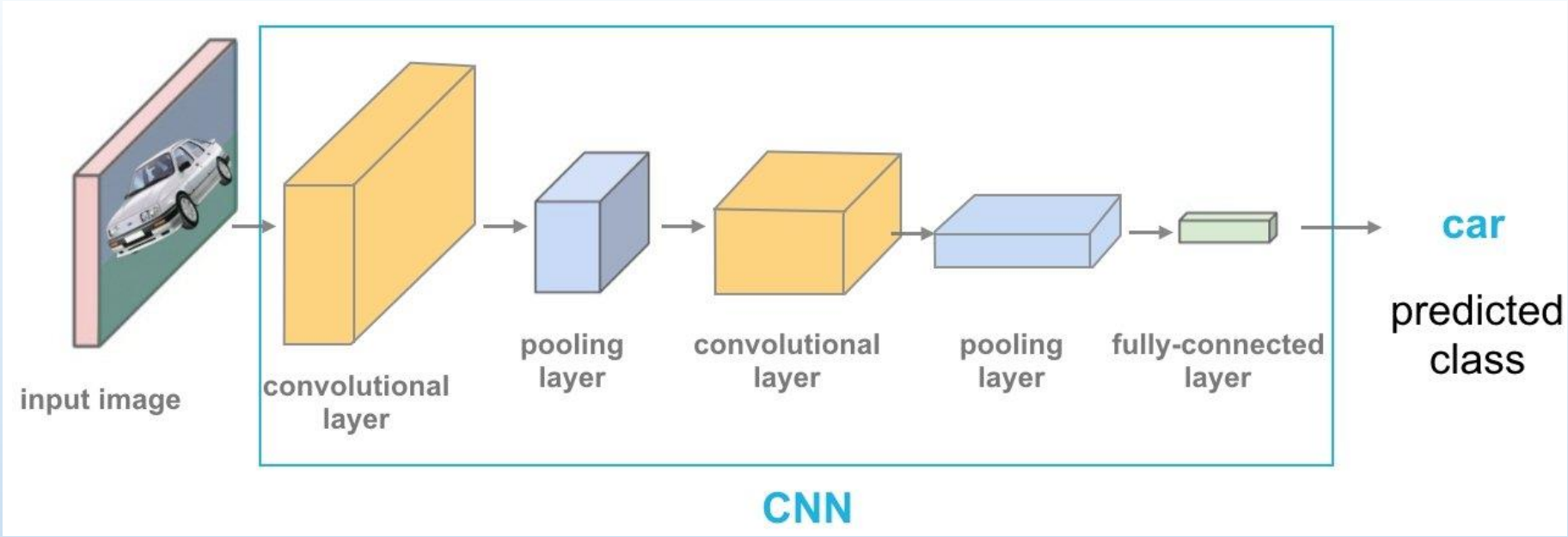
IA et Nous
Vision informatique
Détection d'objets

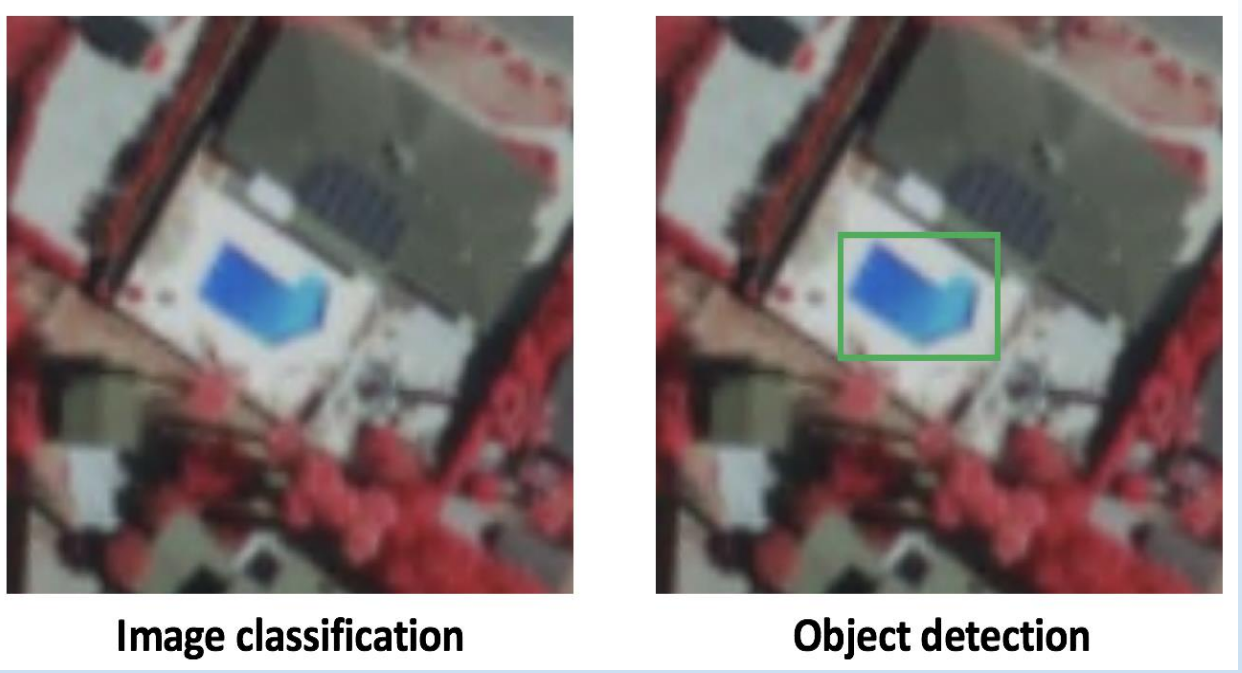
Rappel sur les CNN



La combinaison des couches d'un CNN permet au réseau de neurones conçu d'apprendre à identifier et à reconnaître l'objet d'intérêt dans une image.

Les réseaux de neurones convolutifs simples sont conçus pour la ***classification d'images*** *et la détection d'objets avec un* ***seul objet dans l'image.***

Classification et détection d'objets



La sortie d'un modèle de détection d'objet doit inclure :

- *Probabilité qu'il y ait un objet,*
- *Hauteur de la boîte englobante,*
- *Largeur de la boîte englobante,*
- *Coordonnée horizontale du point central de la boîte englobante,*
- *Coordonnée verticale du point central de la boîte englobante.*

Classification et détection d'objets

Les applications pour les détecteurs d'objets incluent :

- *Conduite autonome*
- *Systèmes de surveillance intelligents*
- *La reconnaissance faciale*
- *Production industrielle*
- *Santé*
- *Agriculture….*

Classification et détection d'objets

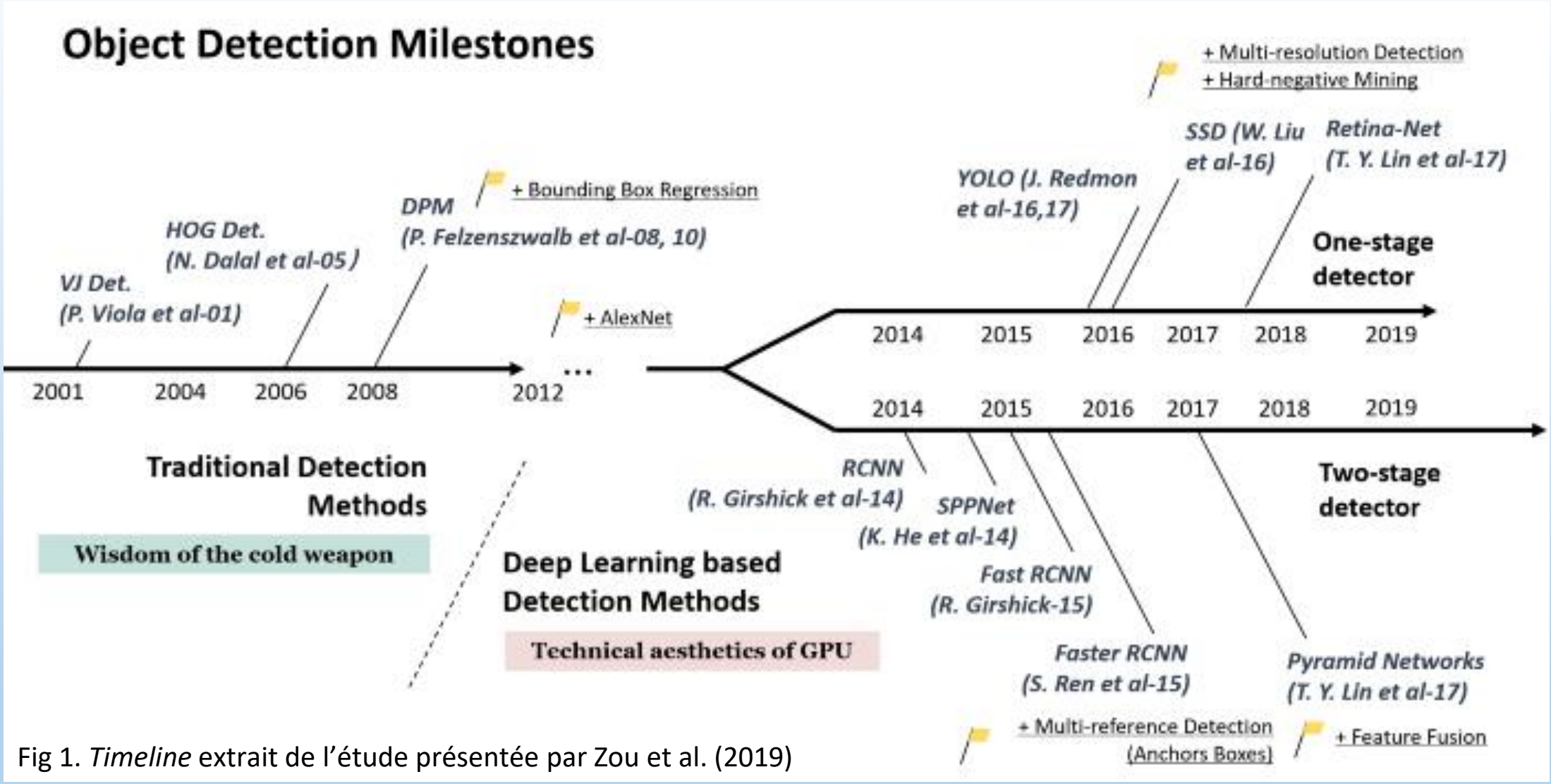
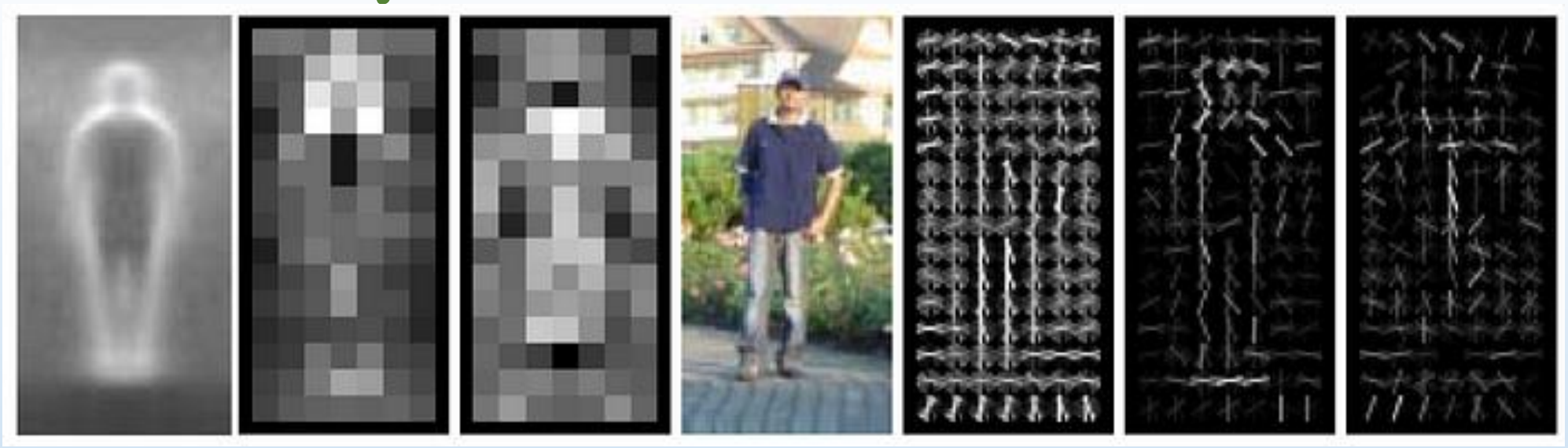


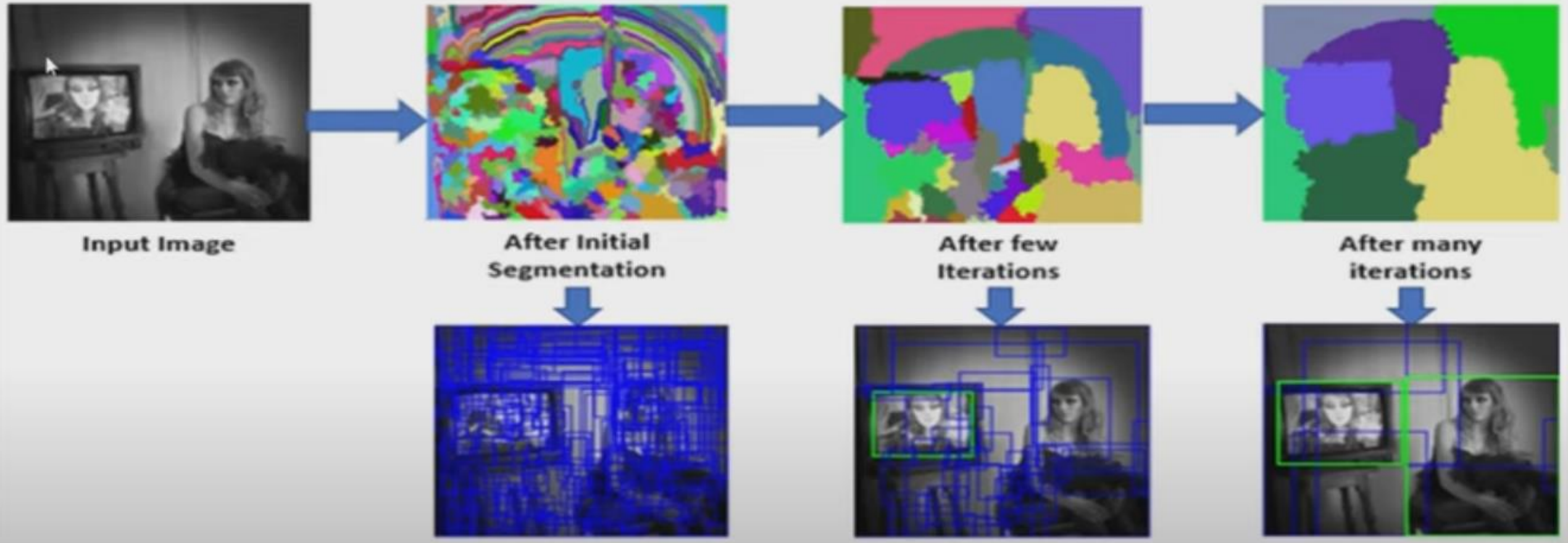
Fig 1. *Timeline* extrait de l'étude présentée par Zou et al. (2019)

Exemple ancienne méthode



HOG Histogramme des gradients orientés 2005

Principe


Input Image
After Initial Segmentation
After few Iterations
After many iterations

Edge Boxes

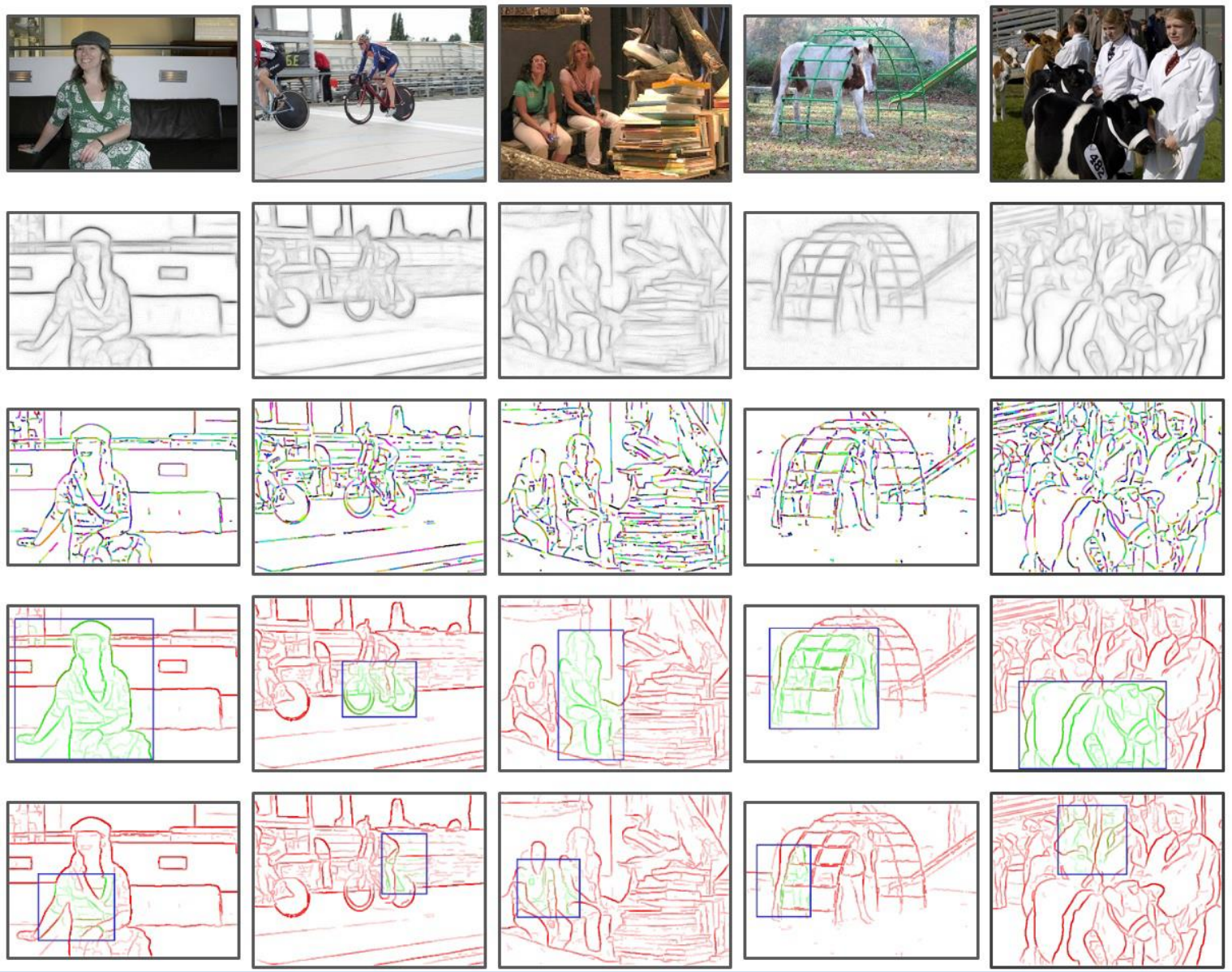
Image initiale

Contours structurés

Groupes de contours

Exemples de boites correctes
Vert = appartient à l'objet

Boites incorrectes



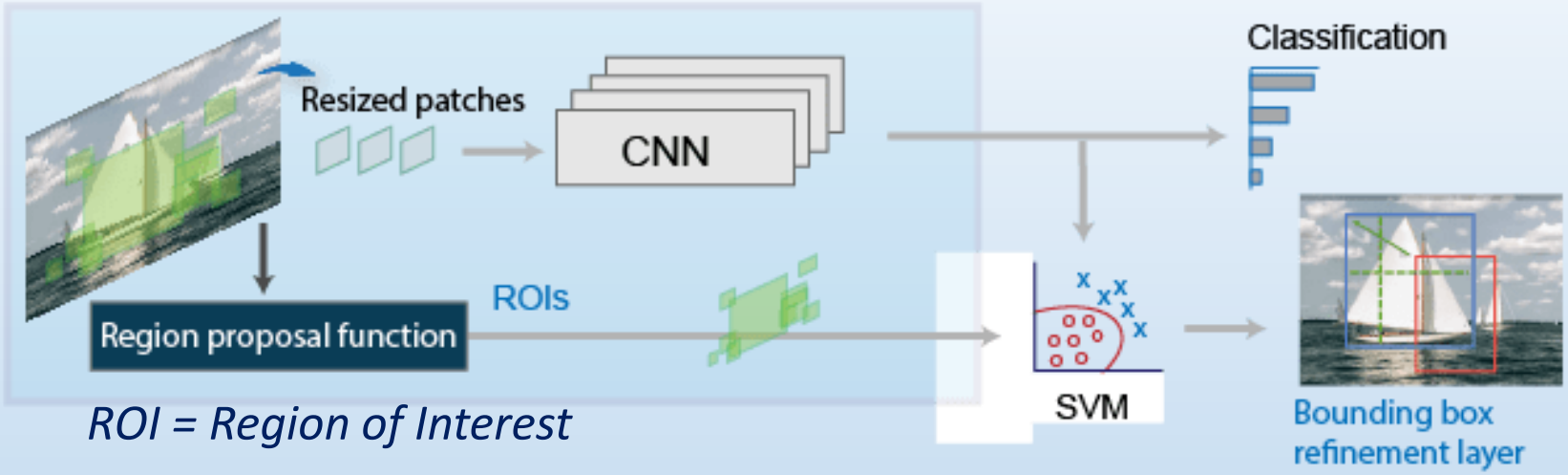
Classification et détection d'objets

Principe : deux approches d'apprentissage en profondeur

1. *Approche en deux étapes R-CNN, Fast et Faster R-CNN.*
   - *Identification des régions dans une image pouvant contenir un objet.*
   - *Détection et Classification de chaque objet uniquement dans ces régions .*
2. *Approche en une étape YOLO, SSD …*
   - *Le réseau est capable de trouver tous les objets d'une image* ***en une seule passe*** *(d'où 'single-shot' ou 'look once') à travers le convnet*

Algorithmes en deux étapes

R-CNN
Regional Convolution Network



ROI = Region of Interest

R-CNN génère des propositions de régions en utilisant un algorithme tel que Edge Boxes.

Au départ 2000 ROI sont proposées.

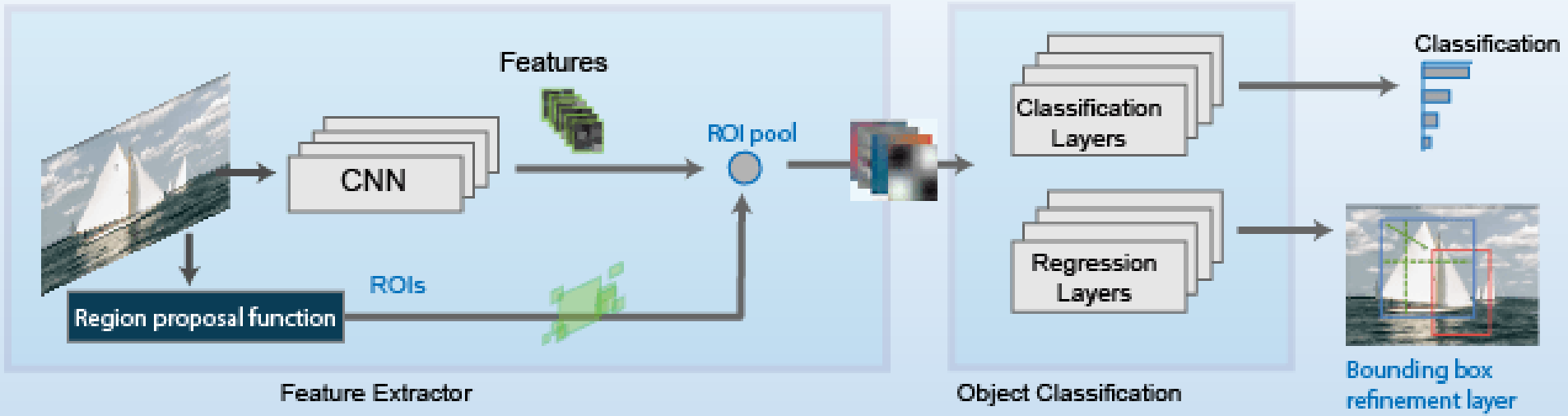
Les régions proposées sont coupées de l'image et redimensionnées.

Le CNN classe les régions recadrées et redimensionnées.

Les boîtes englobantes de proposition de région sont affinées par une machine à vecteurs de support (SVM)

Fast R-CNN

Fast Regional Convolution Network

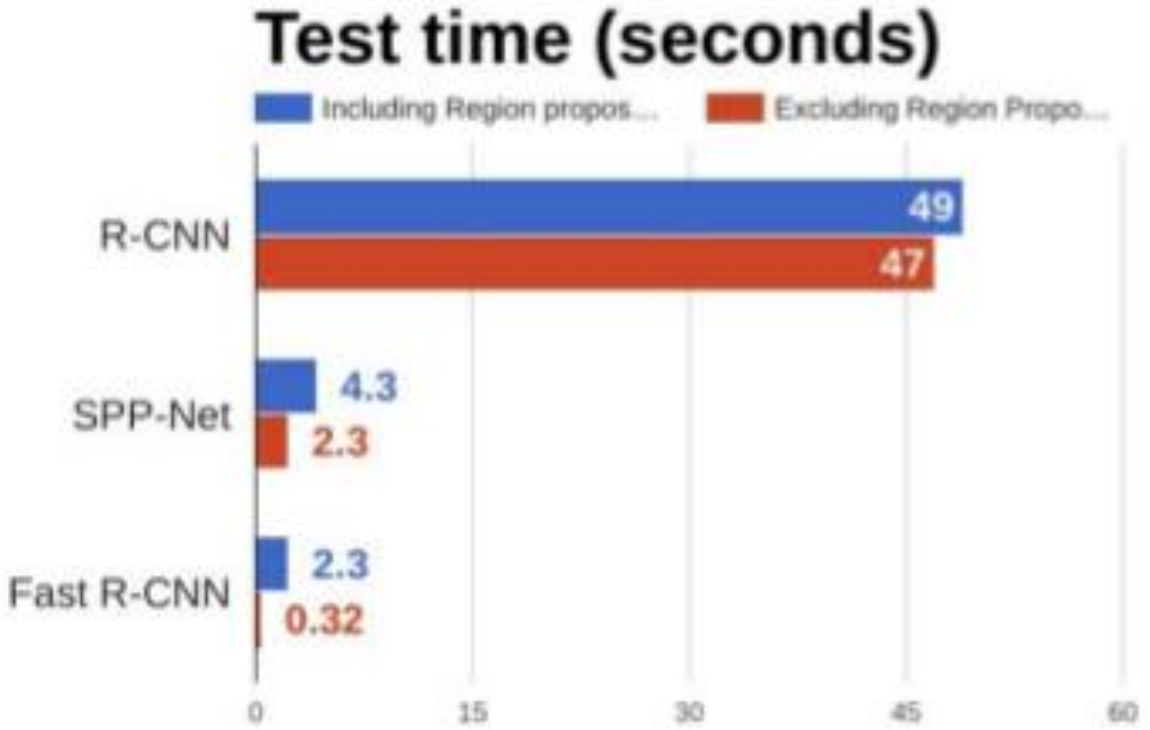


Fast R-CNN utilise aussi un algorithme comme Edge Boxes pour générer des propositions de régions.

Fast R-CNN traite l'intégralité de l'image. Alors qu'un détecteur R-CNN doit classer chaque région, Fast R-CNN regroupe les caractéristiques CNN correspondant à chaque proposition de région.

Fast R-CNN est plus efficace que R-CNN, car il n'est plus nécessaire de fournir les calculs pour les 2000 ROI

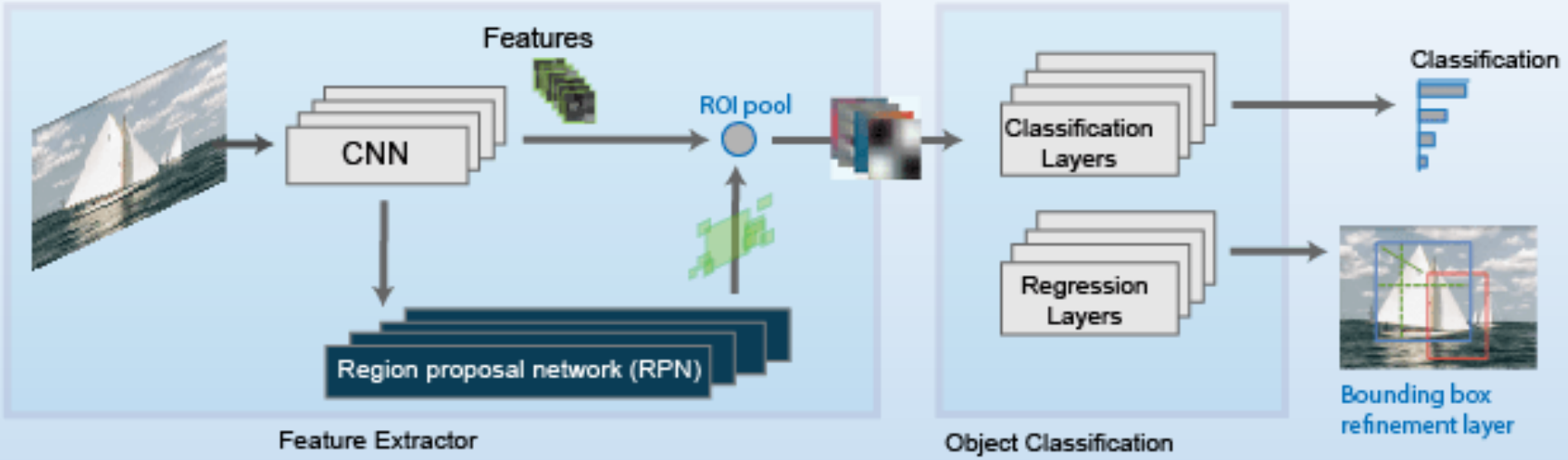
Perfos R-CNN et Fast R-CNN



Comparaison des algorithmes de détection d'objets

Faster R-CNN

Faster Regional Convolution Network

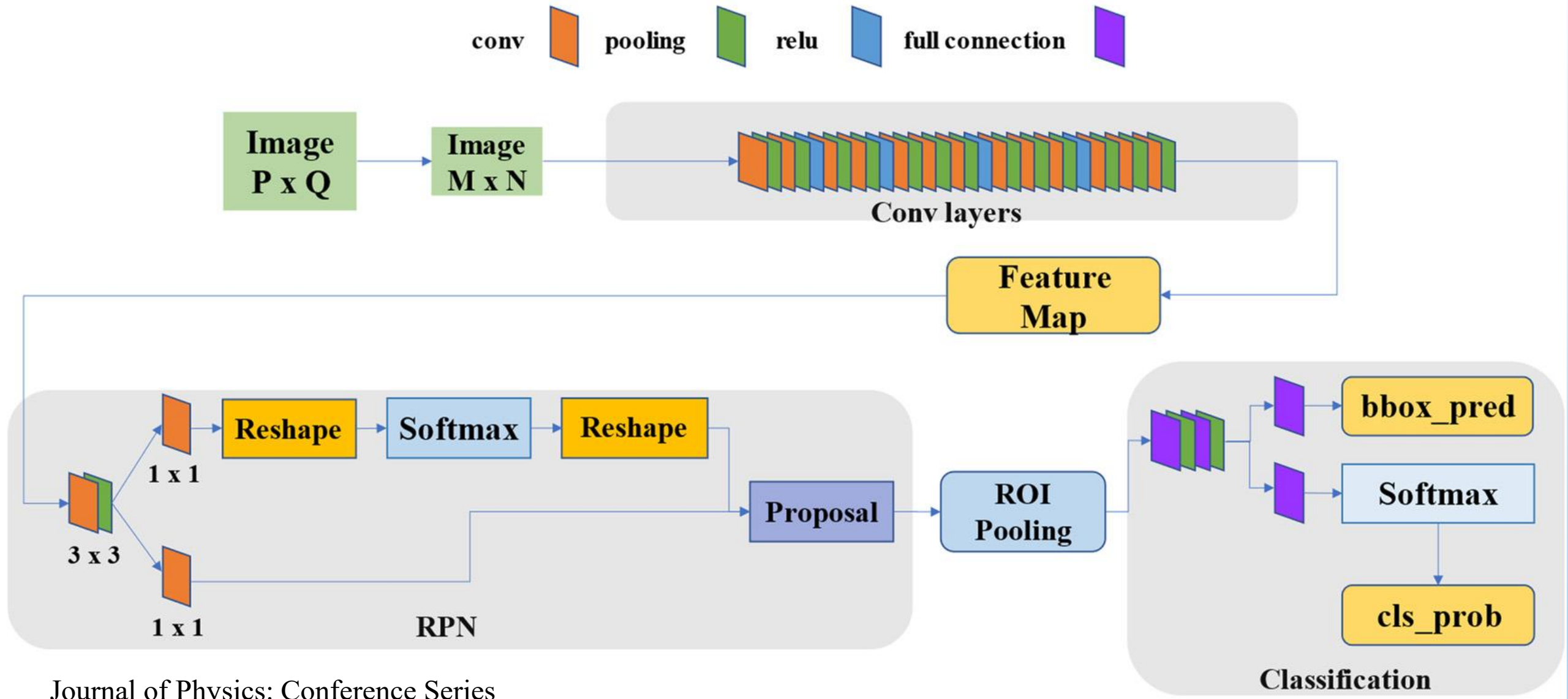


*Faster R-CNN ajoute un **réseau de proposition de région (RPN)** qui est simplement un **réseau de neurones** qui génère des propositions de région directement*

Le RPN utilise des boîtes d'ancrage pour la détection d'objets .

Un détecteur d'objets qui utilise des boîtes d'ancrage peut traiter une image entière à la fois, ce qui est plus rapide et rend possible la détection d'objets en temps réel.

Faster R-CNN

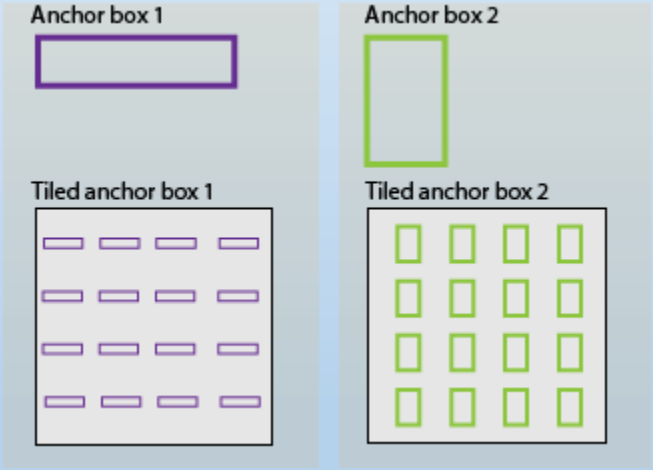


Journal of Physics: Conference Series

Faster R-CNN

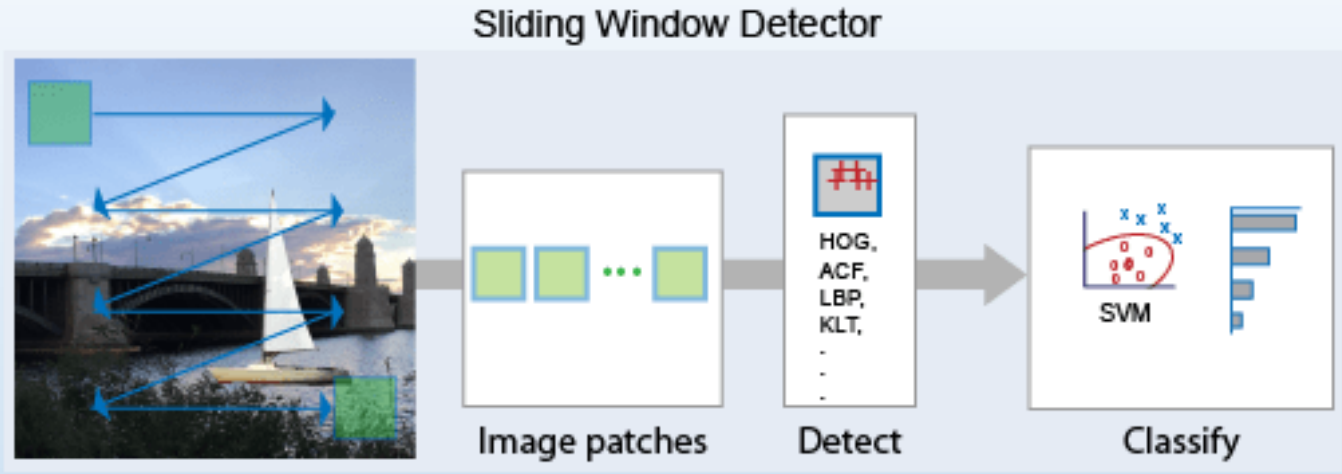
Boîtes d'ancrage Anchor Boxes

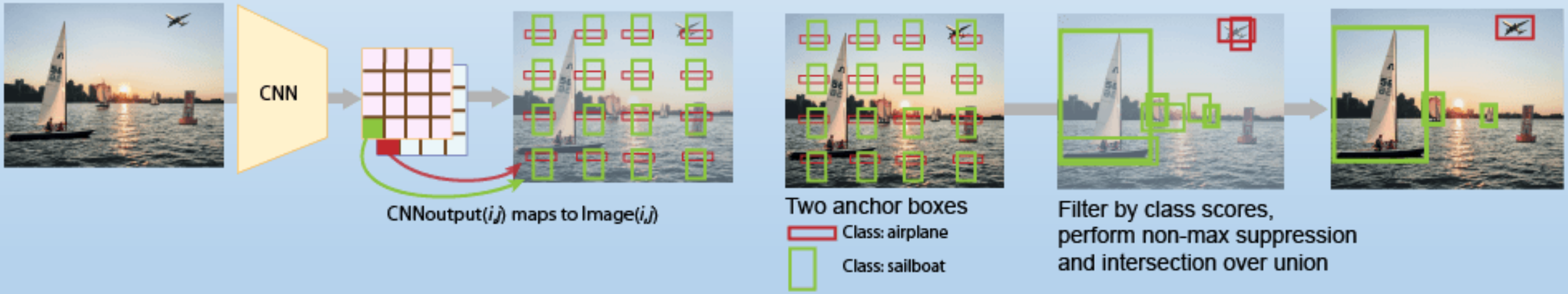
Ensemble de boîtes englobantes prédéfinies d'une certaine hauteur et largeur.
Ces cases sont définies pour capturer l'échelle et le rapport d'aspect des classes d'objets spécifiques à détecter
Elles sont choisies en fonction de la taille des objets dans les ensembles de données d'apprentissage.



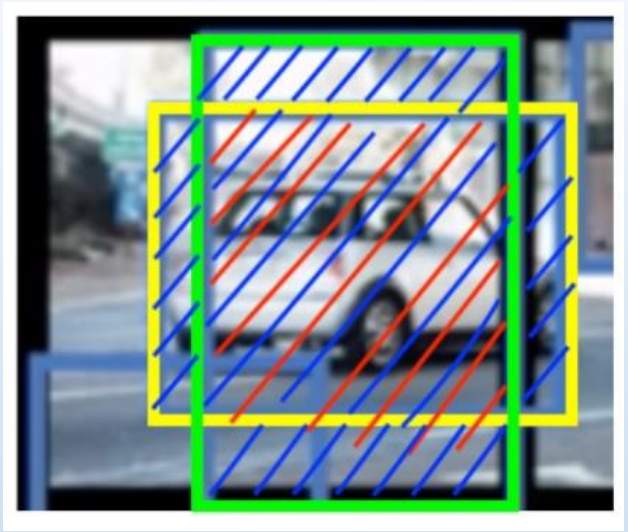
Faster R-CNN

Boîtes d'ancrage Anchor Boxes





Non Max Suppression
Intersection over Union



1) On supprime les anchor boxes ayant une probabilité inférieure à un certain seuil, 0.6 par exemple;

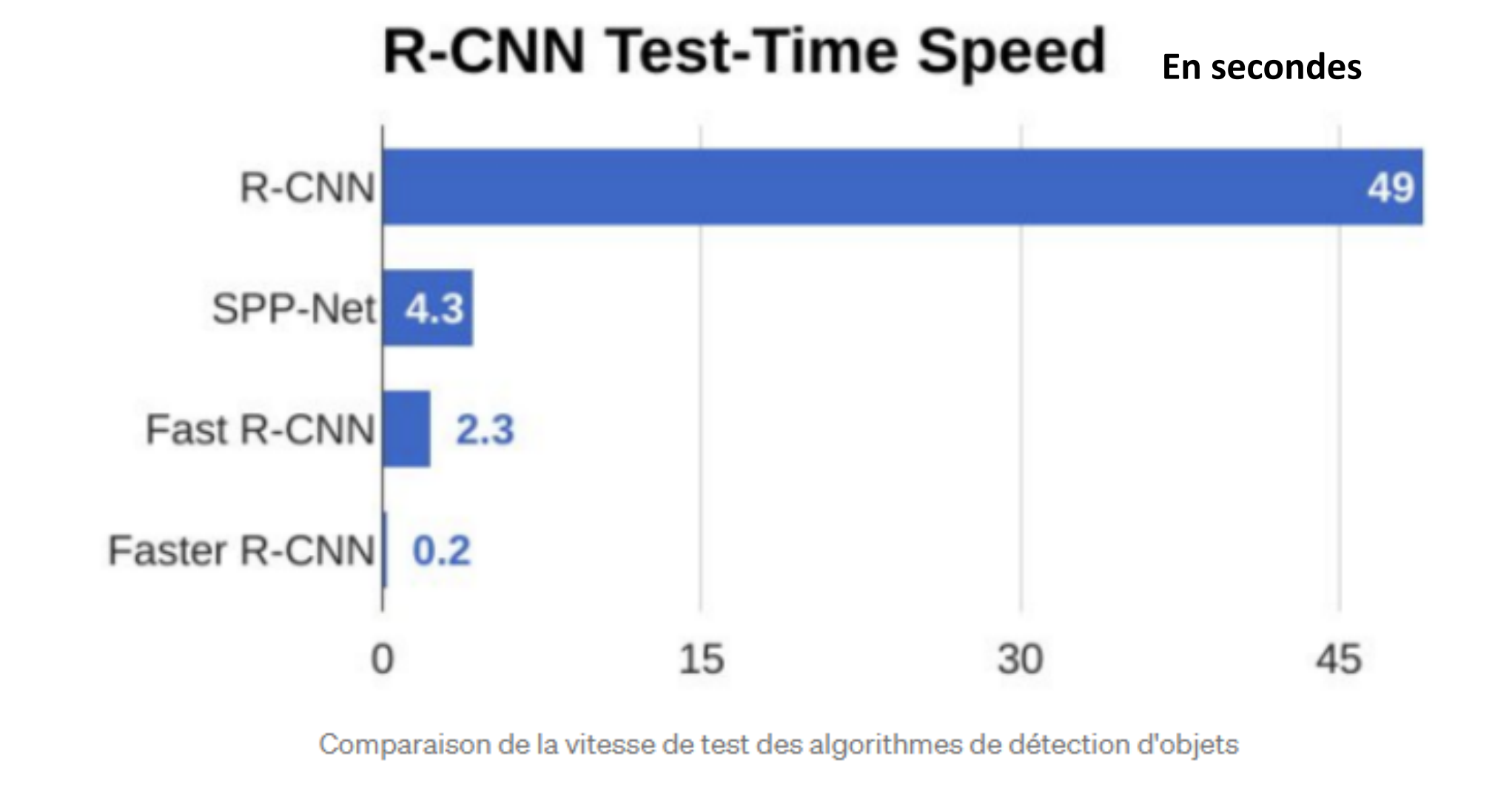
2) on sélectionne l'anchor box avec la probabilité de détection la plus élevée;

3) on supprime les anchor boxes qui intersectent l'anchor box sélectionnée en 2) ayant un IoU supérieur à un certain seuil, 0.5 généralement.

En clair, on supprime les anchor boxes trop proches les unes des autres dans cette étape car elles labelisent le même objet

on répète 2) et 3).

Perfos R-CNN, Fast et Faster



Comparaison de la vitesse de test des algorithmes de détection d'objets

Algorithmes en une étape

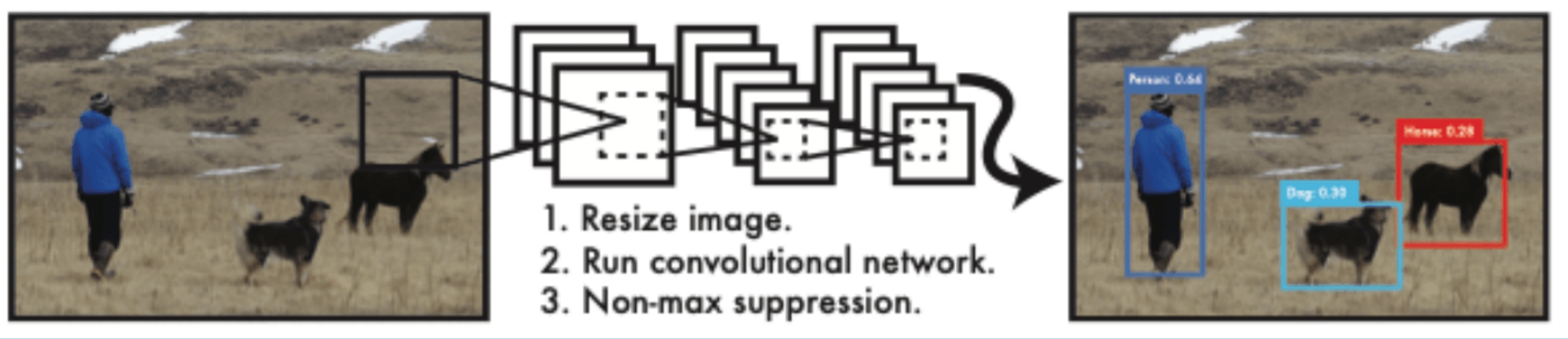
YOLO : You Only Look Once
SSD : Single Shot Detector

YOLO : You Only Look Once

YOLOv3 (You Only Look Once, Version 3)

Détecte et identifie des objets en temps réel dans des vidéos, des flux en direct ou des images.

YOLO utilise des fonctionnalités apprises par un réseau de neurones à convolution profonde



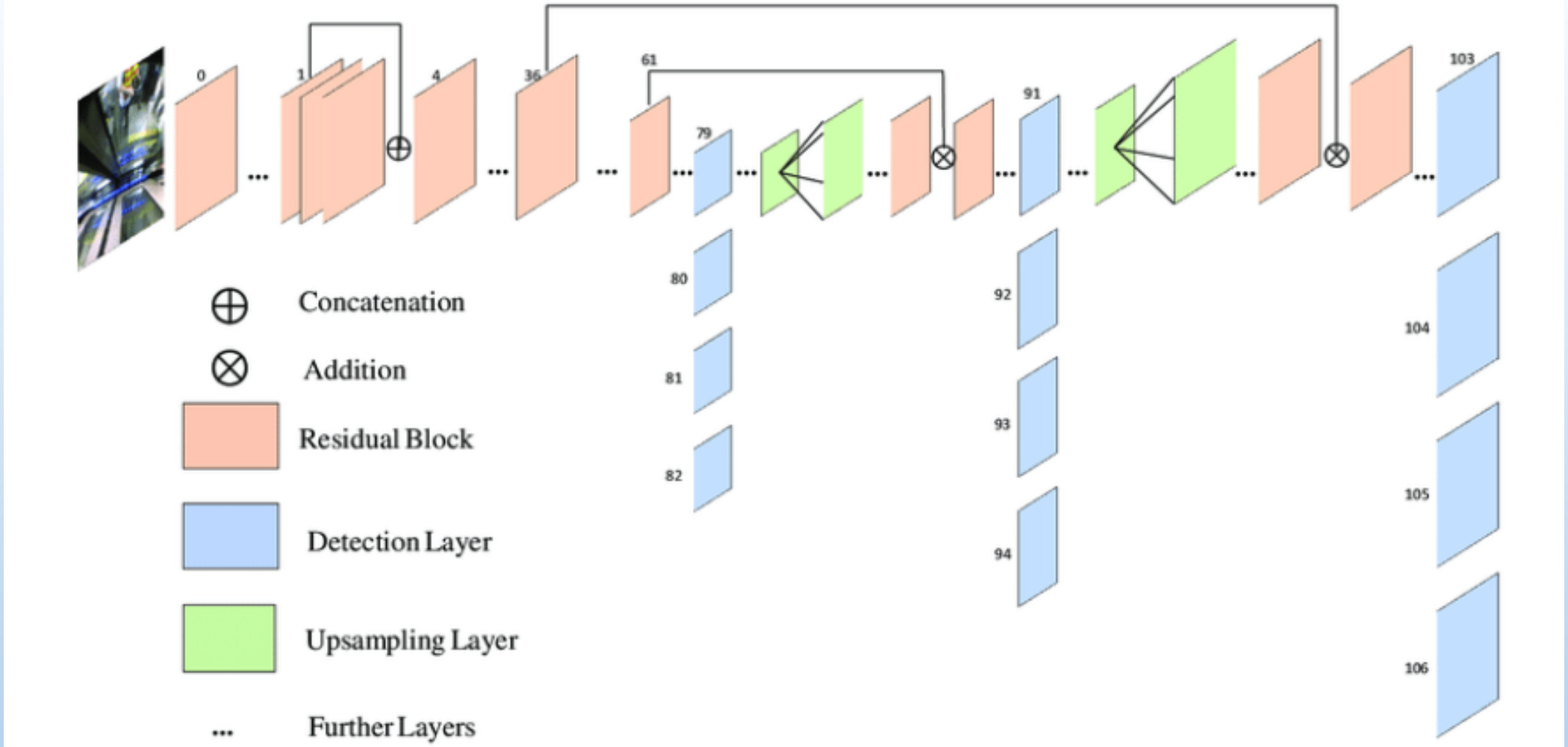
YOLO : You Only Look Once

Pour les détecteurs d'objets classiques, les caractéristiques apprises par les couches convolutives sont transmises à un classificateur qui effectue la prédiction de détection.

Dans YOLO, la prédiction est basée sur une couche convolutive qui utilise des convolutions 1 × 1.

YOLO est nommé ***"vous ne regardez qu'une seule fois"*** *car sa prédiction utilise des convolutions 1 × 1 ; la taille de la carte de prédiction correspond exactement à la taille de la carte d'entités qui la précède.*

YOLO : You Only Look Once



Architecture réseau de YOLOv3.

YOLO

Entrée : Images (n, 416, 416, 3)

si images de taille différente, redimensionnement

Détection d'objets réalisée en 3 échelles aux couches 82, 94, 106

On utilise 3 strides différents

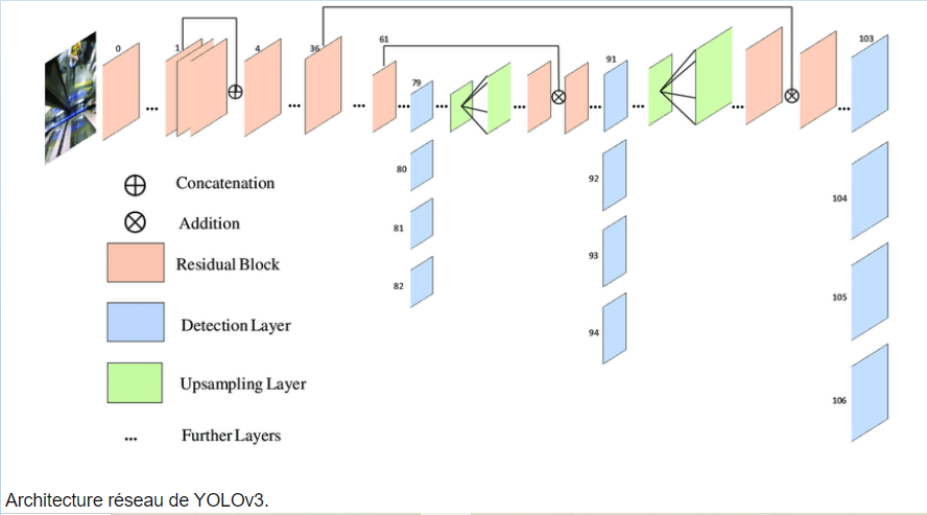
stride 32 ➔ grille de 13X13 grands objets

stride 16 ➔ grille de 26X26 objets moyens

stride 8 ➔ grille de 52X52 petits objets

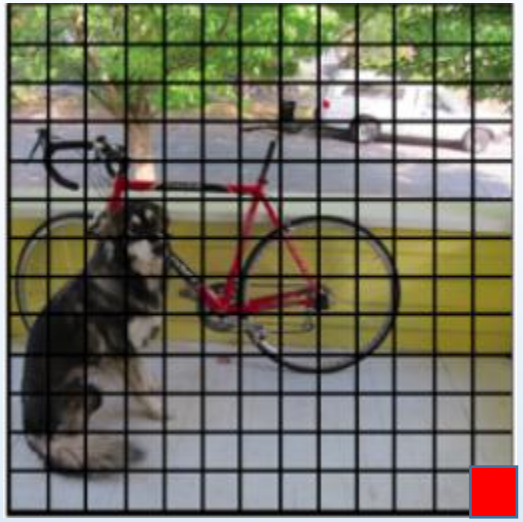
Détection objets

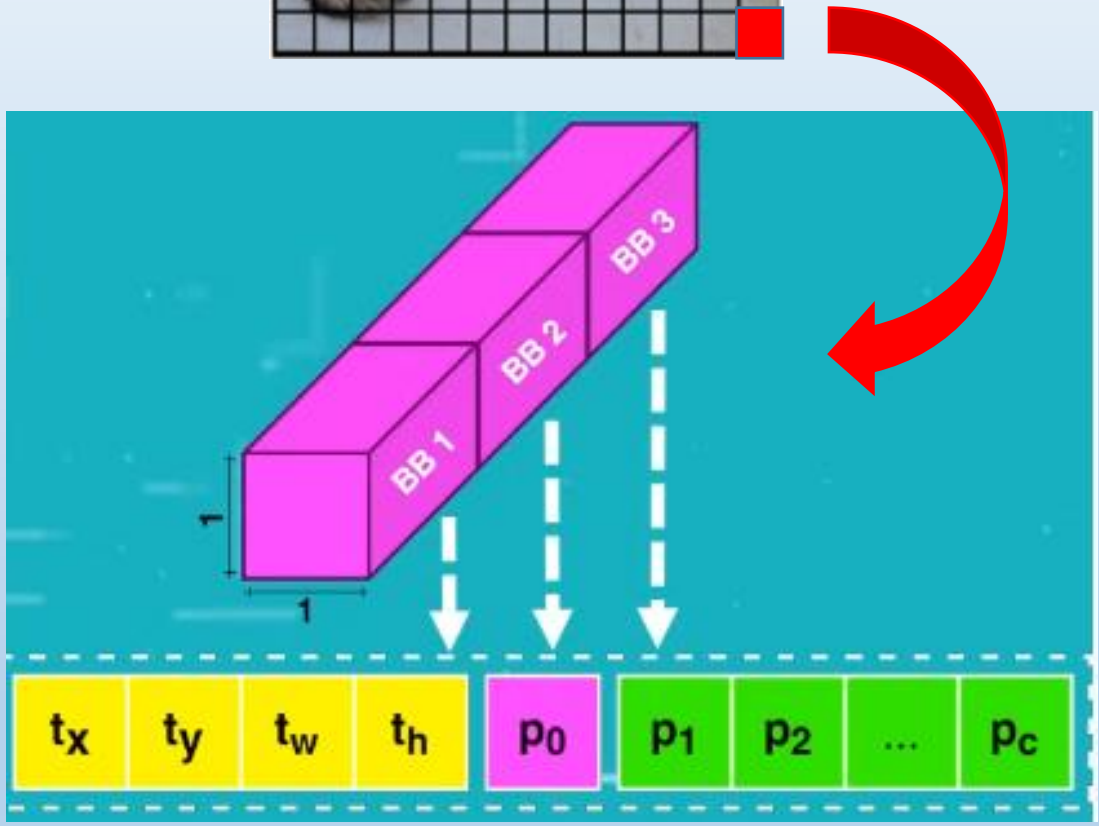
Noyau de 1X1 appliqué à chaque échelle



Architecture réseau de YOLOv3.

YOLO





Noyaux de détection 1X1

Produisent 3 Bounding Boxes/cellule

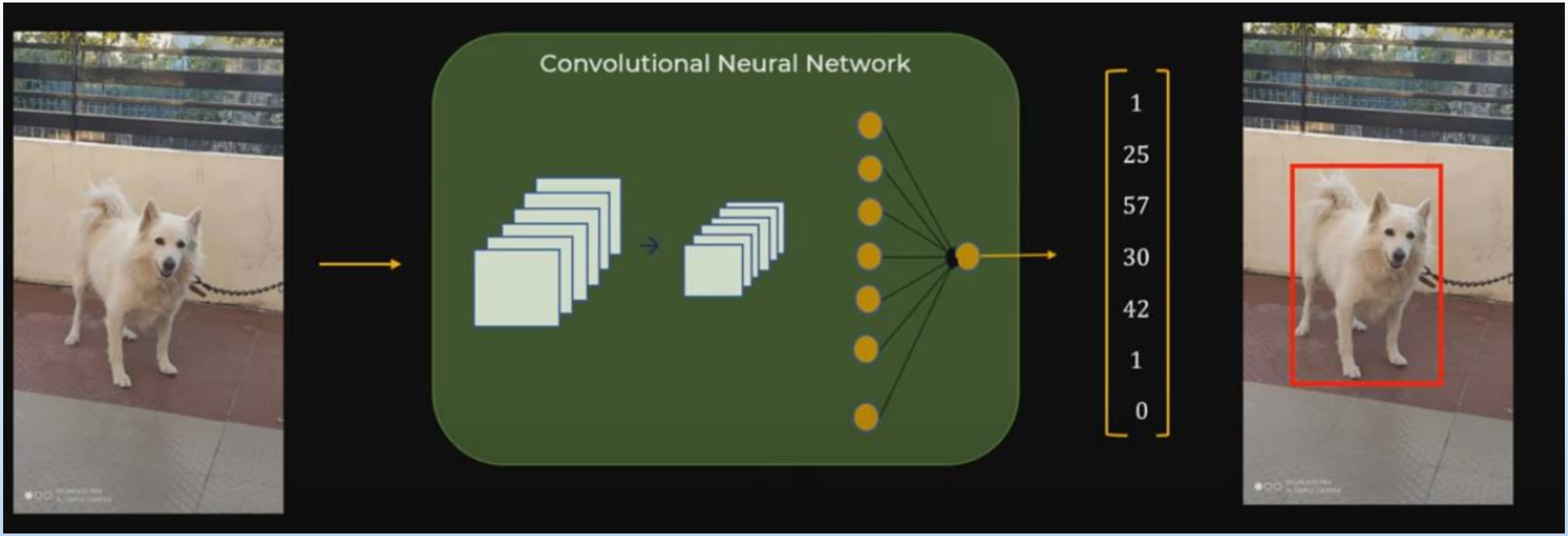
Soit 13x13x3 BB pour la grille

5 + C paramètres :

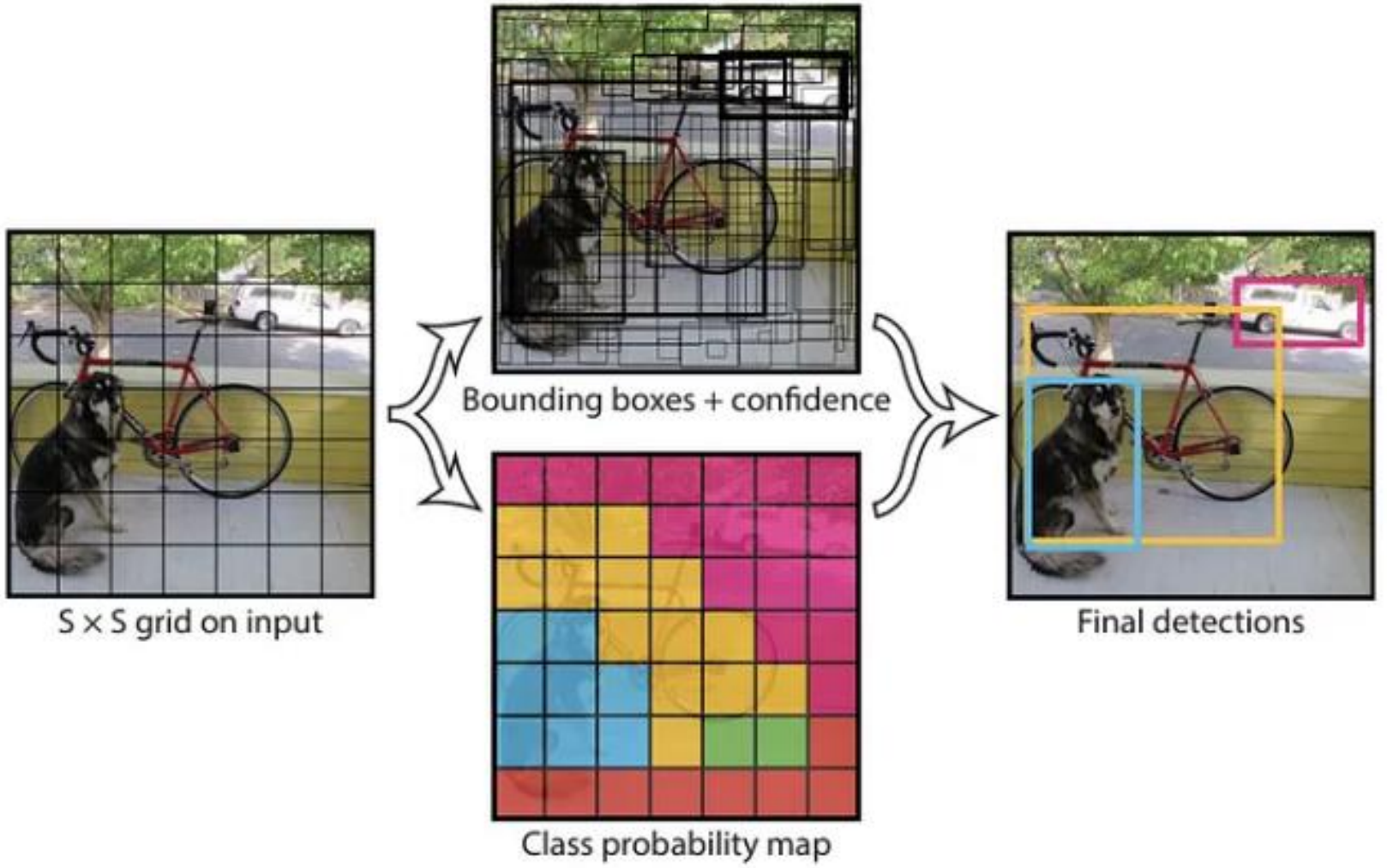
- *Coordonnées*
- *Largeur/Hauteur*
- *P0 Score détection objet*
- *C Probabilités classes*

Exemple :
si c=80 ➔ 3x85 attributs/cellule

YOLO

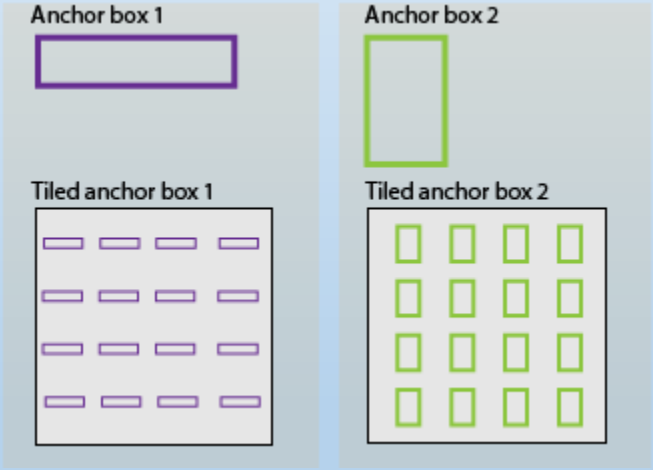


YOLO

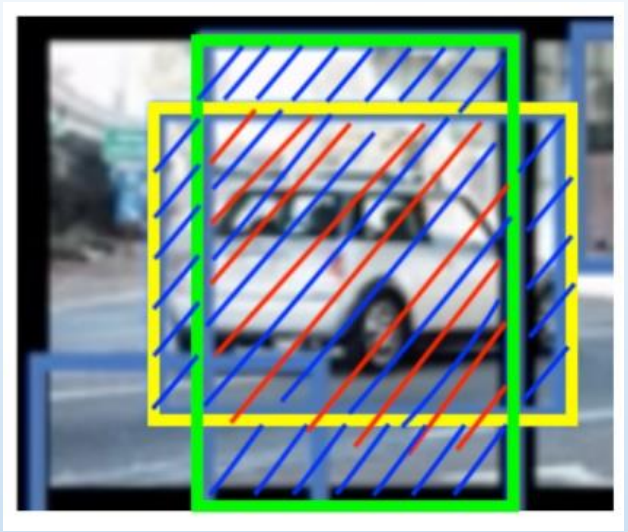

Bounding boxes + confidence
S × S grid on input
Class probability map
Final detections

Boîtes d'ancrage Anchor Boxes

Ensemble de boîtes englobantes prédéfinies d'une certaine hauteur et largeur.
Ces cases sont définies pour capturer l'échelle et le rapport d'aspect des classes d'objets spécifiques à détecter
Elles sont choisies en fonction de la taille des objets dans les ensembles de données d'apprentissage.



Non Max Suppression
Intersection over Union



1) On supprime les anchor boxes ayant une probabilité inférieure à un certain seuil, 0.6 par exemple;

2) on sélectionne l'anchor box avec la probabilité de détection la plus élevée;

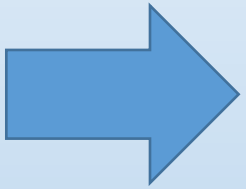
3) on supprime les anchor boxes qui intersectent l'anchor box sélectionnée en 2) ayant un IoU supérieur à un certain seuil, 0.5 généralement.

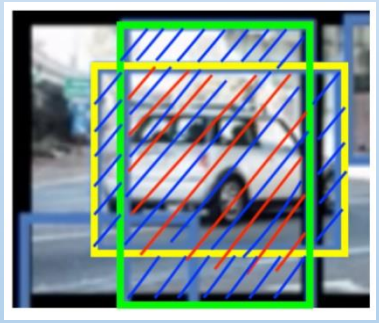
En clair, on supprime les anchor boxes trop proches les unes des autres dans cette étape car elles labelisent le même objet

on répète 2) et 3).

YOLO

Boites ancrage





Algorithme de suppression non max :

- ***Proba faible***
- ***Proche, IoU Intersection on Union***

YOLO

Image1

Image2

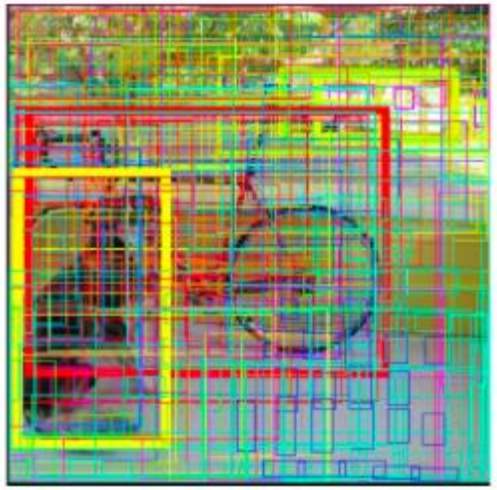


Image3

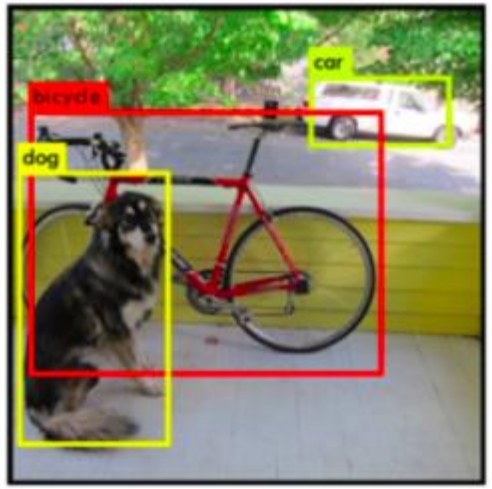
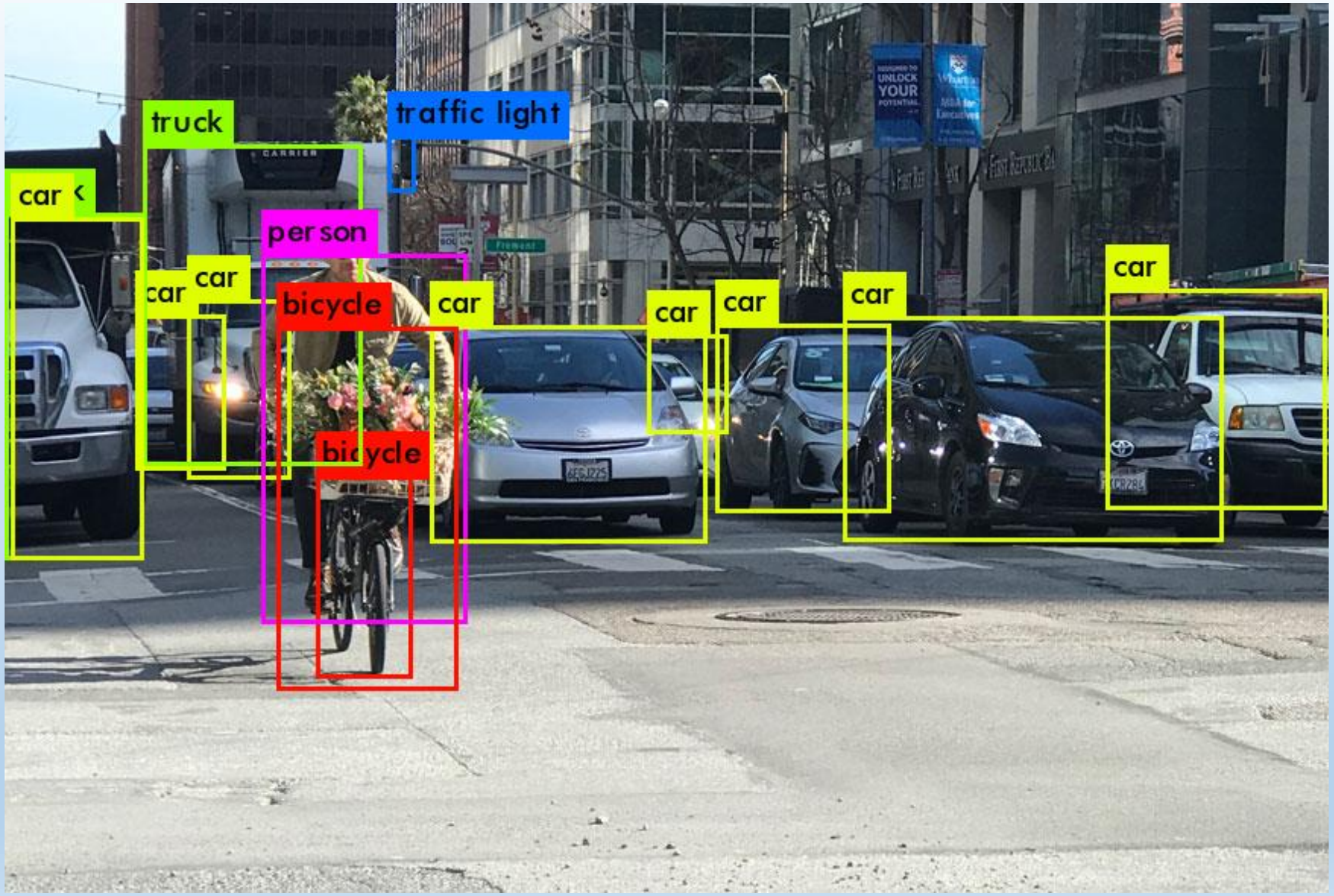


Image4

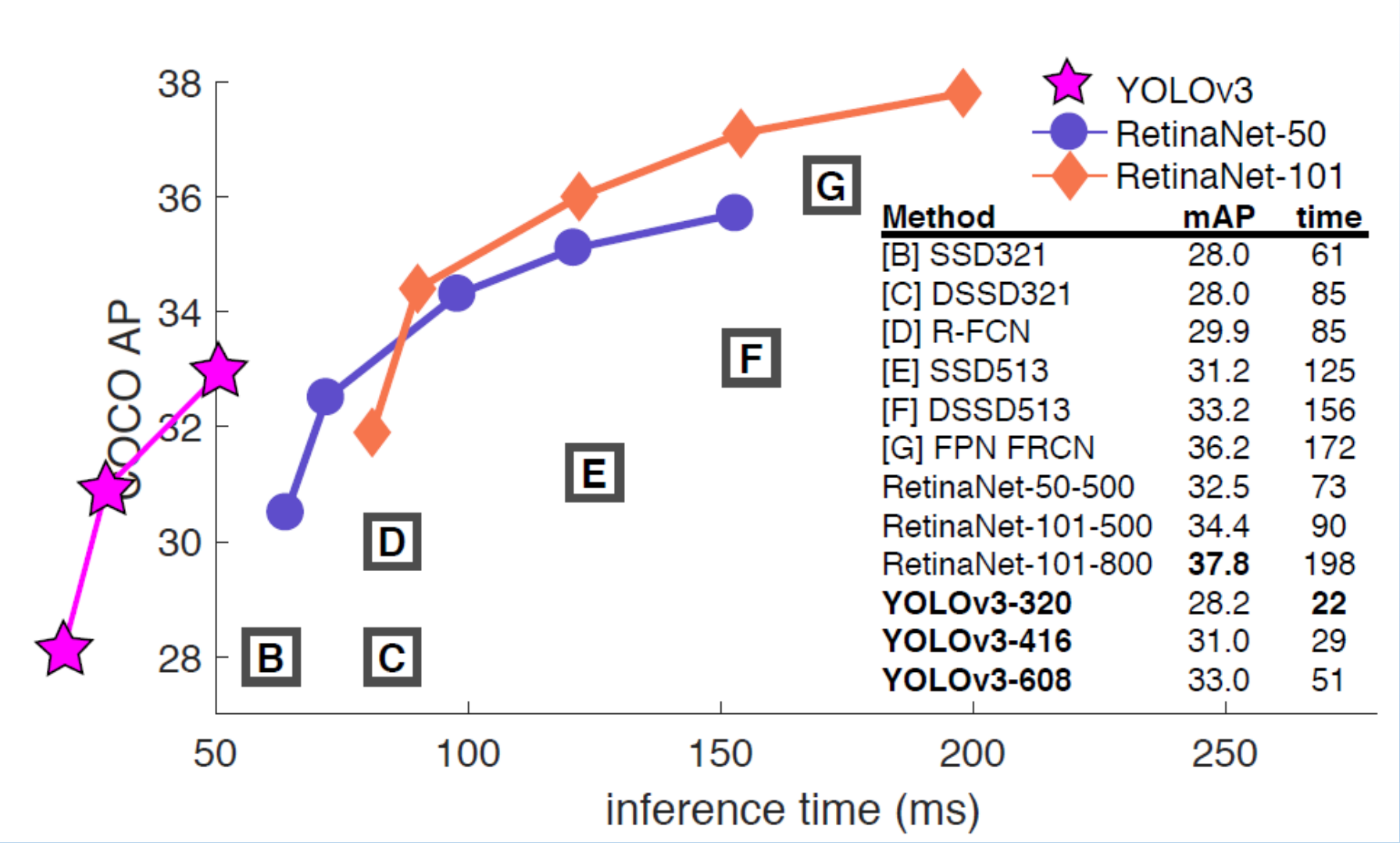
YOLO v3

- YOLO v3 106 couches
- Les 20 premières couches convolutives sont pré-entraînées les données de classification ImageNet 1000-class.

YOLO


truck
traffic light
car
person
car
car
bicycle
car
car
car
car
car
bicycle

Performances YOLO



Method	mAP	time
[B] SSD321	28.0	61
[C] DSSD321	28.0	85
[D] R-FCN	29.9	85
[E] SSD513	31.2	125
[F] DSSD513	33.2	156
[G] FPN FRCN	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	**37.8**	198
YOLOv3-320	28.2	**22**
YOLOv3-416	31.0	29
YOLOv3-608	33.0	51

YOLO

https://www.youtube.com/watch?v=MPU2HistivI

YOLO

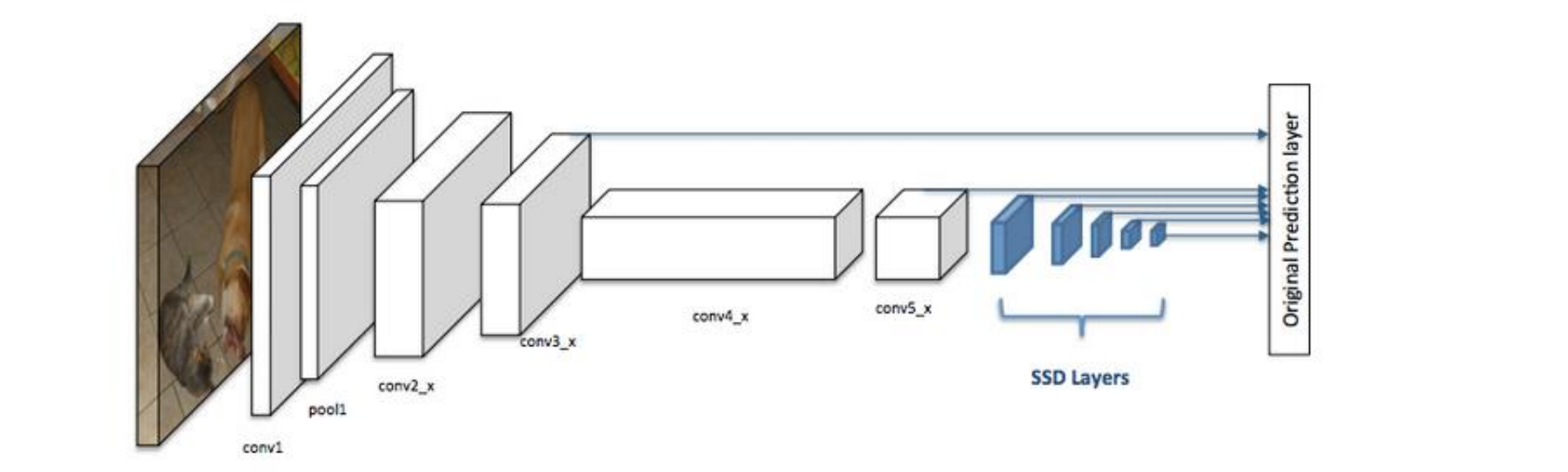
Source INRIA

https://pixees.fr/classcode-v2/

Ouvrir un compte et se connecter

Puis ouvrir **Class'code IAI et N° 3**

Détection d'objets
Single Shot Detector SSD



Les premières couches (boîtes blanches) sont la colonne vertébrale, les dernières couches (boîtes bleues) représentent la tête du SSD.

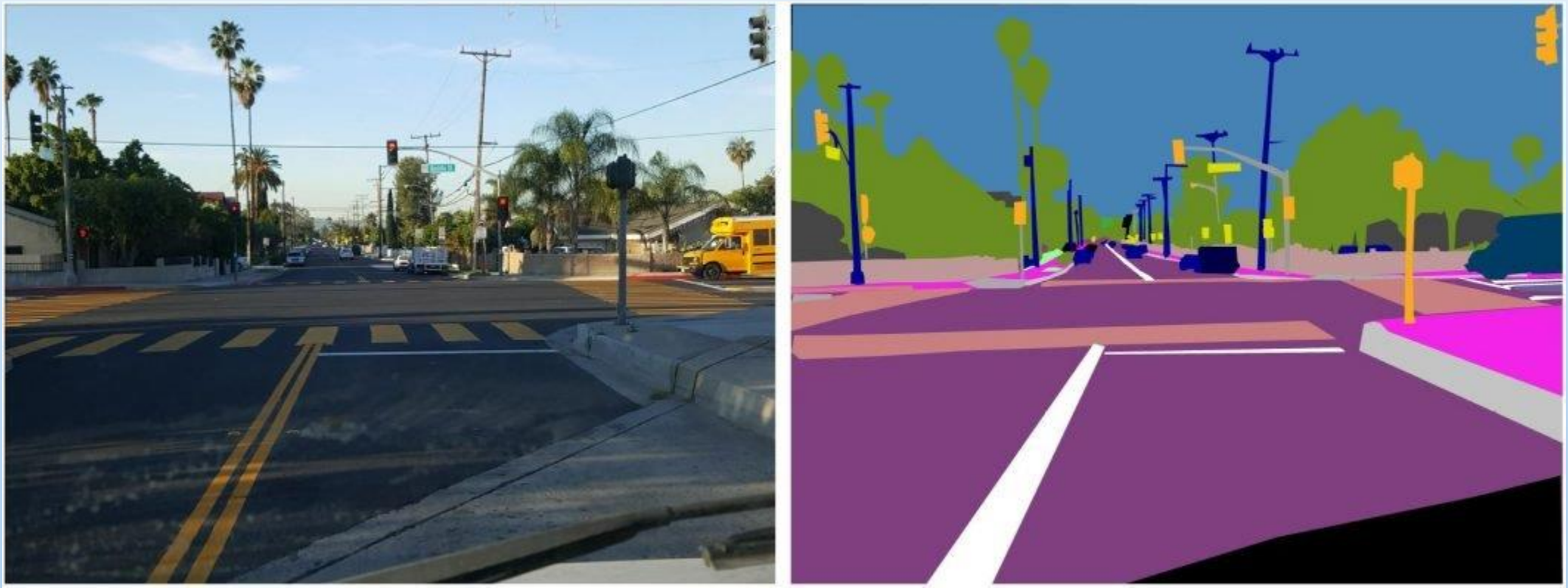
```
ssd = SingleShotDetector(data, grids=[4], zooms=[1.0], ratios=[[1.0, 1.0]])
```

Segmentation d'image

Segmentation d'image

La segmentation d'image consiste à regrouper des parties d'une image appartenant à la même classe d'objets.

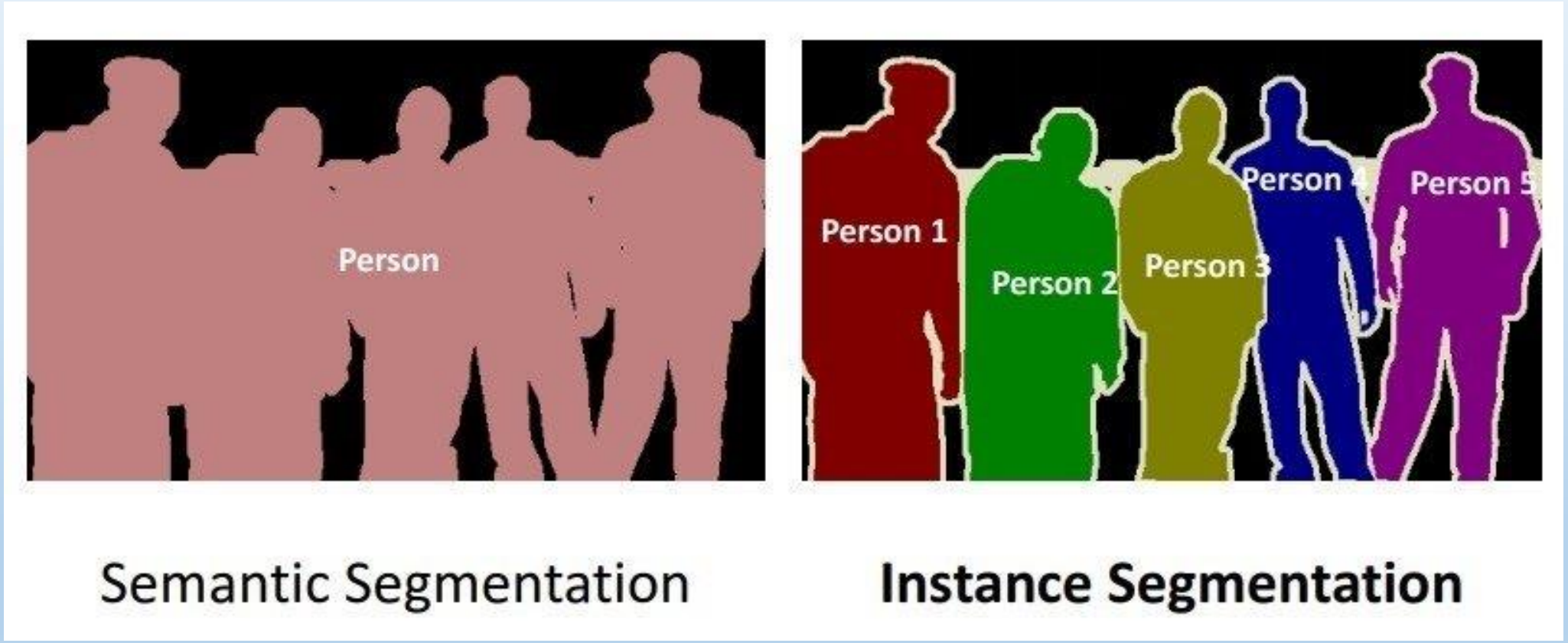
Ce processus est également appelé classification au niveau des pixels.



Segmentation d'image

Deux types :

1. *Segmentation sémantique*
2. *Segmentation des instances*



Segmentation d'image

Segmentation d'images et apprentissage en profondeur

Plusieurs algorithmes de segmentation d'images ont été développés.

Les méthodes antérieures incluent le seuillage, le regroupement basé sur des histogrammes, la croissance de régions, le regroupement de k-moyennes ou les bassins versants.

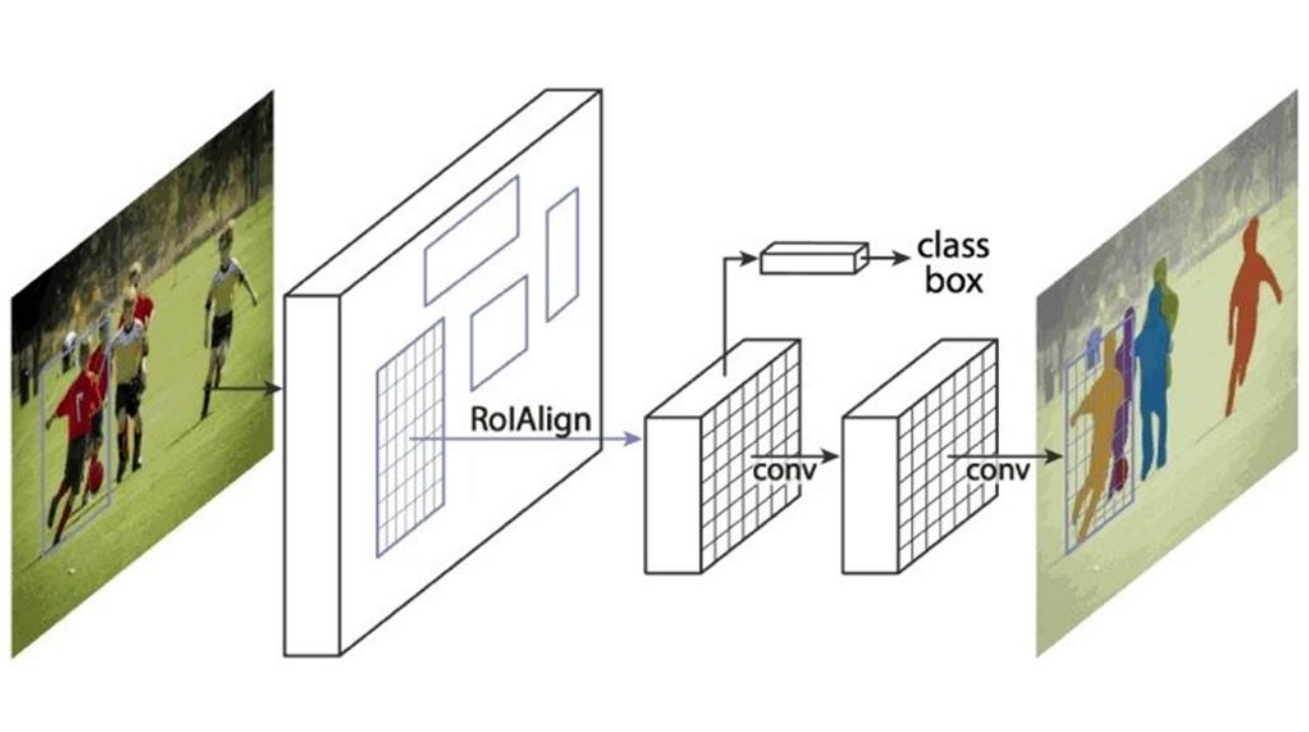
Cependant, des algorithmes plus avancés sont basés sur des contours actifs, des coupes de graphes, des champs aléatoires conditionnels et de Markov et des méthodes basées sur la parcimonie.

Au cours des dernières années, les modèles d' apprentissage en profondeur ont obtenu des performances remarquables.

.

Mask R-CNN

CNN + RPN + Classifieur binaire de masques .



En sortie

- ***Bounding Box***
- ***Label** de **l'objet détecté***
- ***Masque***

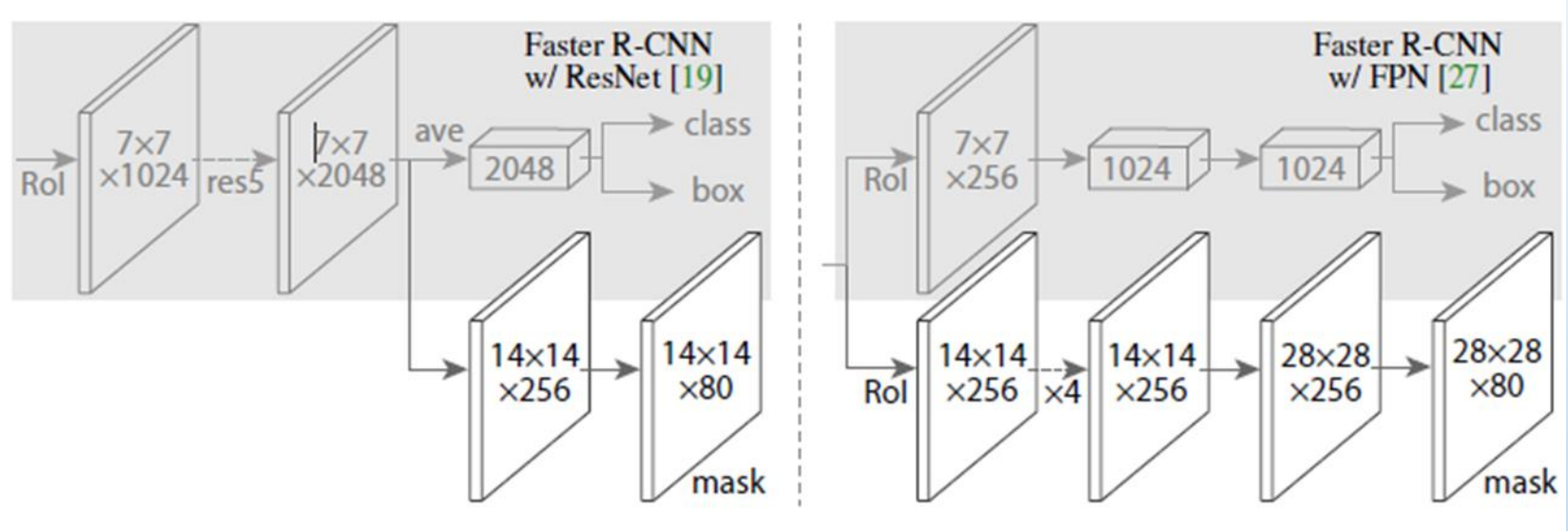
A chaque objet on associe un label et un masque

Tous les pixels de l'image sont classés : appartenance à un masque

Mask R-CNN

CNN + RPN + Classifieur binaire de masques .

L'élément clé de Mask R-CNN est l'alignement pixel à pixel, qui n'existe pas dans Fast/Faster R-CNN.

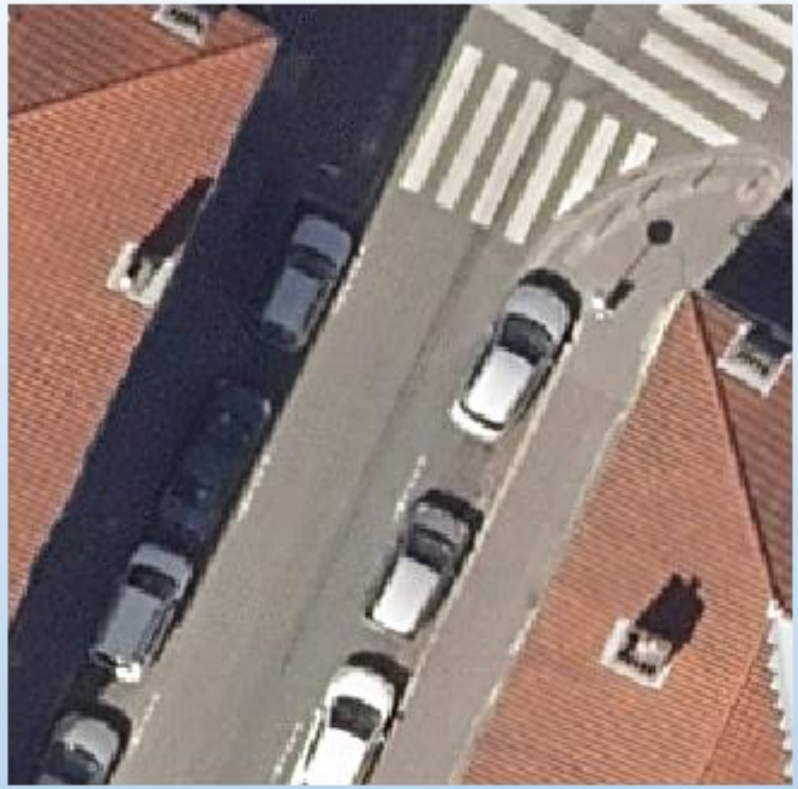


Le masque R-CNN adopte la même procédure en deux étapes :

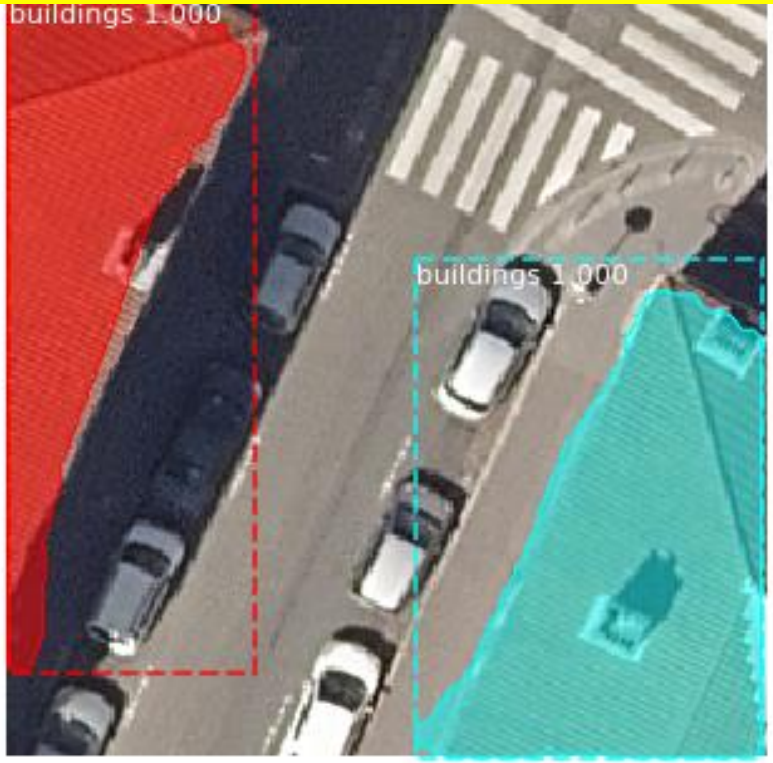
- *première étape identique : RPN,*
- *deuxième étape, parallèlement à la prédiction de la classe et du décalage de boîte, Mask R-CNN génère un masque binaire pour chaque RoI.*

Mask R-CNN

Image



Prédiction



- ***Bounding Box***
- ***Label objet***
- ***Pixels appartenant à l'objet***

Exemples d'applications

Mask R-CNN a été utilisé pour améliorer OpenStreetMap en identifiant les terrains de sport sur des images satellites

Evolutions matérielles

Evolutions de ces dernières années

La technologie d'imagerie a beaucoup progressé ces dernières années.

Les caméras sont plus petites, moins chères et de meilleure qualité.

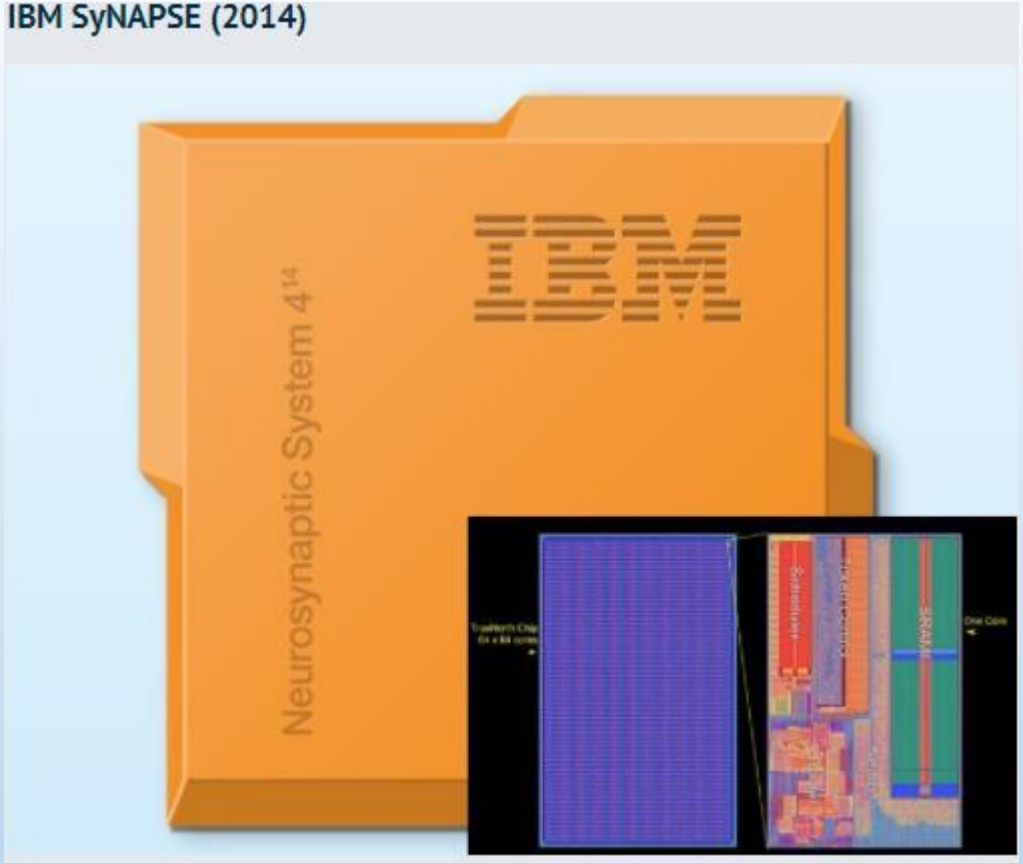
La puissance de calcul a considérablement augmenté et est devenue beaucoup plus efficace.

Les plates-formes informatiques ont évolué vers la parallélisation grâce au traitement multicœur, à l'unité de traitement graphique (GPU) et aux accélérateurs d'IA tels que les unités de traitement de tenseur (TPU)

Un tel matériel permet d'effectuer une vision par ordinateur pour la détection et le suivi d'objets en temps quasi réel.

Le développement rapide des réseaux de neurones à convolution (CNN) et la puissance de calcul améliorée du GPU sont les principaux moteurs de la grande avancée de la détection d'objets basée sur la vision par ordinateur.

Capacité des puces



En août 2014, IBM a présenté la première puce électronique « cognitive » vraiment puissante, TrueNorth.

Ses 5,4 milliards de transistors, gravés en 28 nm, sont organisés en s'inspirant du cerveau biologique, en neurones et en synapses.

On y trouve un réseau de 4094 cœurs neuro-synaptiques, soit un million de neurones et 256 millions de synapses programmables.

En regroupant 48 de ces processeurs dit « neuromorphiques » ensembles, IBM a pu égaler la composition d'un cerveau de rongeur, avec environ **48 millions de neurones**.

Le processeur sait identifier et reconnaître des éléments dans des images générées par 50 à 100 caméras à 24 images par seconde

Les puces sont, pour ne rien gâcher, très peu énergivores (seulement 70 mW par processeur).

Cerveau

- *85×10^{9} neurones .*
- *10^{4} Synapses/neurone ➔ 10^{15} synapses*

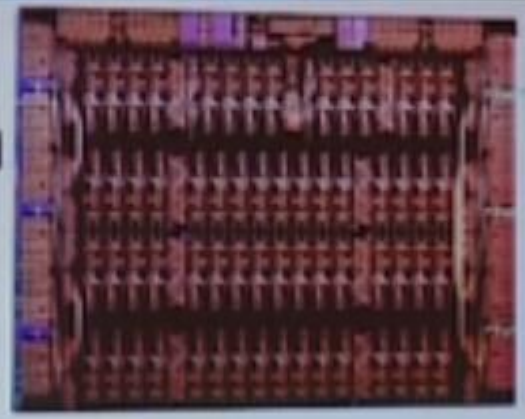
Fast Processors Today

•2016

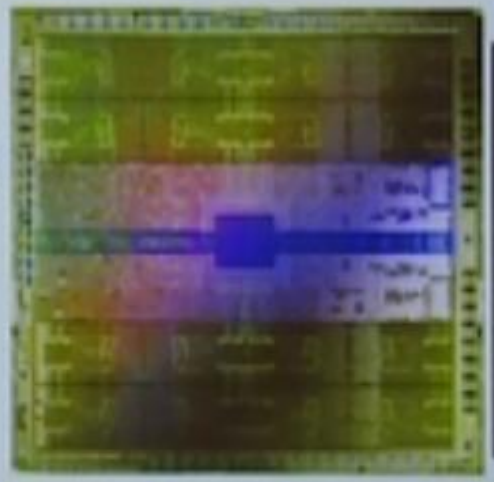
•Yann Lecun

Y LeCun

- Intel Xeon Phi CPU
  - $2x10^{12}$ operations/second
  - 240 Watts
  - 60 (large) cores
  - $3000



- NVIDIA Titan-Z GPU
  - $8x10^{12}$ operations/second
  - 500 Watts
  - 5760 (small) cores
  - $3000



- Are we only a factor of 10,000 away from the power of the human brain?
  - Probably more like 1 million: synapses are complicated
  - A factor of 1 million is 30 years of Moore's Law
  - 2045?

Capacité des puces

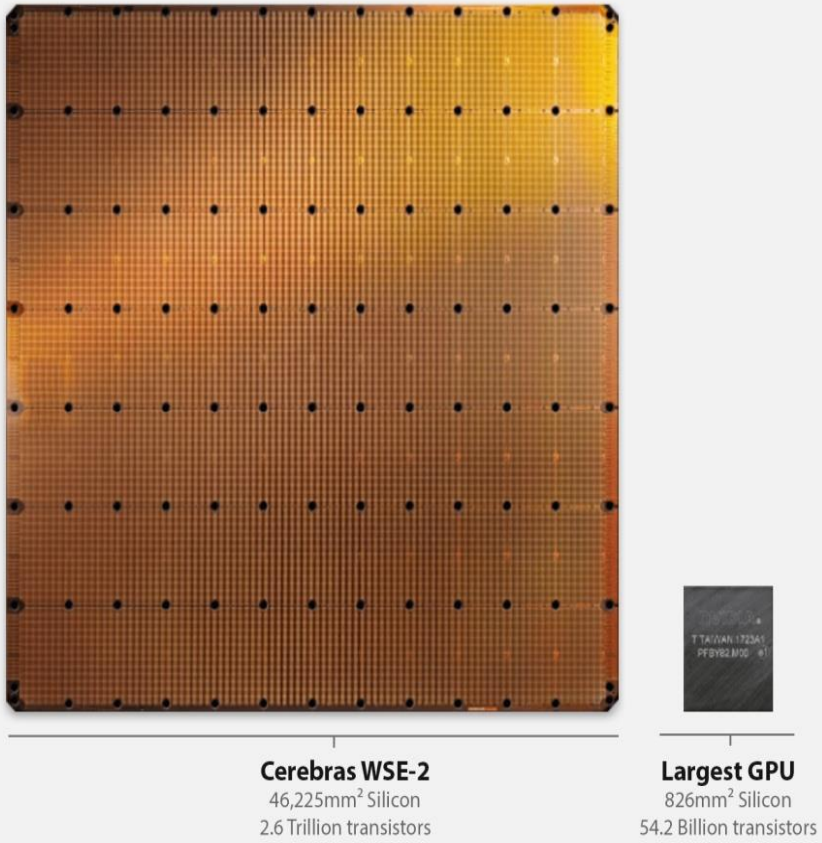
Avril 2021

La Société américaine Cerebras a développé le Wafer Scale Engine V2 à 850 000 cœurs intégré dans le système CS2

Puis a associé 192 systèmes CS-2 pour atteindre près de 163 millions de cœurs.

120 billions (120 10^{12}) connexions potentielles.

Notre cerveau : 10^{15} connexions synaptiques, soit un facteur 10 !!!



850 000 cœurs
Gravure en 7 nm

Utilisés pour entrainer GPT-3

Ce système consomme 23 kw !!

Cerveau humain 20 W

Détection d'objets

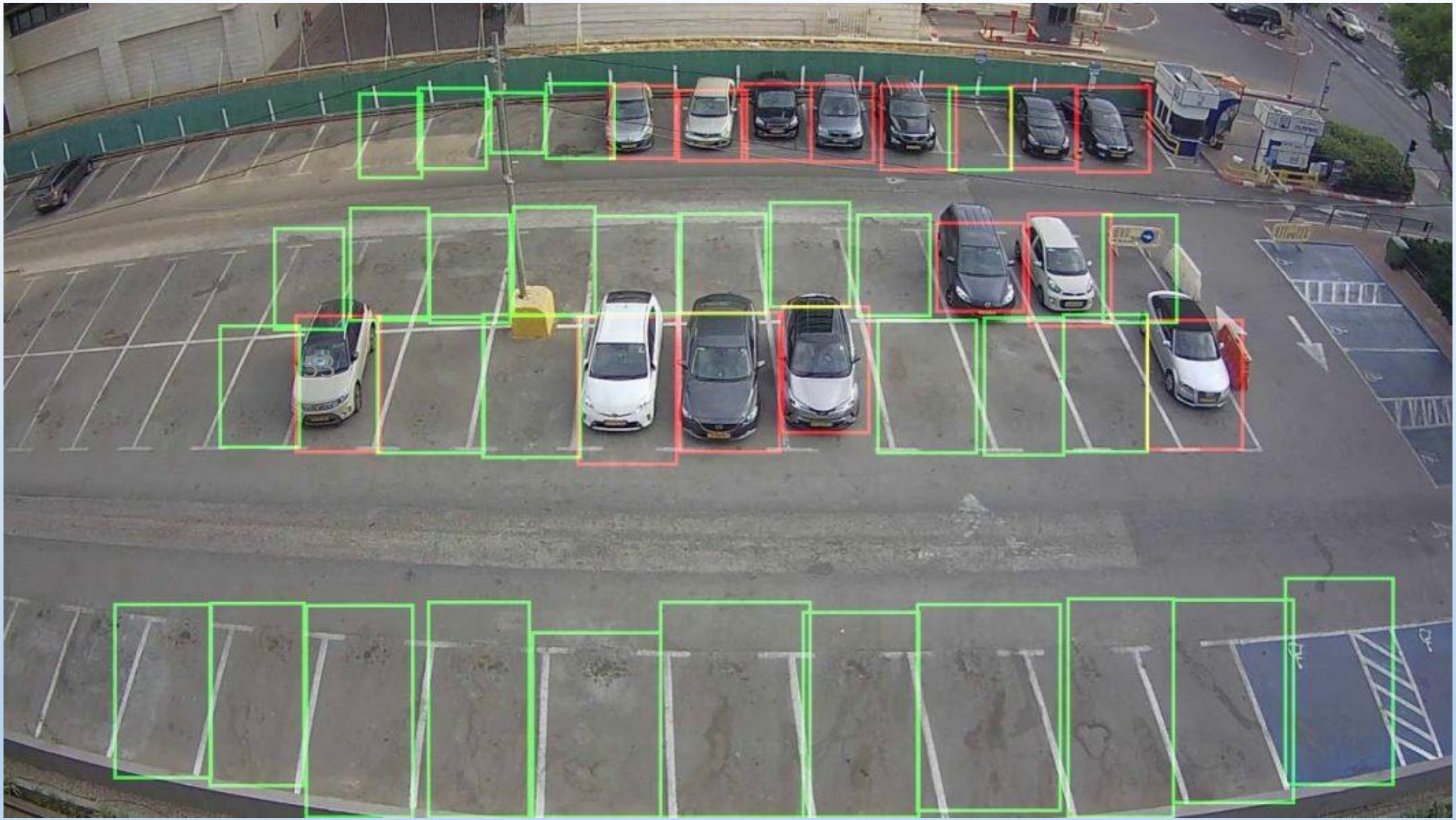
Applications

Exemples d'applications

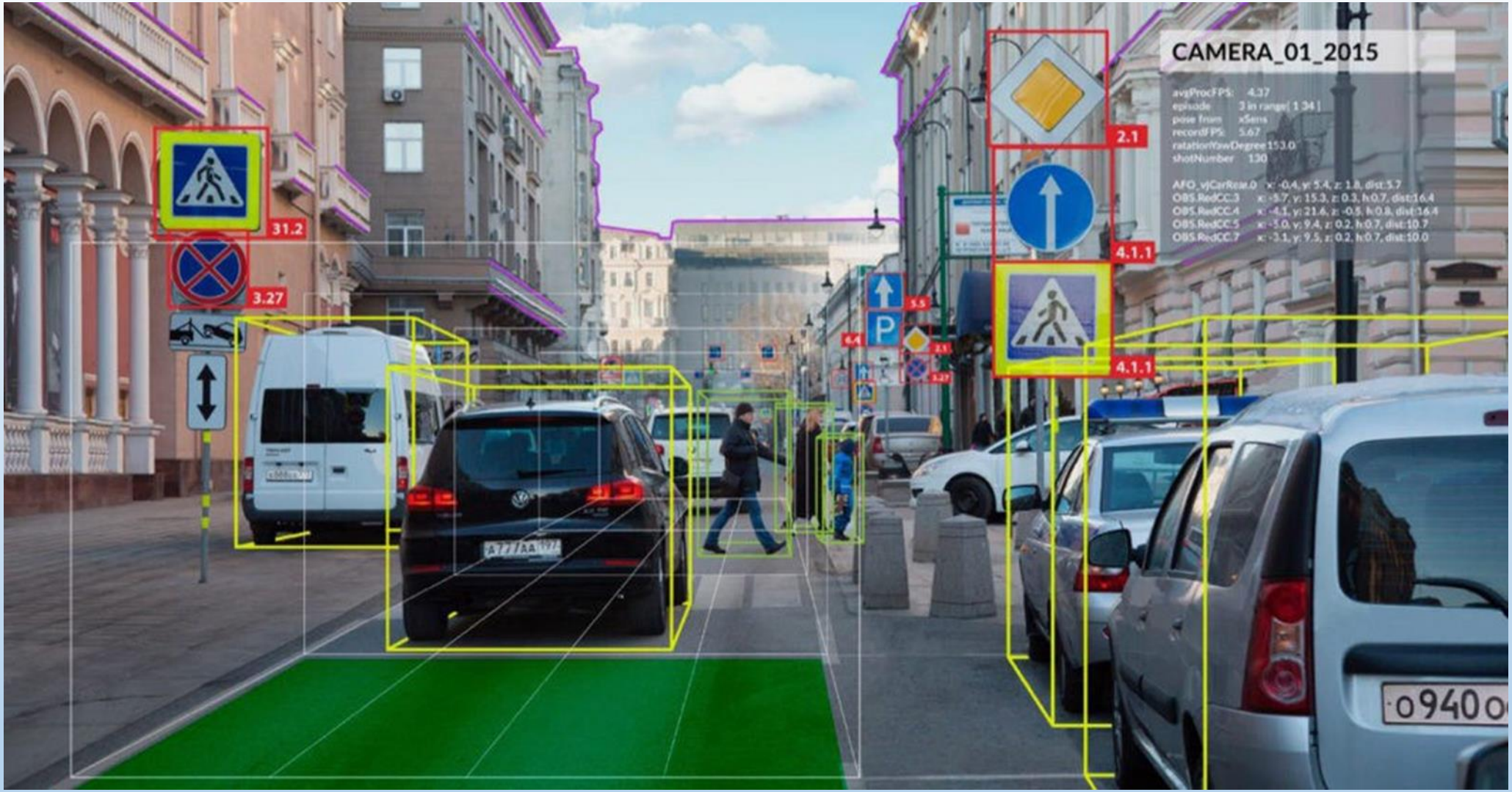
Large éventail d' applications de vision par ordinateur dans le monde réel :

- *Détection des panneaux de signalisation,*
- *Biologie,*
- *Evaluation des matériaux de construction*
- *Vidéosurveillance.*
- *Véhicules autonomes*
- *Systèmes avancés d'aide à la conduite (ADAS) : détection des surfaces navigables, des piétons….*

Applications



Exemples d'applications



Exemples d'applications

Détection d'objets basée sur le Deep Learning pour les véhicules (voitures, camions, vélos, etc.). Un exemple de trame d'une application commerciale temps réel avec reconnaissance IA sur le flux des caméras IP, construite sur Viso Suite .

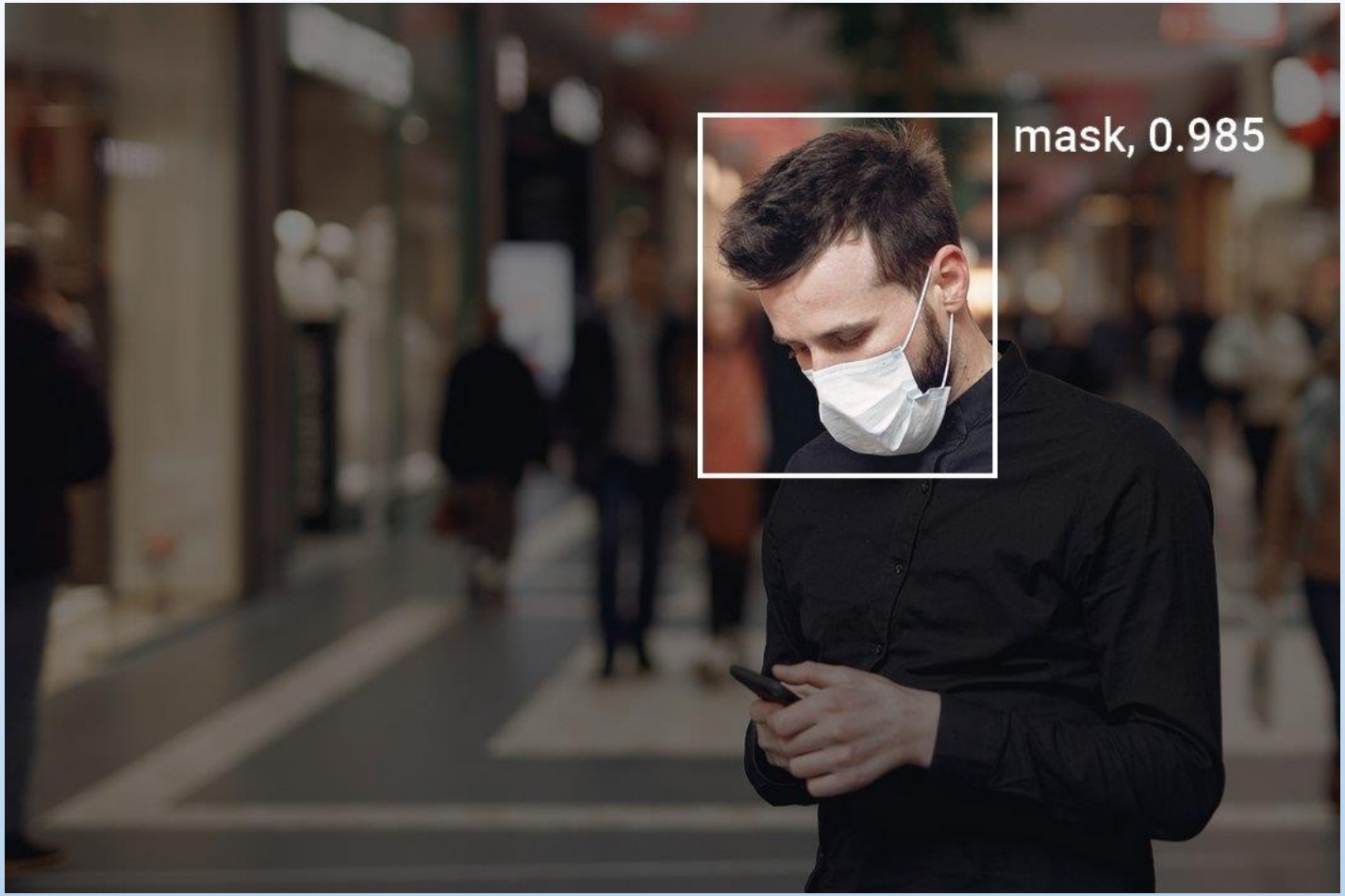
Exemples d'applications

https://viso.ai/wp-content/uploads/2021/01/9-cars-street-above-vehicle-detection.mp4?_=1

Détection d'objets basée sur le Deep Learning pour les véhicules (voitures, camions, vélos, etc.).

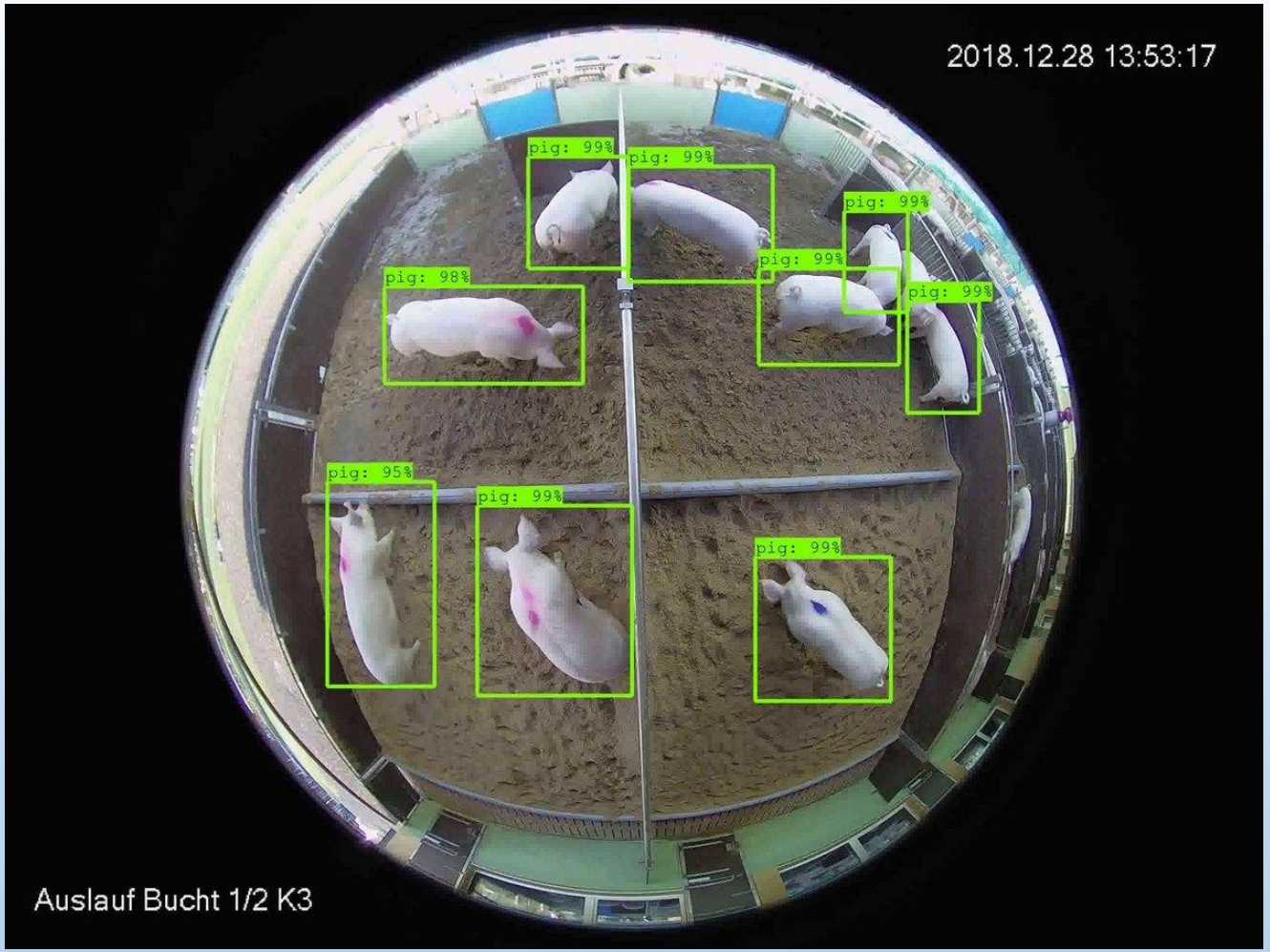
Exemple de trame d'une application commerciale temps réel avec reconnaissance IA sur le flux des caméras IP, construite sur ***Viso Suite***

Exemples d'applications



Détection de masque basée sur la caméra

Exemples d'applications



Système de surveillance des animaux pour une agriculture intelligente

Exemples d'applications

Carte thermique générée par Computer Vision pour analyser le comportement des clients et les flux de personnes

Exemples d'applications

Détection de la distance sociale avec la vision par ordinateur

Exemples d'applications

Estimation de pose avec apprentissage en profondeur
dans les applications de santé

Exemples d'applications
Cartographie

Sytèmes d'IA /couplés à disponibilité d'images satellites ou aériennes

➔ *révolution dans le domaine de la* ***cartographie.***

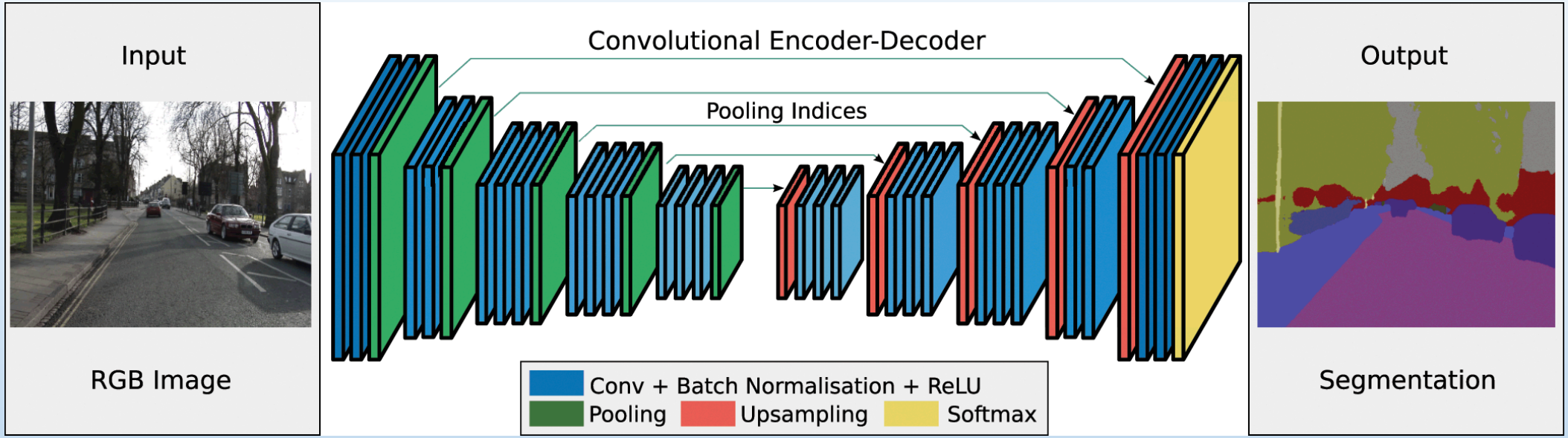
Constitution de bases de données géographiques à partir de ***données raster****.*

Pour cela, l'opération à mettre en œuvre est une opération de ***segmentation sémantique :*** *identifier et détourer les éléments d'une image en associant chaque pixel à une catégorie donnée en cartographie, cela peut par exemple être les catégories suivantes : bâtiments, routes, végétation, etc.*

Les objets ou zones d'intérêt pourront ensuite être facilement reconstruits par vectorisation du résultat.

L'apprentissage supervisé*, et plus particulièrement les modèles de* ***deep-learning****, permettent de réaliser efficacement et de manière automatique des opérations de segmentation sémantique.*

Exemples d'applications
Cartographie



SegNet : architecture représentant les étapes de convolution et déconvolution.

Entrainement des systèmes de détection

Données d'entraînement
Annotations d'images

Bases de données :
- ***Données d'entraînement***
- ***Données de validation***
- ***Données de test***

Construites en général sur le mode participatif

Annotations d'images

Ces applications nécessitent pour apprentissage la disponibilité de jeux de données annotées.

Images Annotées :

- *ImageNet*
- *…*

Ensembles de données de segmentation d'images les plus populaires sont :

- *Le PASCAL Visual Object Classes (VOC)*
- *Le Microsoft Common Objects in Context MS COCO*
- *ADE20K*
- *KITTI*
- *Ensemble de données YouTube-Objects*
- *….*

Comment annoter les images ?

Comment annoter des images ?

Étape 1 : Préparer l’ ensemble de données d'image.

Étape 2 : Spécifier les étiquettes de classe des objets à détecter.

Étape 3 : Dans chaque image, dessiner un cadre autour de l'objet à détecter.

Étape 4 : Sélectionner l'étiquette de classe pour chaque boîte dessinée.

Étape #5 : Exporter les annotations au format requis (COCO JSON etc..)

Bibliothèques d'images annotées

ImageNet

- *base de données d'images annotées*
- *Pour travaux de recherche en vision par ordinateur.*

En 2016 :

- *plus de dix millions d'URLs annotées à la main pour indiquer quels objets sont représentés dans l'image*
- *plus d'un million d'images bénéficient de boîtes englobantes autour des objets.*

***Concours annuel** : ImageNet Large Scale Visual Recognition Challenge (ILSVRC), ou "Compétition ImageNet de Reconnaissance Visuelle à Grande Échelle".*

En 2022 : 14 197 122 images, 21841 synsets indexés

Le jeu de données ImageNet le plus utilisé, ILSVRC 2012-2017, est composé d'environ 1.5 millions d'images, réparties en environ 90% d'images d'entraînement, 3% de validation et 7% de test

Synset : synonym set *ex : car, auto, automobile*,

Ensembles de données de segmentation d'images

PASCAL Visual Object Classes (VOC) Challenge *fournit des ensembles de données d'images et des annotations accessibles au public.*

*Le **PASCAL VOC** est l'un des ensembles de données les plus populaires en vision par ordinateur, avec des images annotées disponibles pour 5 tâches : classification, segmentation, détection, reconnaissance d'action et disposition de la personne.*

*Pour les tâches de segmentation, PASCAL VOC prend en charge **21 classes d'étiquettes d'objets** :*

> *véhicules, maison, animaux, avion, vélo, bateau, bus, voiture, moto, train, bouteille, chaise, table à manger, plante en pot, canapé, TV/moniteur, oiseau, chat, vache, chien, cheval, mouton et personne.*

*Les données d'apprentissage/validation du PASCAL VOC comportent **11 530 images contenant 27 450 objets annotés ROI et 6 929 segmentations.***

Ensembles de données de segmentation d'images

PASCAL Visual Object Classes (VOC) Challenge *fournit des ensembles de données d'images et des annotations accessibles au public.*

*Le **PASCAL VOC** est l'un des ensembles de données les plus populaires en vision par ordinateur, avec des images annotées disponibles pour 5 tâches : classification, segmentation, détection, reconnaissance d'action et disposition de la personne.*

*Pour les tâches de segmentation, PASCAL VOC prend en charge **21 classes d'étiquettes d'objets** :*

> *véhicules, maison, animaux, avion, vélo, bateau, bus, voiture, moto, train, bouteille, chaise, table à manger, plante en pot, canapé, TV/moniteur, oiseau, chat, vache, chien, cheval, mouton et personne.*

*Les données d'apprentissage/validation du PASCAL VOC comportent **11 530 images contenant 27 450 objets annotés ROI et 6 929 segmentations.***

Ensembles de données de segmentation d'images

PASCAL Visual Object Classes (VOC)

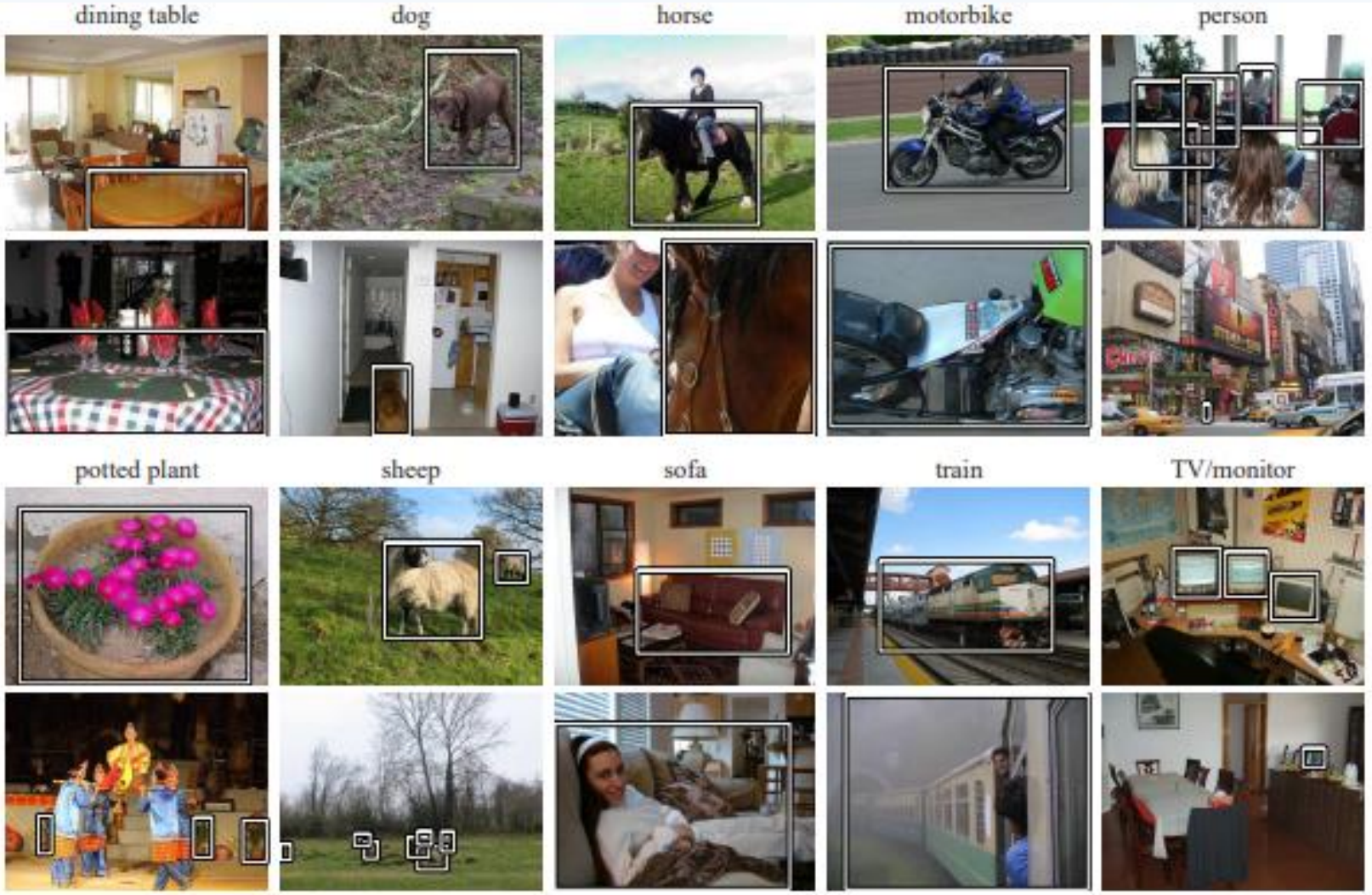


Fig. 1 Example images from the VOC2007 dataset. For each of the 20 classes annotated, two examples are shown. Bounding boxes indicate all instances of the corresponding class in the image which are marked as "non-difficult" (see Sect. 3.3) – bounding boxes for the other classes are available in the annotation but not shown. Note the wide range of pose, scale, clutter, occlusion and imaging conditions.

Ensembles de données de segmentation d'images

Une image annotée du jeu de données MS COCO

Microsoft Common Objects in Context (MS COCO) *est un ensemble de données de détection , de segmentation et de sous-titrage d'objets à grande échelle .*

COCO est basé sur un total de ***2,5 millions d'instances*** *segmentées étiquetées dans* ***328 000 images****, contenant des photos de* ***91 types d'objets*** *(qui seraient facilement reconnus par une personne de 4 ans).*

Ensembles de données de segmentation d'images

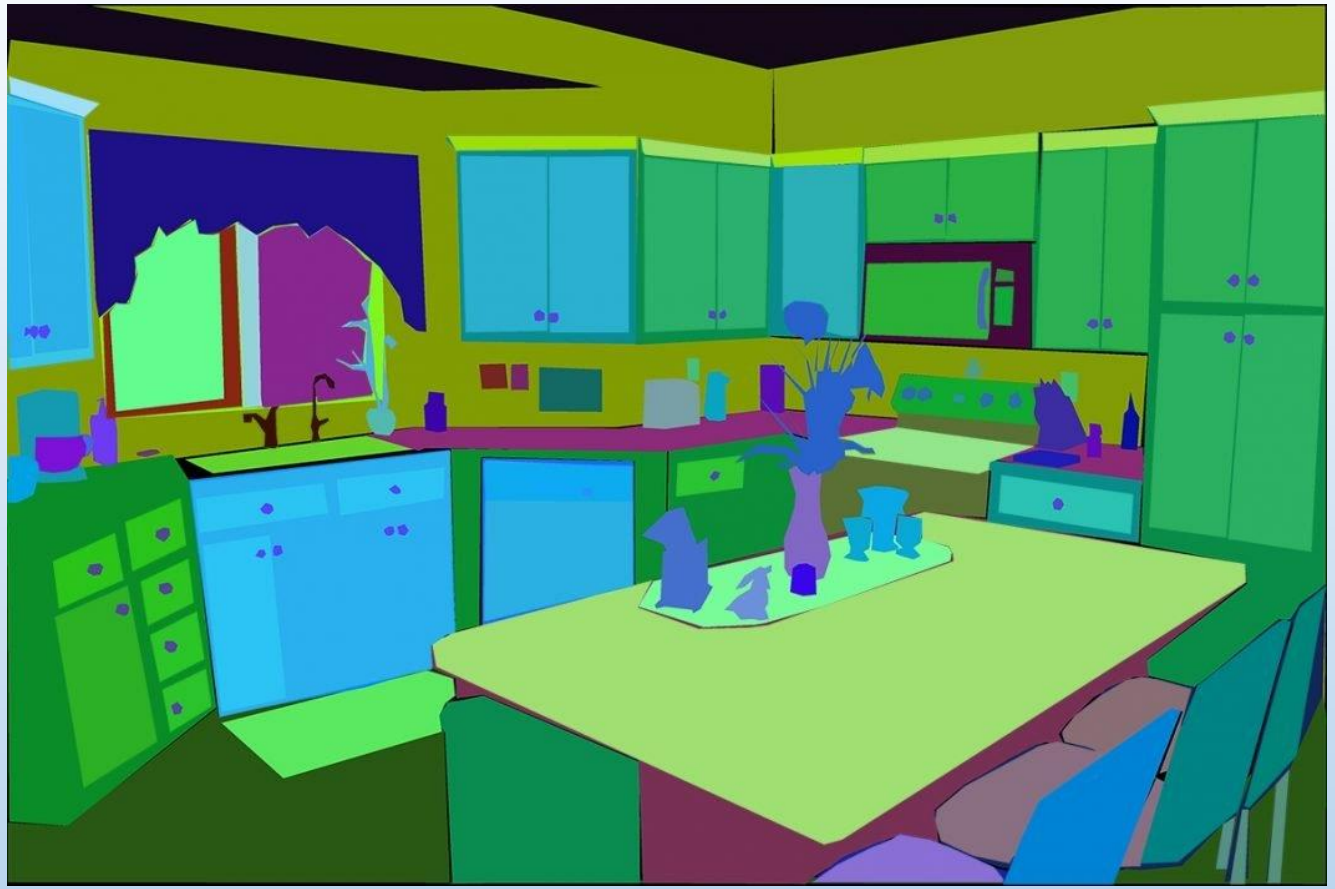
ADE20K

Inclut un masque de segmentation d'objet et un masque de segmentation de pièces.

20 210 images dans l'ensemble d'apprentissage,

2 000 images dans l'ensemble de validation

3 000 images dans l'ensemble de test



Jeu de données ADE20K pour la segmentation d'images
Source : ADE20K

Ensembles de données de segmentation d'images

KITTI

Pour la robotique mobile et la conduite autonome. Il contient des heures de vidéos de scénarios de trafic capturés en conduisant dans la ville de taille moyenne de Karlsruhe (sur les autoroutes et dans les zones rurales). En moyenne, dans chaque image, jusqu'à 15 voitures et 30 piétons sont visibles.

Échantillon de segmentation d'image KITTI - Source : KITTI

Fin