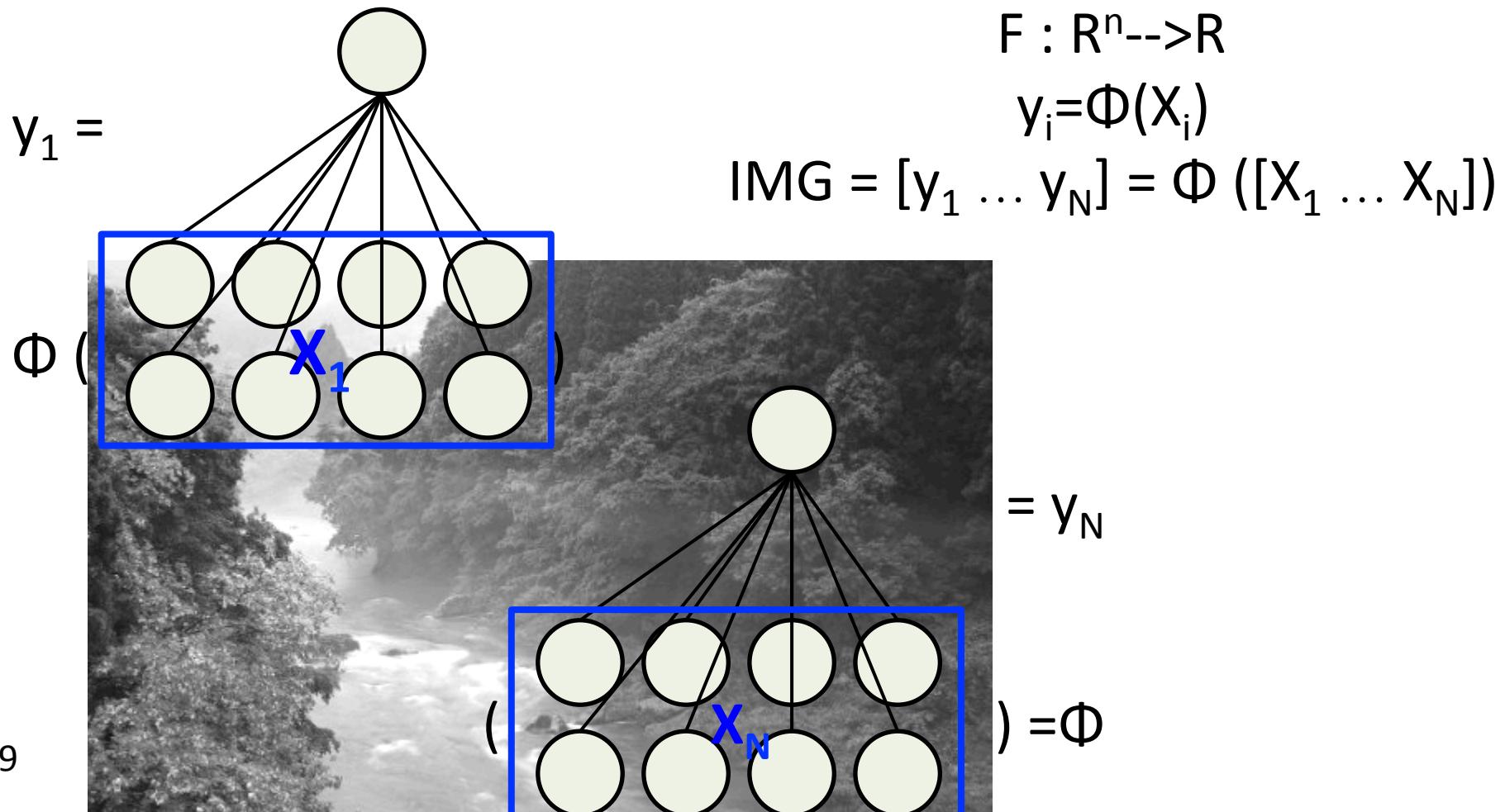
Travailler directement avec les images

- Champ de perception visuel



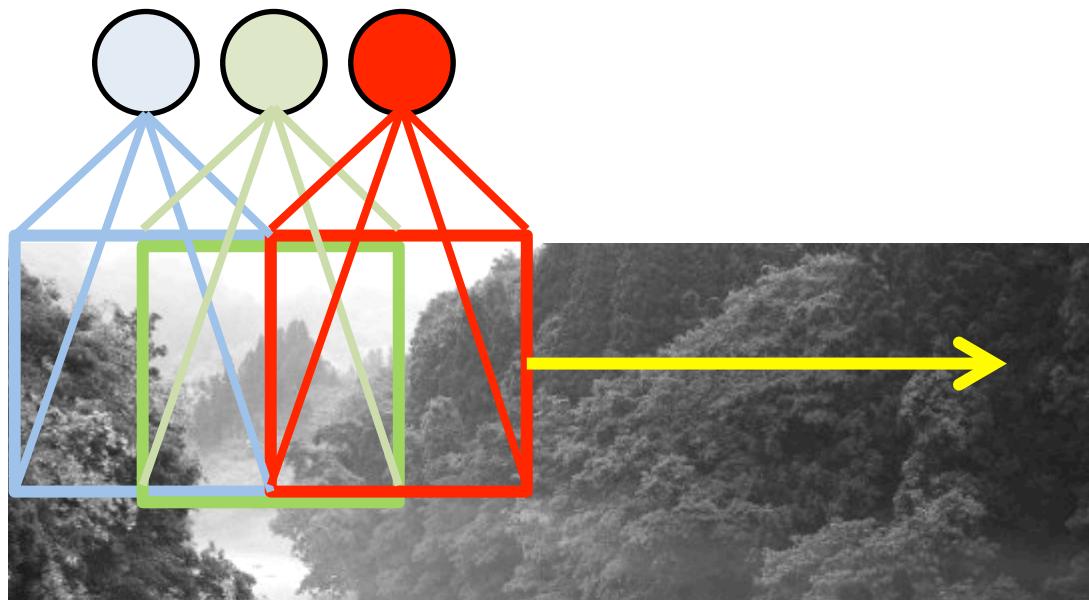
Analyse locale

- Réduire le nombre de paramètres
 - en reportant les mêmes à différents endroits de l'image...
 - ... un peu comme le SIFT



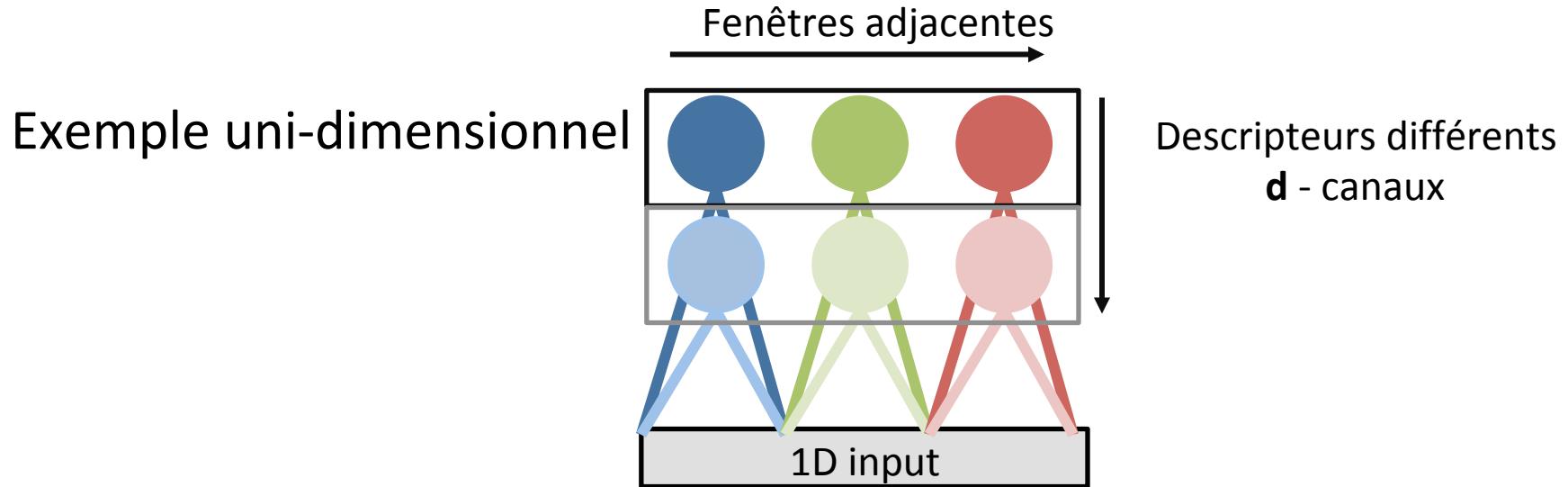
Analyse locale

- Un descripteur offre une vue spécifique en balayant les régions adjacentes de l'image
 - Préciser la taille de l'unité d'analyse
 - Préciser le recouvrement des zones adjacentes



Analyse locale multidescripteurs

- Plusieurs fonctions différentes peuvent être appliqués à une même unité d'analyse (patch)



$$F : R^n \rightarrow R^d$$

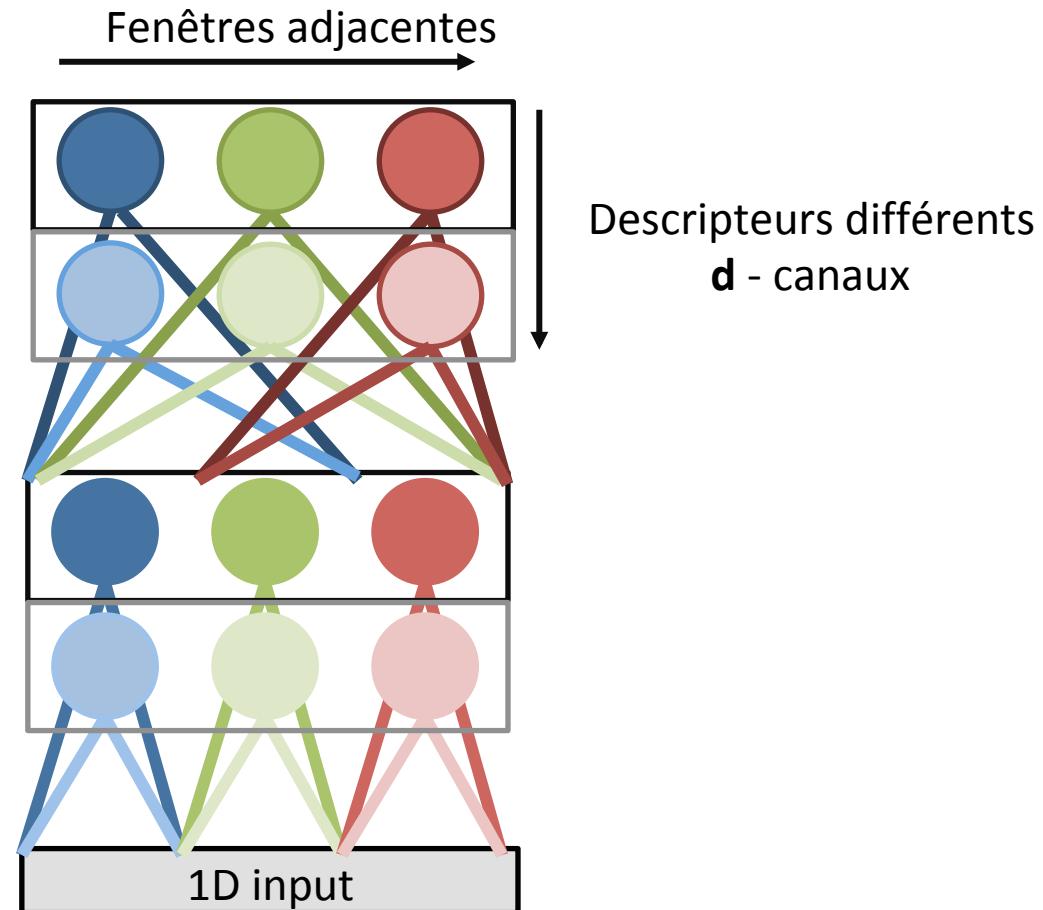
$$Y_i = \Phi(X_i)$$

$$IMG = [Y_1 \dots Y_N] = \Phi([X_1 \dots X_N])$$

Analyse locale multidescripteurs

- Enchaîner les niveaux et construire des descripteurs de plus complexes

Exemple uni-dimensionnel



Identifier les patrons

- Analyser les différents unités extraites de l'ensemble des images à traiter.
- Construire des clusters



16 unités
(patchs)
 $(w * w * 3)$



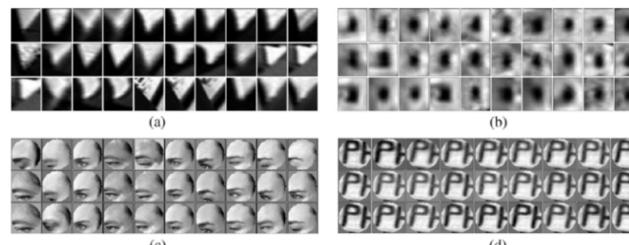
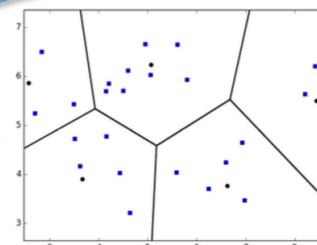
16 u $(w * w * 3)$



28 u $(w * w * 3)$

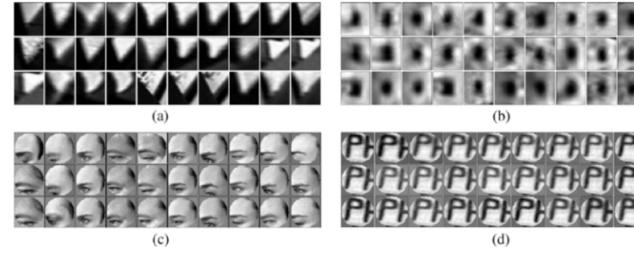
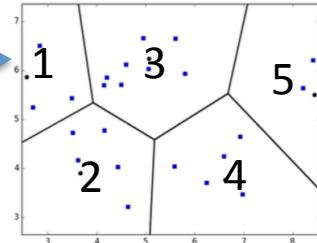


28 u $(w * w * 3)$



[Sivic et Zisserman, 2003]

Exploiter le partitionnement

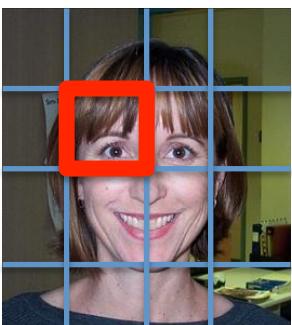


[Sivic et Zisserman, 2003]

$$\Phi(\boxed{\square}) = [_, _, _, _, _]$$

Affecter une valeur à l'unité par rapport à son positionnement par rapport aux clusters.

La valeur place le $\boxed{\square}$ dans l'espace R^d où d le nombre de clusters



14 16 u ($w * w$)



16 u ($w * w$)



28 u ($w * w$)



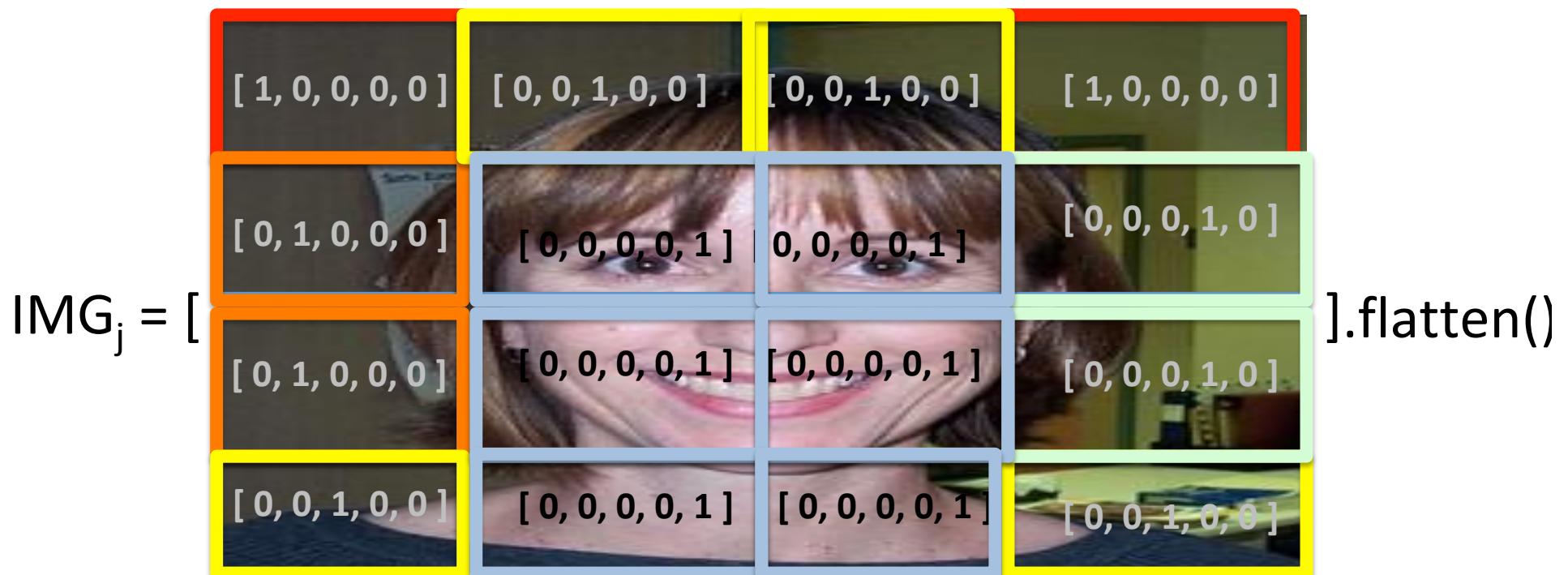
28 u ($w * w$)

Décrire une unité d'analyse

- Hard

$$-\Phi(\boxed{}) = [v_1, _, _, _, v_d]$$

où $v_i=1$ si $\boxed{}$ appartient au cluster i
 $v_i=0$ sinon



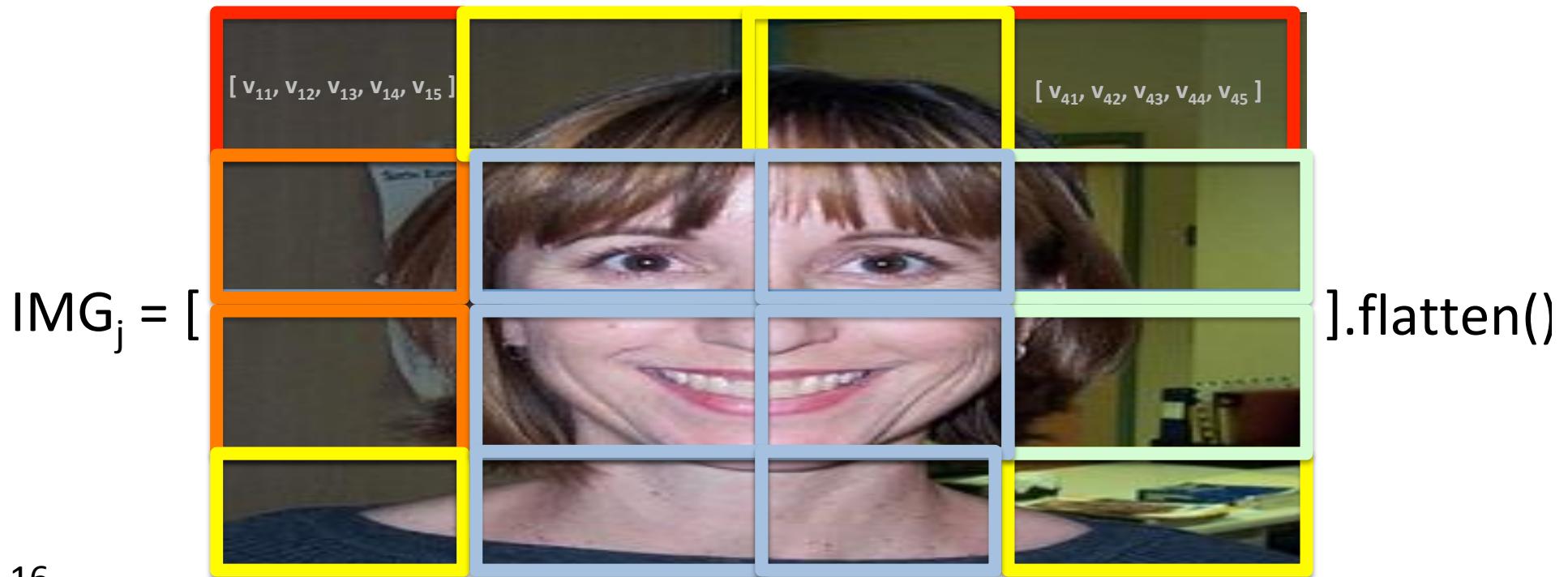
Décrire une unité d'analyse

- Soft

$$-\Phi(\boxed{\quad}) = [v_1, _, _, _, v_d]$$

où $v_i = 1/(1+\exp(-d_i))$

avec $d_i = \text{distance}(\boxed{\quad}, \text{centre cluster}_i)$

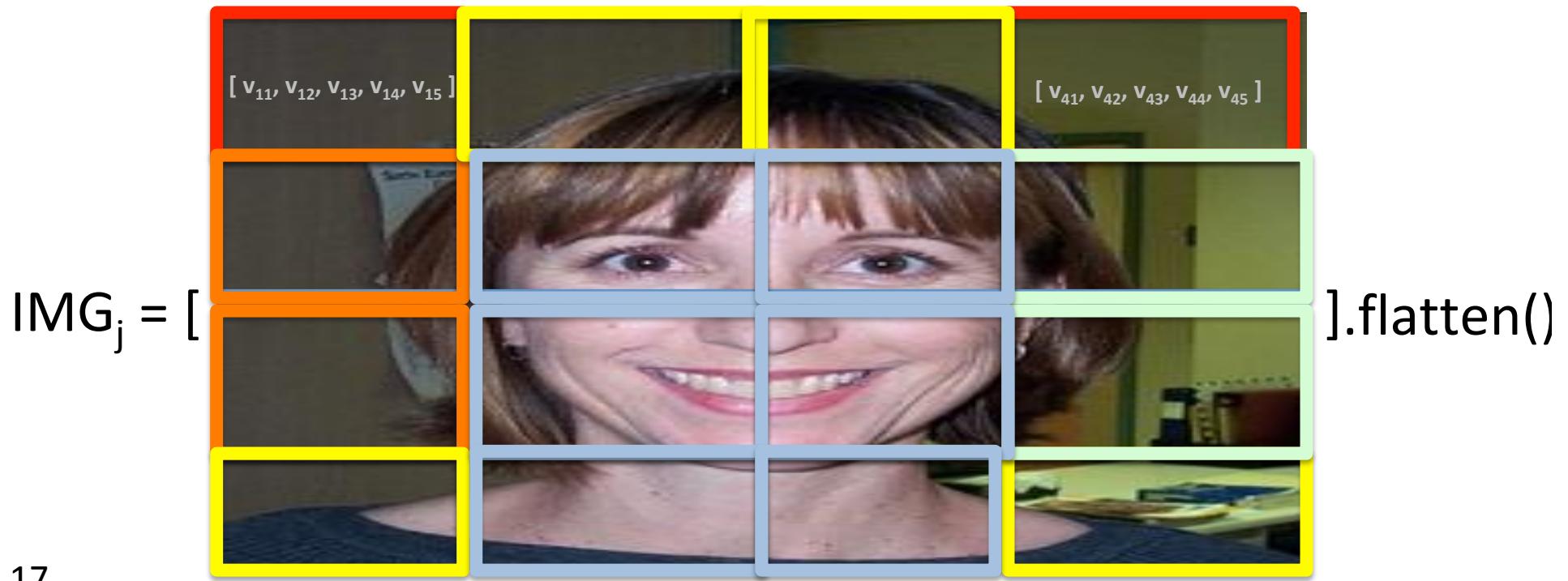


Décrire une unité d'analyse

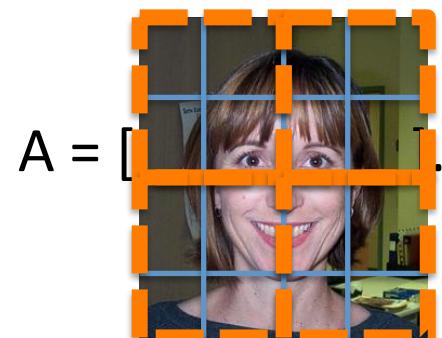
- Convolution

$$-\Phi(\square) = [v_1, _, _, _, v_d]$$

où $v_i = \text{sum}(\square * \text{centre cluster}_i)$

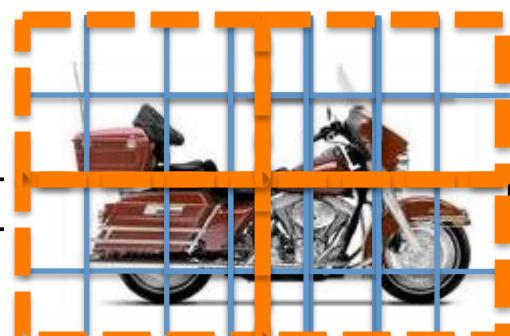
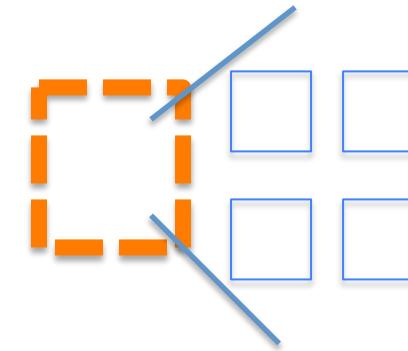


Uniformiser la description



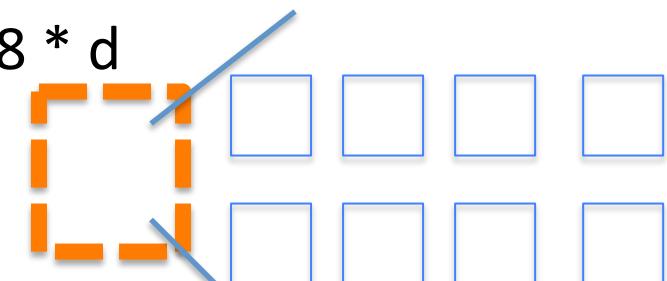
A = [...].flatten().shape --> 16 * d

16 u (w * w * 3)



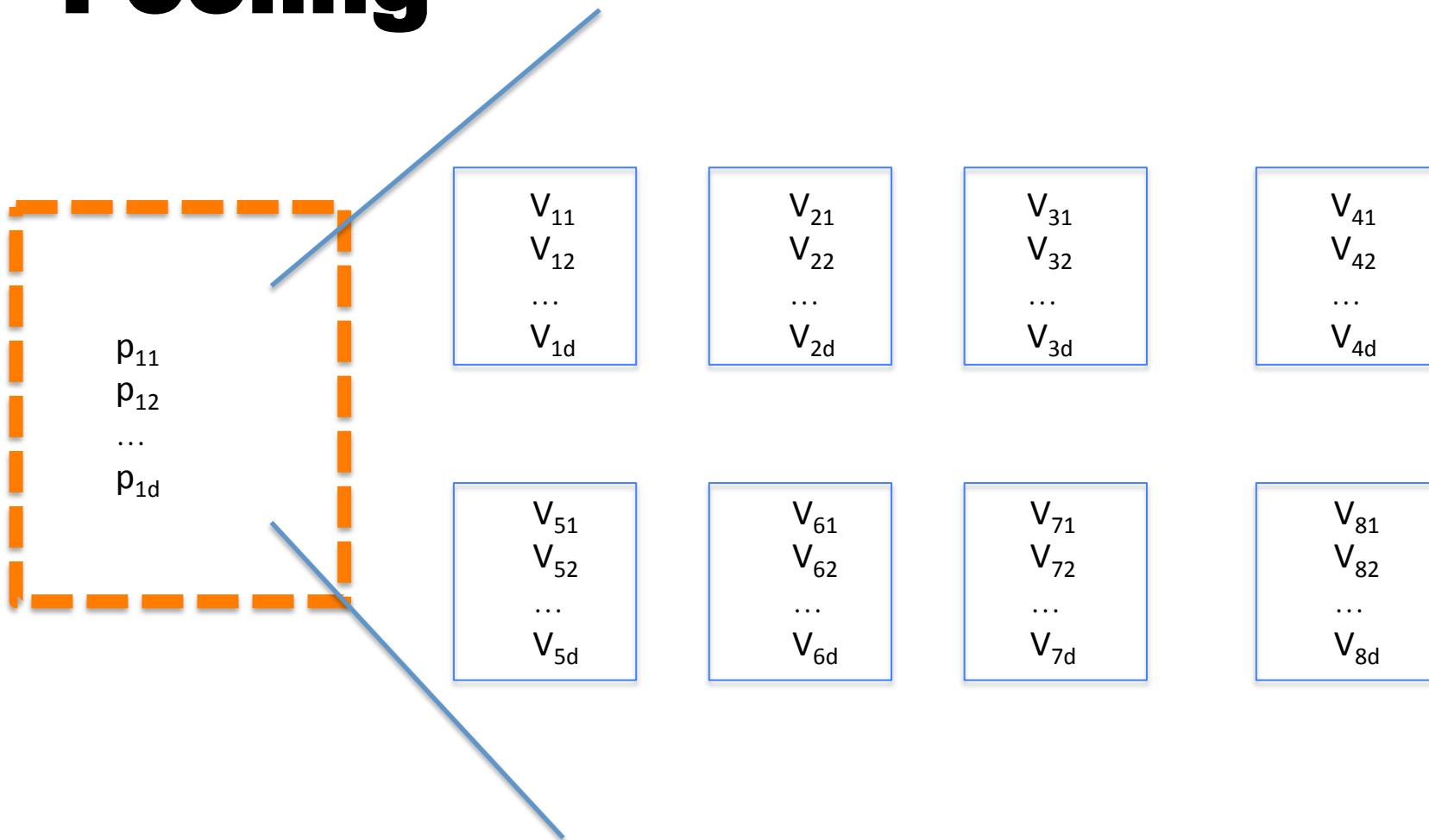
B = [...].flatten().shape --> 28 * d

28 u (w * w * 3)



- Définir une grille de taille fixe pour tout image
- Mécanisme de pooling à noyau de taille variable

Pooling



$p_1 = \text{Pooling}_{\text{avec } i \text{ entre } 1 \text{ et } 8} (v_i)$

$p_{1j} = \text{Pooling}_{\text{selon } j, \text{ avec } i \text{ entre } 1 \text{ et } 8} (v_{ij})$

Max-pooling
Sum-pooling
Avg-pooling

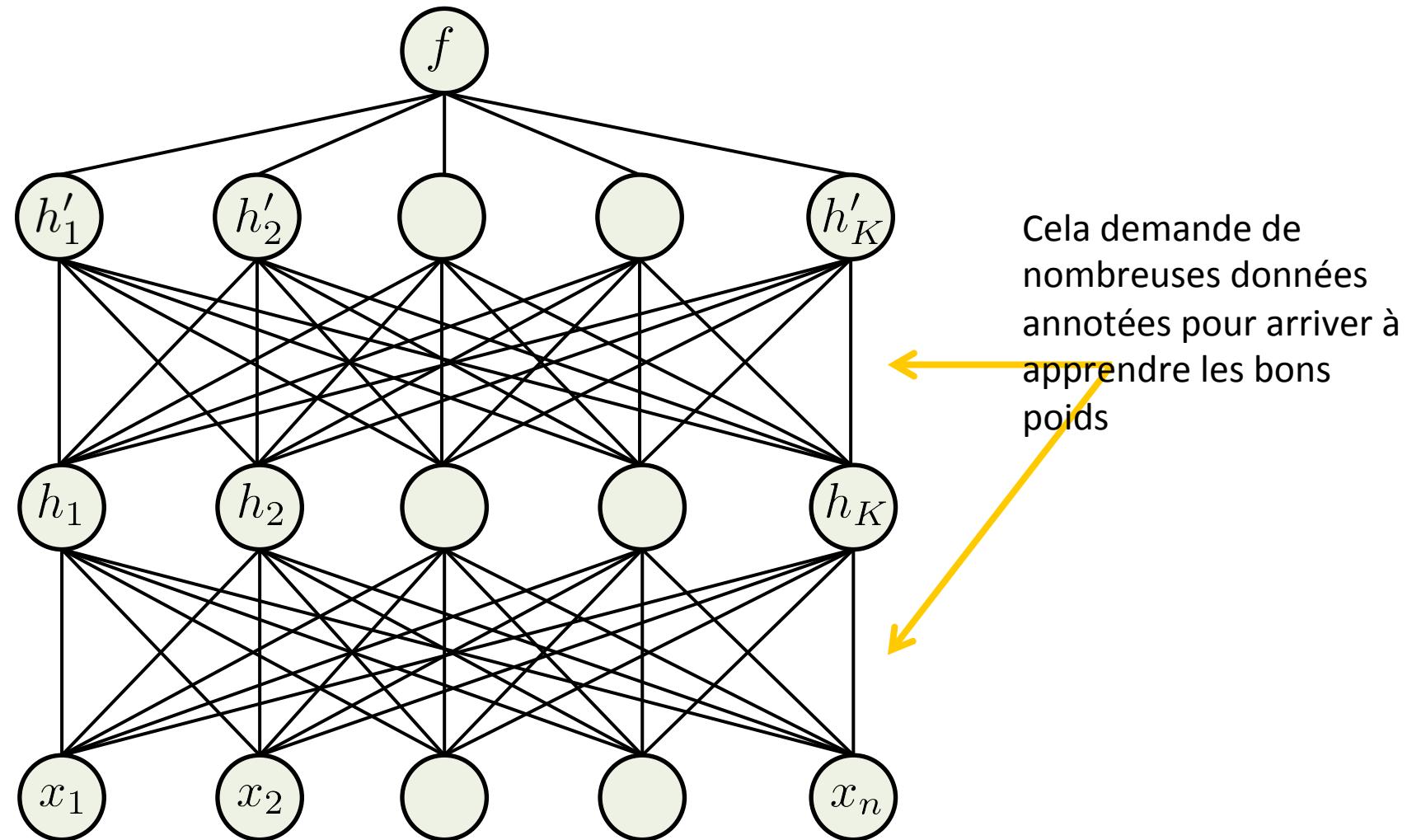
Résumons...

- Extraction d'unités d'analyse d'un ensemble d'images
- Identifier les centres d'un partitionnement à K clusters
 - K clusters définissent le nouvel espace de représentation
- Extraire toutes les unités d'analyse d'une image
- Affecter à chaque unité une valeur dans le nouvel espace (à k dimensions)
- Pooling à noyau variable pour uniformiser la taille des représentations pour une image
- Classification

... vous explorerez tout cela en TP

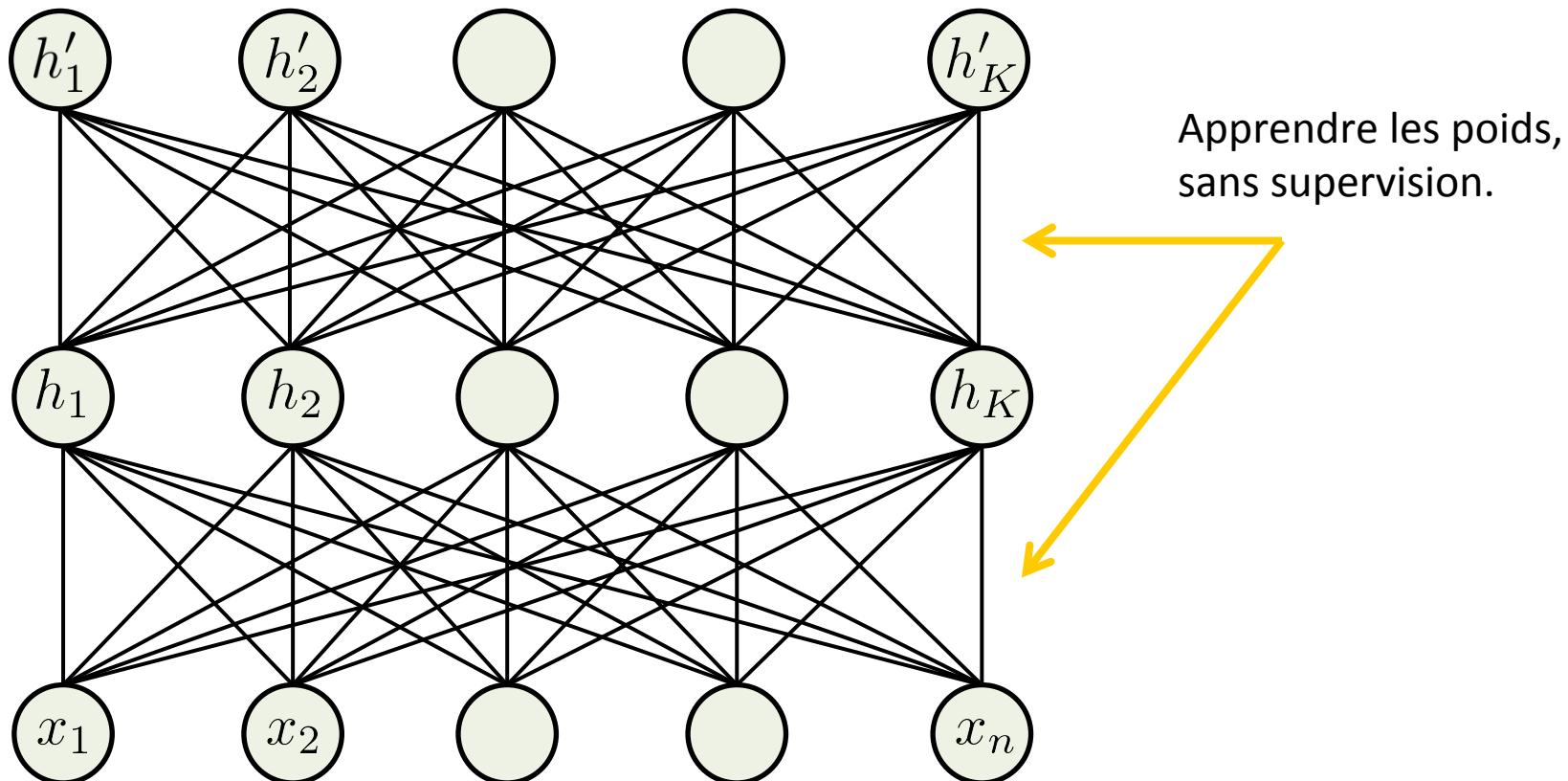
Et l'apprentissage profond ?

- Dans l'apprentissage profond supervisé, on entraîne des *descripteurs* pour une tâche spécifique



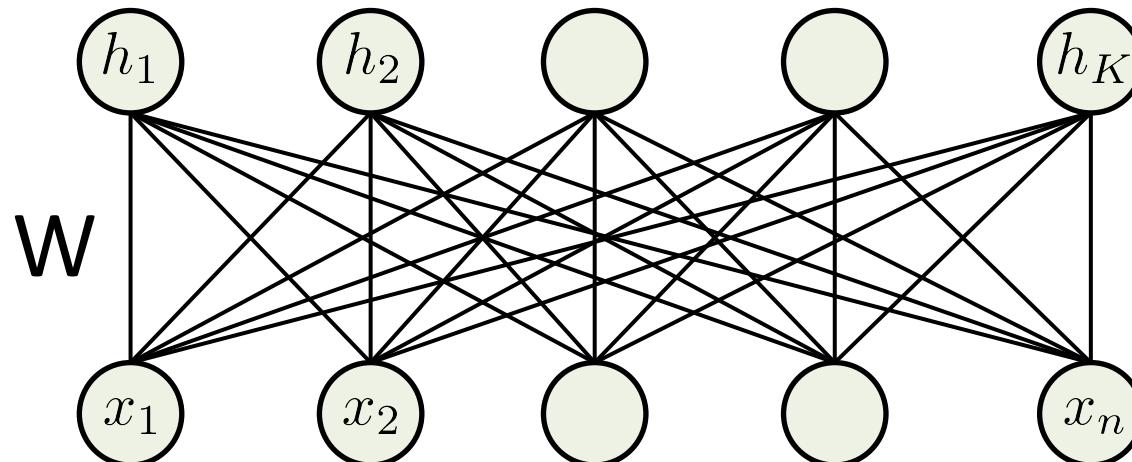
Apprentissage profond non-supervisé

- Peut-on entraîner des descripteurs sans supervision?



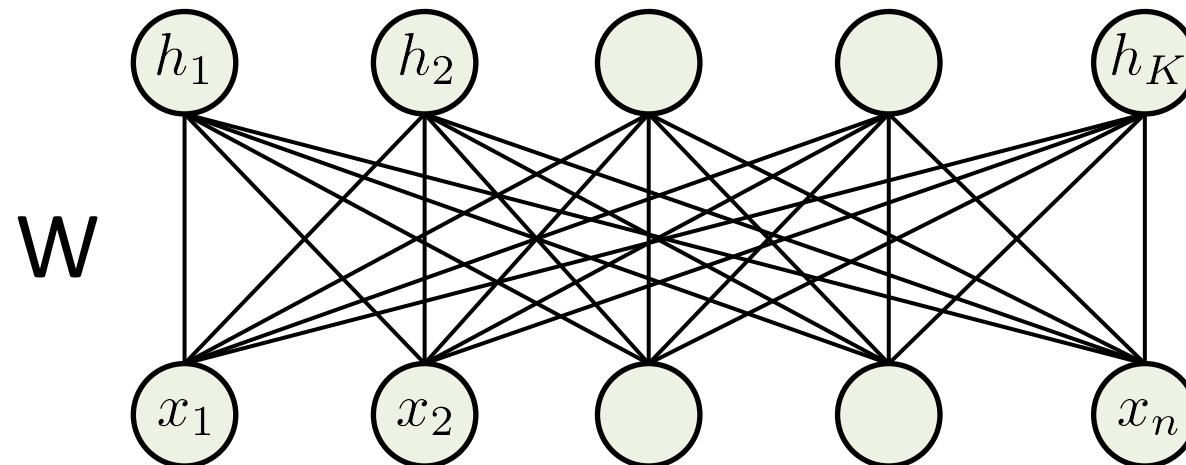
Apprentissage non-supervisé

- Apprendre des représentations avec des données non-annotées.
 - Minimiser une loss non-supervisée
 - Couche par couche
- Ensuite, e.g., apprendre de manière supervisée une tâches spécifique (“Self-taught learning” - Raina et al., ICML 2007)



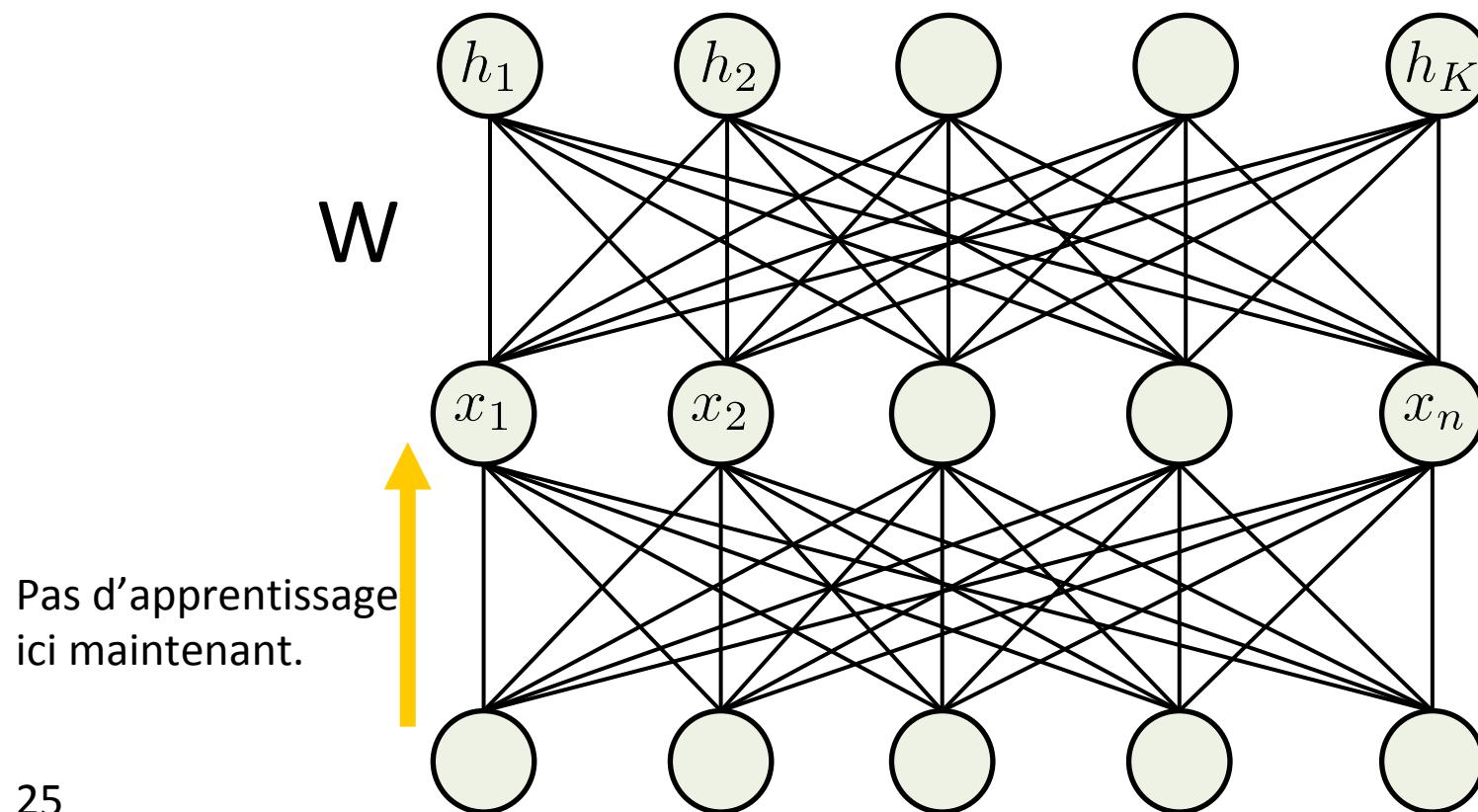
Apprentissage couche par couche

- Apprendre sans données annotées
 - Apprendre le premier niveau d'abord



Apprentissage couche par couche

- Apprendre sans données annotées
 - Apprendre le premier niveau d'abord
 - Apprendre le 2e niveau en se servant des sorties de la première couche



Apprentissage non-supervisé - exemples

- Approches pour obtenir des bons descripteurs:
 - Reconstruction: recréer les données à partir des descripteurs.

$$\mathcal{L}_{\text{recons}}(W_2, W_1) = \sum_i ||W_2 h(W_1 x^{(i)}) - x^{(i)}||_2^2$$

- Parcimonie: décrire de manière aussi succincte que possible les données d'entrées

$$\mathcal{L}_{\text{sparse}}(W_1) = \sum_i ||h(W_1 x^{(i)})||_1$$

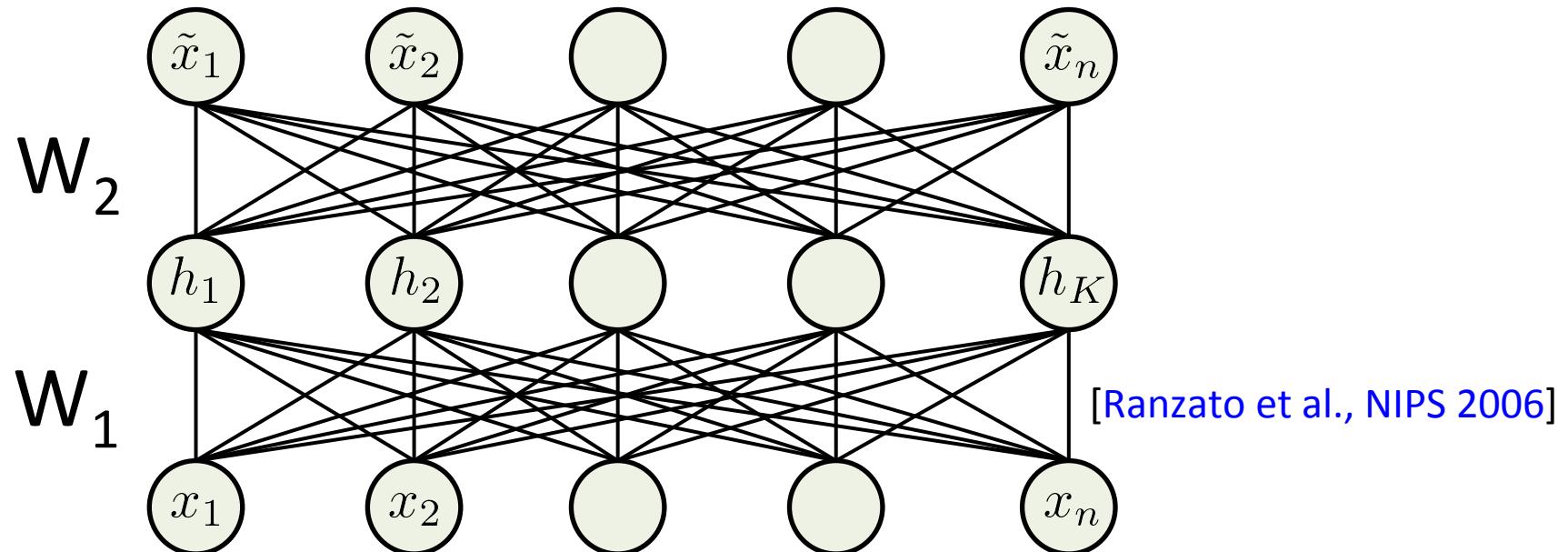
Auto-encoder parcimonieux

- Un ANN à deux couches

$$\underset{W_1, W_2}{\text{minimize}} \sum_i \|W_2 h(W_1 x^{(i)}) - x^{(i)}\|_2^2 + \lambda \|h(W_1 x^{(i)})\|_1$$

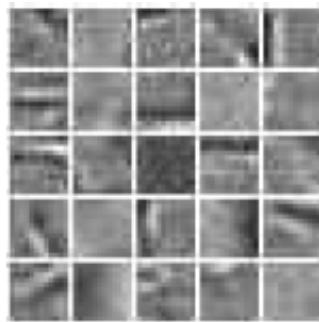
$$h(z) = \text{ReLU}(z)$$

- Après apprentissage, retirer la partie “decoder” et utiliser les descripteurs appris (h).

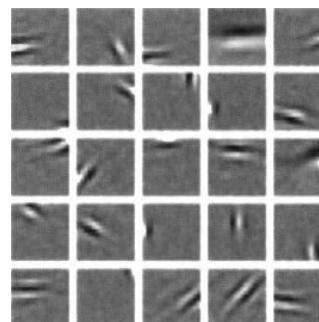


Qu'est-ce qu'on apprend ?

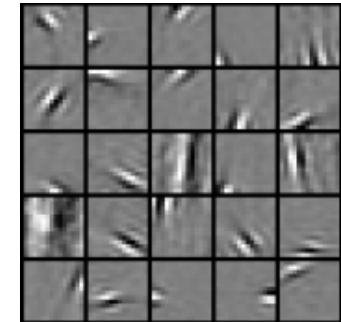
- Lorsque l'on applique aux unités d'analyse



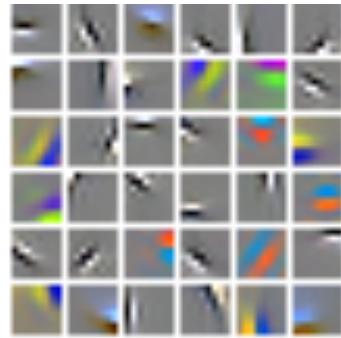
Sparse auto-encoder
[Ranzato et al., 2007]



Sparse coding
[Olshausen & Field, 1996]



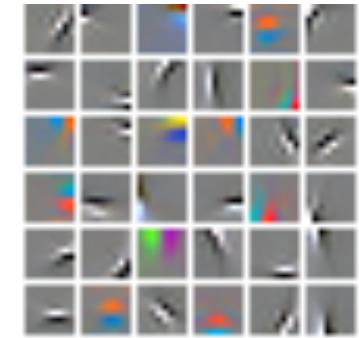
Sparse RBM
[Lee et al., 2007]



Sparse auto-encoder



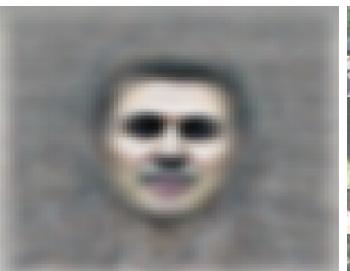
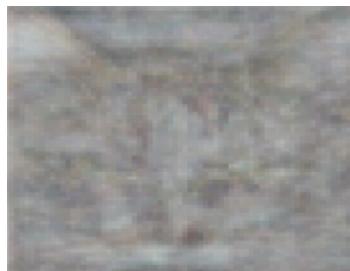
K-means



Sparse RBM

Descripteurs complexes

- Les travaux de [Le et al., 2012; Coates et al., 2012] montrent que les descripteurs de niveau supérieur peuvent apprendre des concepts non-triviaux
 - Par exemple, des descripteurs qui répondent de manière forte aux chats et visages



[Le et al., ICML 2012]



[Coates, Karpathy & Ng, NIPS 2012]

Revenons au TP

- Construction des descripteurs avec KMeans
- Penser à comparer les résultats obtenus avec BoWs et VLAD avec les résultats obtenus avec « vos » descripteurs
- ... pour aller plus loin vous pourrez ensuite adapter le TP en utilisant les auto-encodeurs et comparer les performances