université PARIS-SACLAY

RÉSEAUX CONVOLUTIONNELS POUR L'IMAGE

Vincent Guigue,
inspiré des supports de Nicolas Baskiotis & Benjamin Piwowarski

MIA PARIS-SACLAY
EKINOCS

AGROPARISTECH
Institut des Sciences et Industries du Vivant et de l'Environnement

Introduction

Le domaine du *Computer Vision*

Pourquoi les MLP ne sont pas suffisants ?

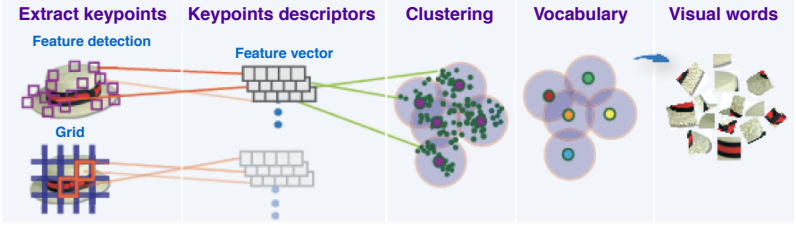
- Image = grille de pixels $\Rightarrow$ sémantique très pauvre
- Fully-connected (MLP) $\Leftrightarrow$ une entrée = un rôle propre
  $\Rightarrow$ dépendance à la position du pixel

$\neq$ Dans une image, l'absolu n'a pas d'importance, le relatif est bien plus important (mais pas seulement)

Les *invariants* sont très importants en image car un objet peut prendre de multiples formes !

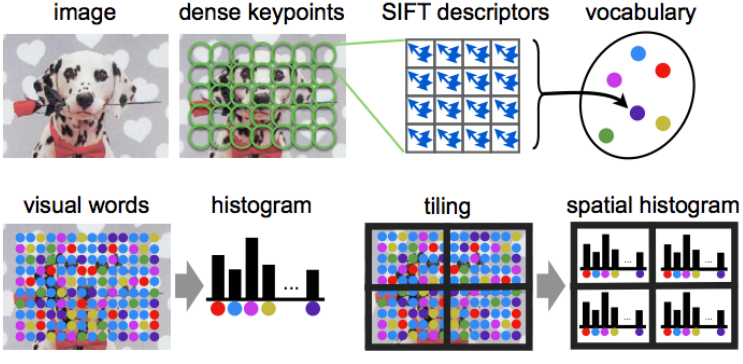
Avant le deep

1. extraire un dictionnaire de *features*
2. agréger les features (représentation type *Bag Of Word*)
3. utiliser un classifieur sur cette représentation pour classifier

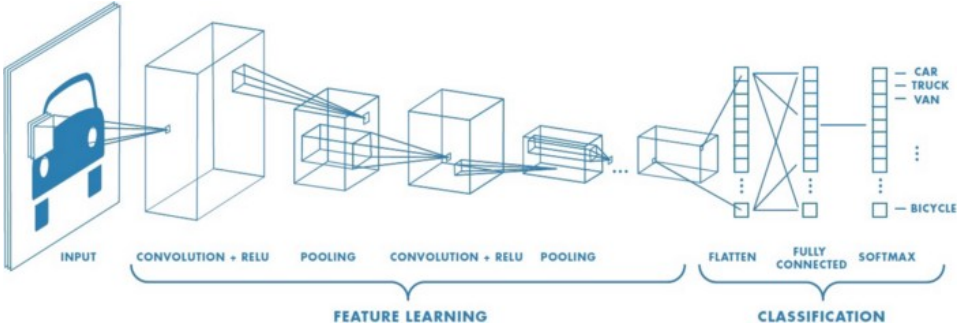


Avant le deep

1. extraire un dictionnaire de *features*
2. agréger les features (représentation type *Bag Of Word*)
3. utiliser un classifieur sur cette représentation pour classifier



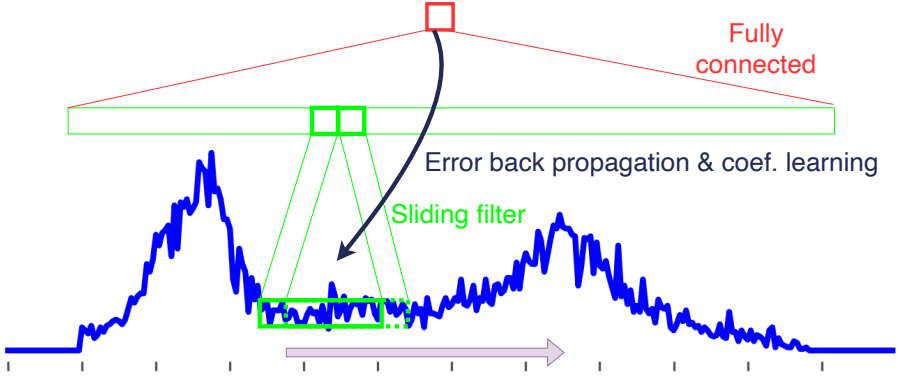
Réseaux convolutifs



Convolution

Convolution 1D

Commençons par un exemple 1D pour bien comprendre.

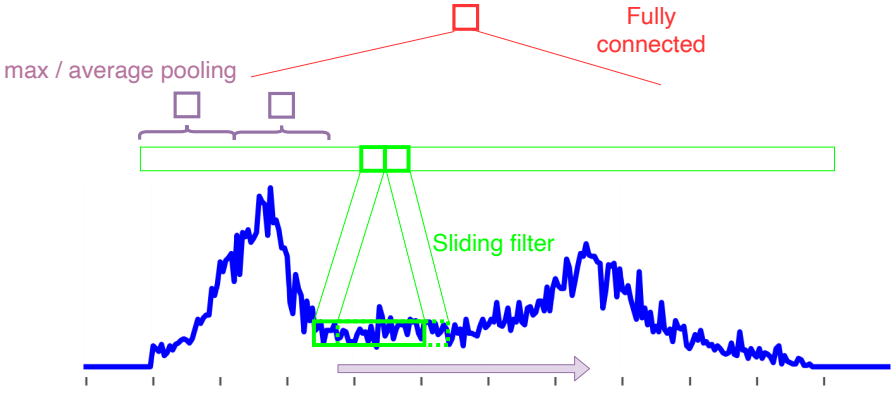


- Fenêtre glissante
- Peu de paramètres
- Largeur / stride / padding

- Apprentissage des motifs à détecter
- Invariance / dépendance à la localisation

Convolution 1D

Commençons par un exemple 1D pour bien comprendre.

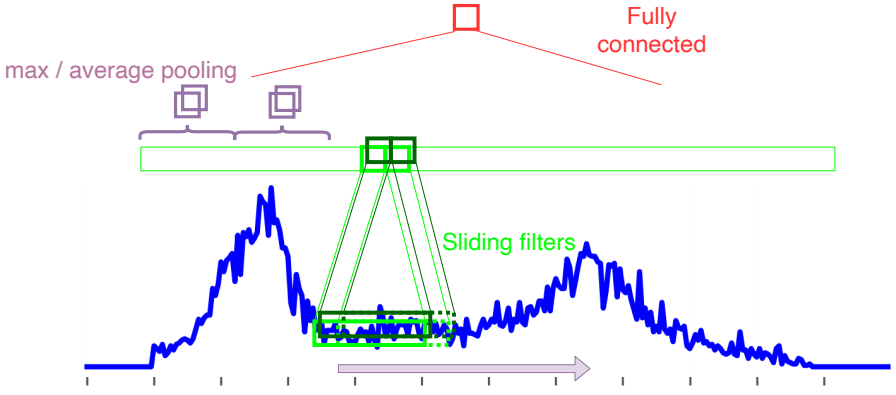


- Fenêtre glissante
- Peu de paramètres
- Largeur / stride / padding

- Apprentissage des motifs à détecter
- Invariance / dépendance à la localisation

Convolution 1D

Commençons par un exemple 1D pour bien comprendre.

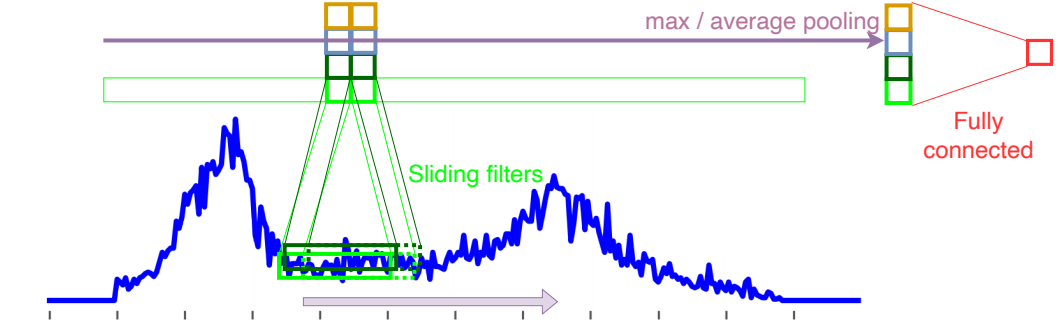


- Fenêtre glissante
- Peu de paramètres
- Largeur / stride / padding

- Apprentissage des motifs à détecter
- Invariance / dépendance à la localisation

Convolution 1D

Commençons par un exemple 1D pour bien comprendre.

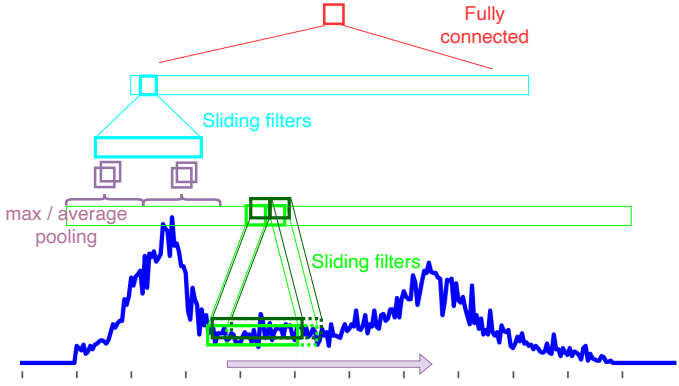


- Fenêtre glissante
- Peu de paramètres
- Largeur / stride / padding

- Apprentissage des motifs à détecter
- Invariance / dépendance à la localisation

Convolution 1D

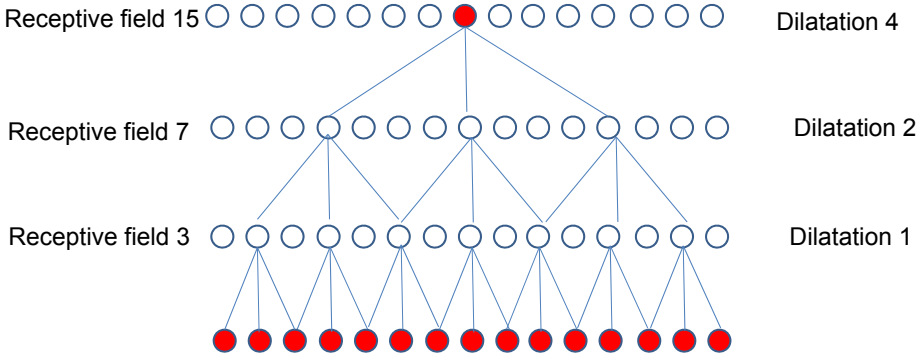
Commençons par un exemple 1D pour bien comprendre.



- Fenêtre glissante
- Peu de paramètres
- Largeur / stride / padding
- Apprentissage des motifs à détecter
- Invariance / dépendance à la localisation

Convolution 1D

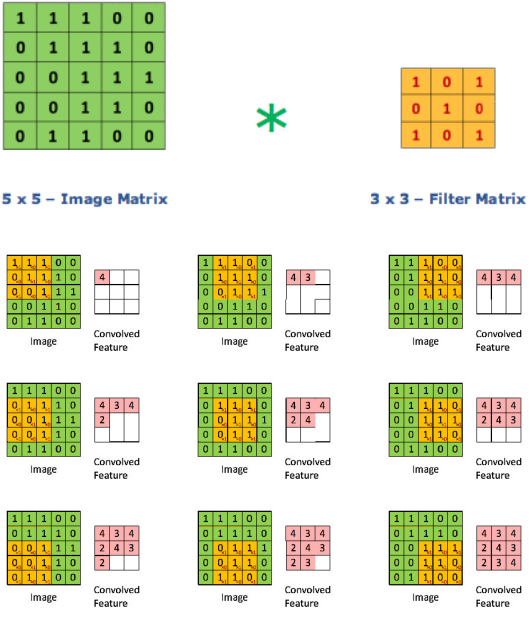
Commençons par un exemple 1D pour bien comprendre.



- Fenêtre glissante
- Peu de paramètres
- Largeur / stride / padding

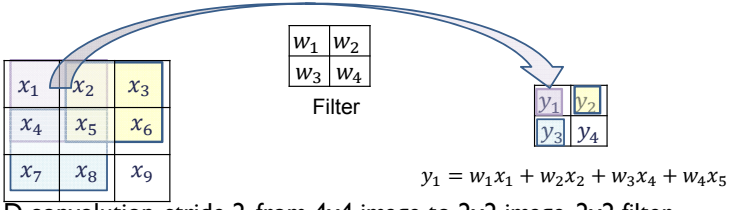
- Apprentissage des motifs à détecter
- Invariance / dépendance à la localisation

Convolution 2D



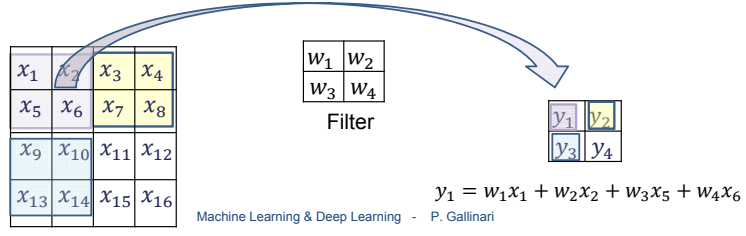
2D/3D CNN

- 2D convolution, stride 1, from 3x3 image to 2x2 image, 2x2 filter



$$y_1 = w_1x_1 + w_2x_2 + w_3x_4 + w_4x_5$$

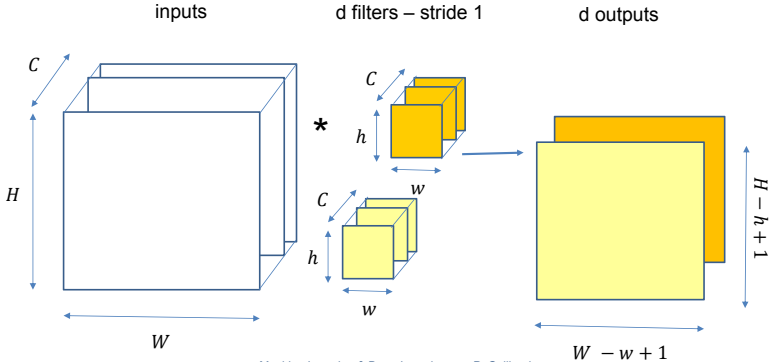
- 2 D convolution, stride 2, from 4x4 image to 2x2 image, 2x2 filter



$$y_1 = w_1x_1 + w_2x_2 + w_3x_5 + w_4x_6$$

+ pooling on spatial 2D windows

2D/3D CNN



Most of the time, we perform $N\times$ 2 dimensional convolutions instead of 3D conv.

$\Rightarrow$ It is linked to the nature of the data

Couche de *Pooling*

Pooling (ou subsampling) : Réduire la dimensionalité de sortie

- Max Pooling : on prend le max sur une fenêtre
- Average Pooling : on fait la moyenne
- Sum Pooling : la somme

...

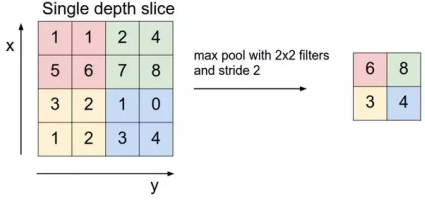
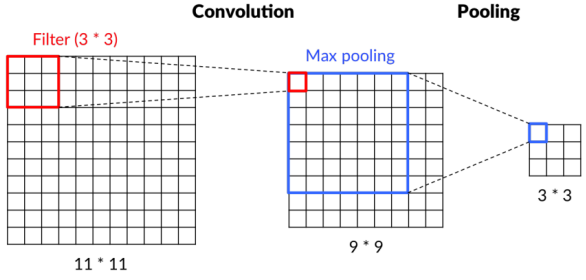
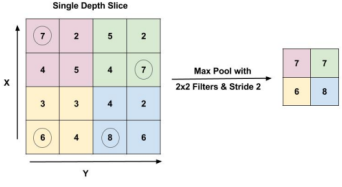
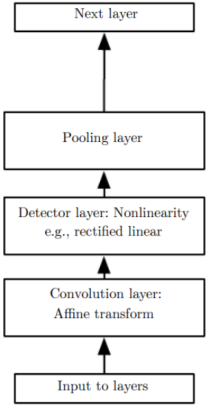


illustration :
https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148

Couche convolutionnelle usuelle







Exemple

Reconnaissance de caractères [Duda et al 00]

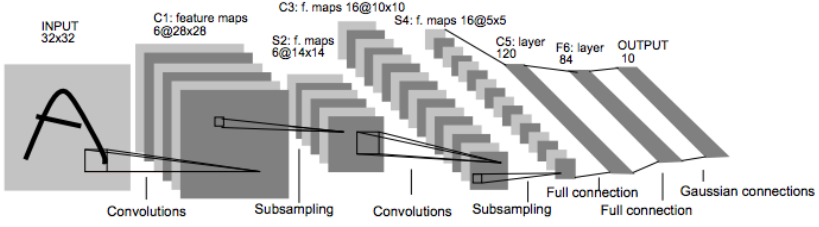
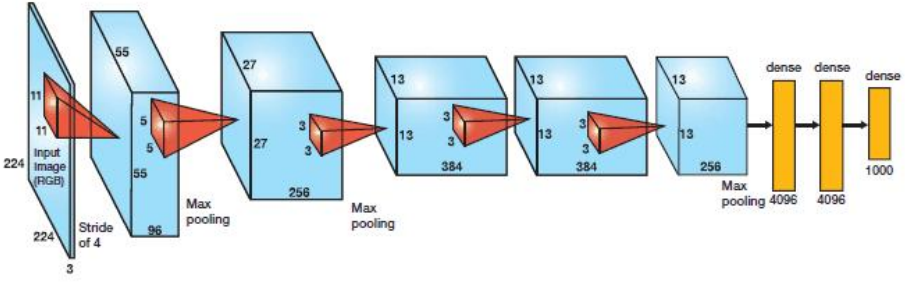


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

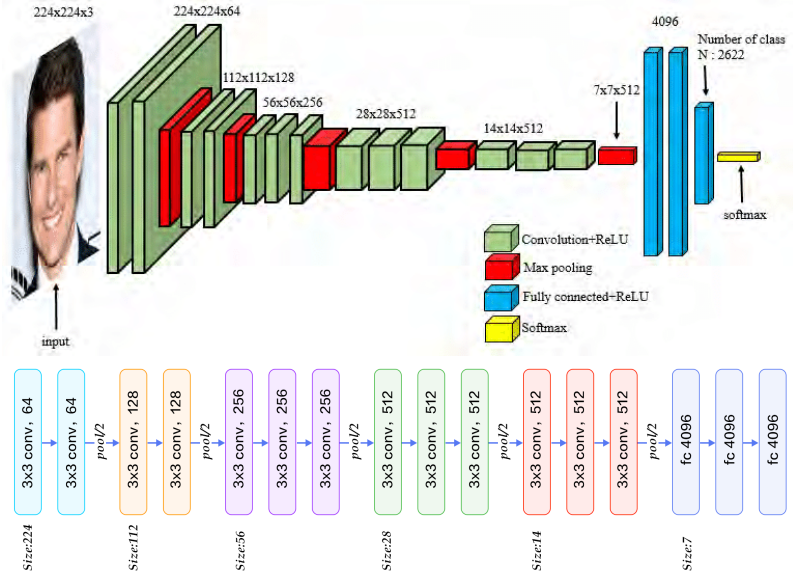
Archis en image

AlexNet (2012)

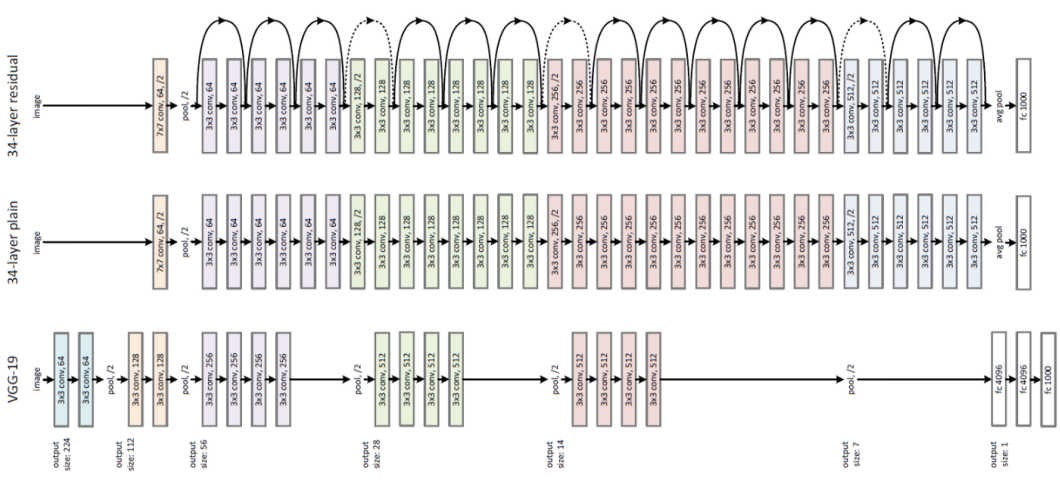


- 11×11, 5×5, 3×3 convolutions
- Max-pooling, ReLU activations
- Dropout et Data-augmentation

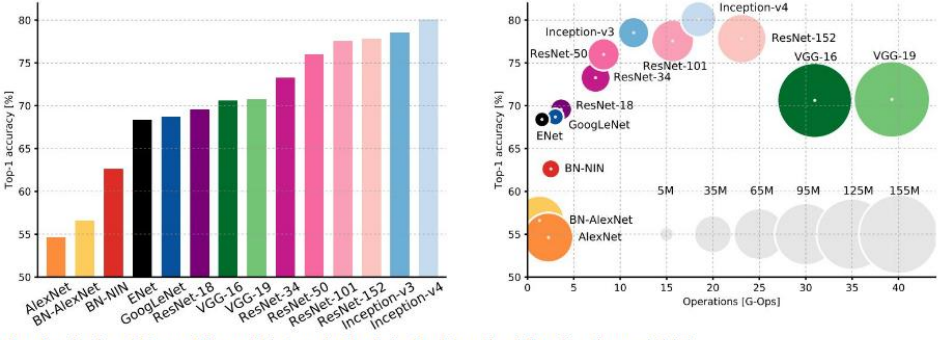
Deep CNN / VGG / ResNet



Deep CNN / VGG / ResNet



Un grand nombre d'architectures



An Analysis of Deep Neural Network Models for Practical Applications, 2017.

Image Reconstruction

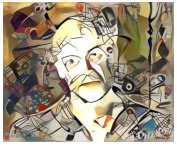
Generate images by combining content and style

Makes use of a discriminatively trained CNN

Image generation

- inverse problem on the CNN

https://deepart.io



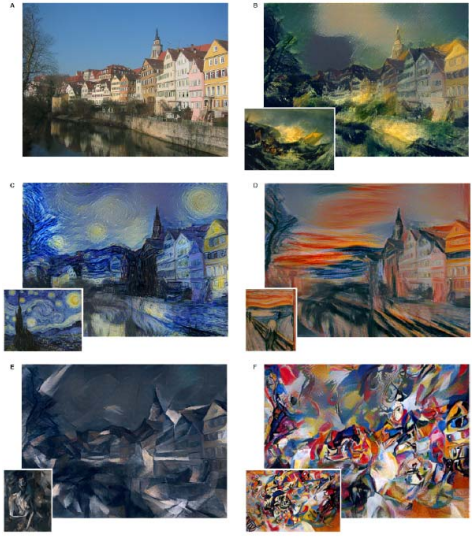


Image Reconstruction

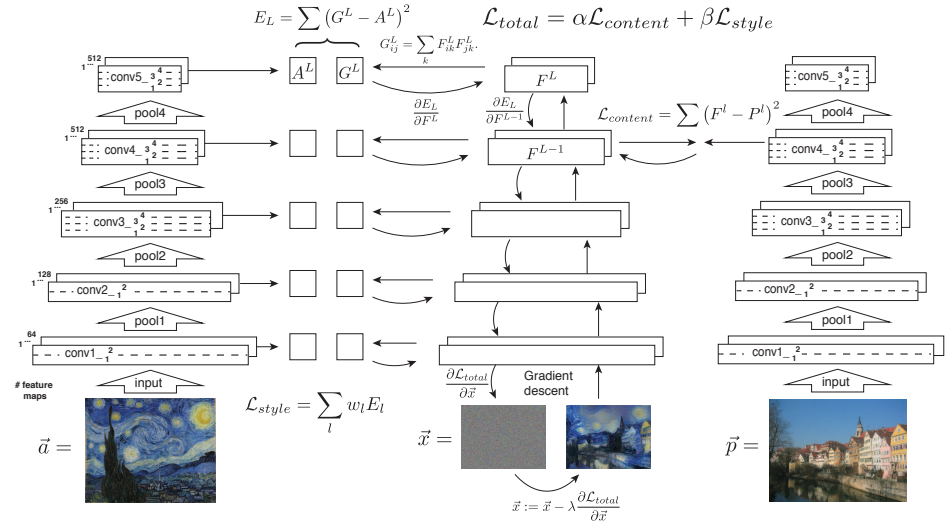


Image Reconstruction

Other use cases where image reconstruction is required:

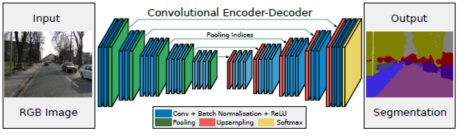
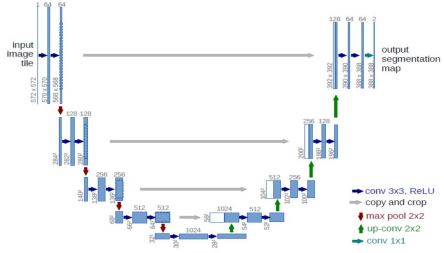


Fig. 2. An illustration of the SegNet architecture. There are no fully connected layers and hence it is only convolutional. A decoder upsamples its input using the transferred pool indices from its encoder to produce a sparse feature map(s). It then performs convolution with a trainable filter bank to densify the feature map. The final decoder output feature maps are fed to a soft-max classifier for pixel-wise classification.

SegNet – (Badrinarayanan 2017)

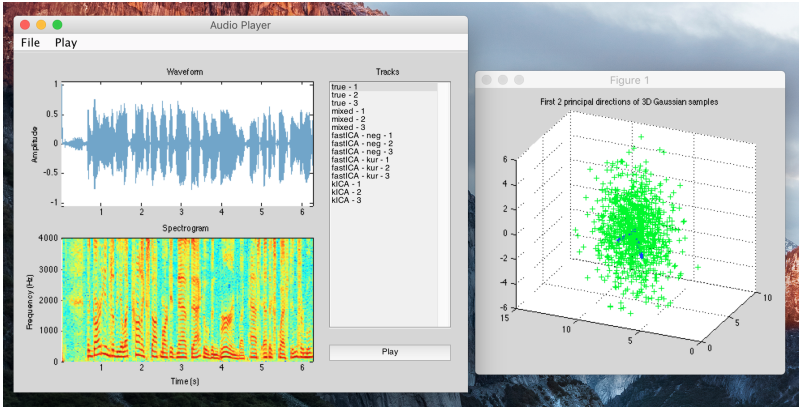


U-Net, (Ronneberger 2015)

CNN & Time-Frequency representation

The example of source separation (that makes great progress over the last 5 years)

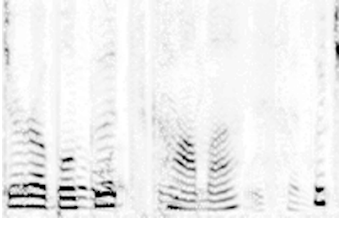
Original problem: ICA (independant componant analysis)
SVD algorithm (unsupervised) in time or time frequency domain:

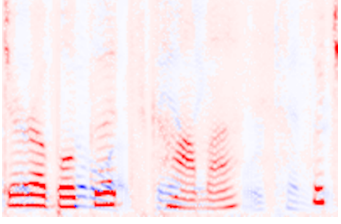


CNN & Time-Frequency representation

The example of source separation (that makes great progress over the last 5 years)

New Problem:
A supervised classification problem in the time frequency domain

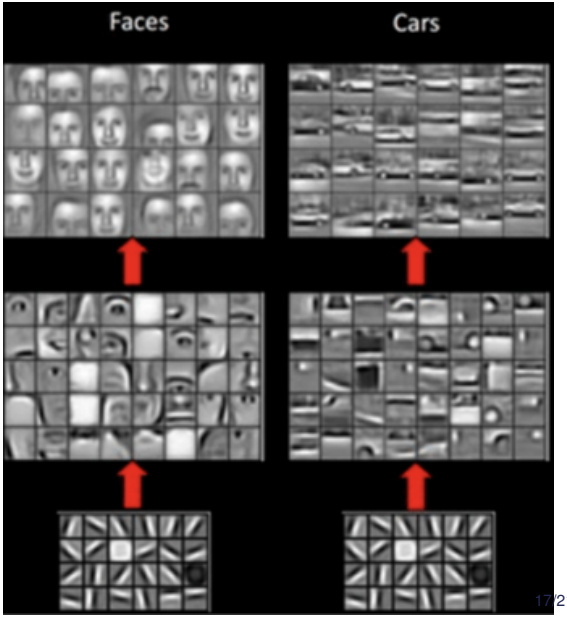




Introspection

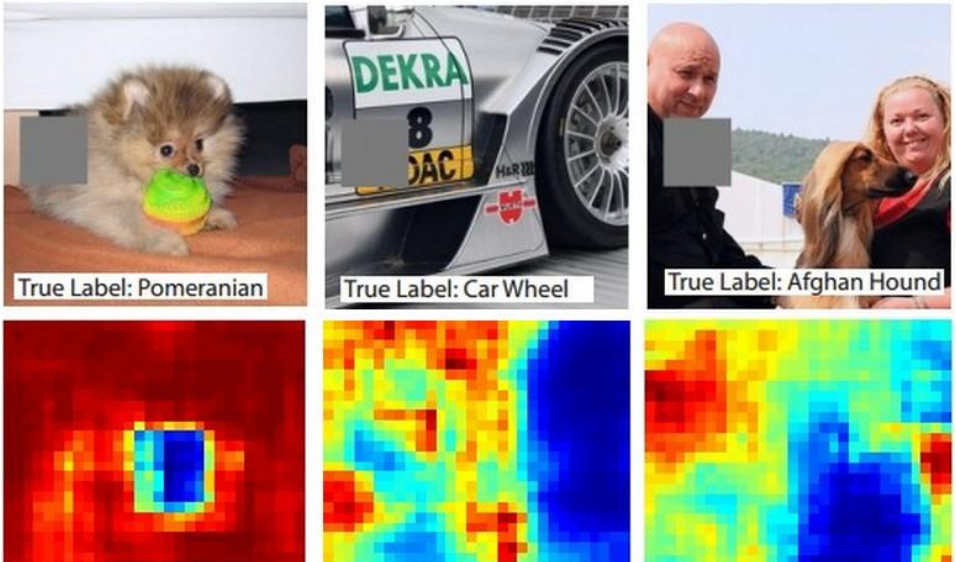
Introspection d'un CNN

- Comprendre la classification : quelle région de l'image active la classification, quels filtres sont les plus importants . . .
- Les filtres sont de plus en plus abstraits, produisant des mots élémentaires visuels
- Les couches conservent les informations topologiques

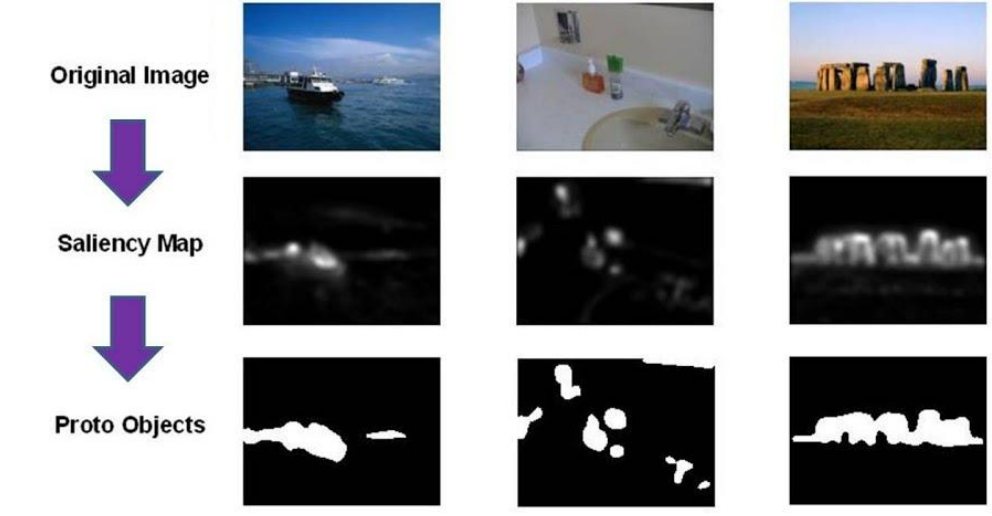


Occlusion

Sensibilité de la classe détectée aux occlusion:

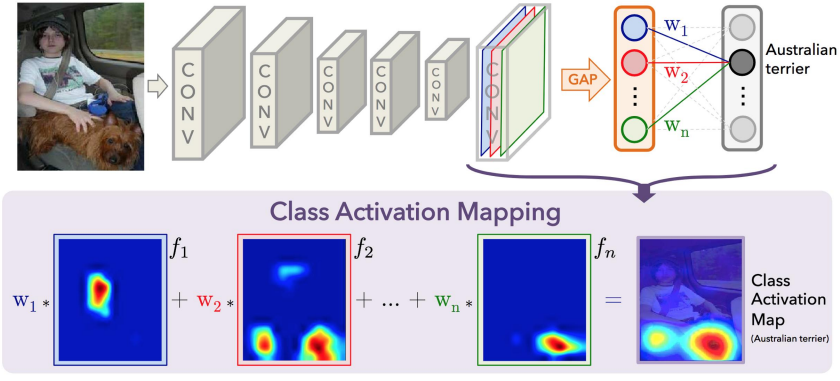


Carte de saillance



Elles sont calculées en prenant le gradient de la sortie par rapport à l'image d'entrée. Elles permettent de mettre en avant les pixels auxquels la

Class Activation Map

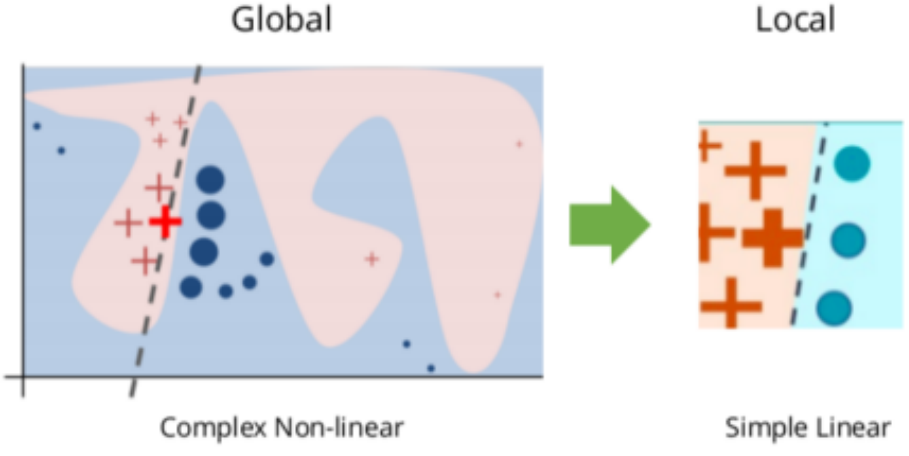


Class Activation Maps :

- Global Average Pooling (1 filtre = 1 feature) + pondération
- Combinaison linéaire
- Agrégation pondérée des filtres pour indiquer les régions d'intérêts.

Local Interpretable Model-Agnostic Expl. (LIME)

- Local linear model
- Weight interpretation



Local Interpretable Model-Agnostic Expl. (LIME)