



# Détection d'objets Java

OpenCV, CNN et Yolo



Encadré Par:

M Bahri Abdelkhalak

Réaliser Par:

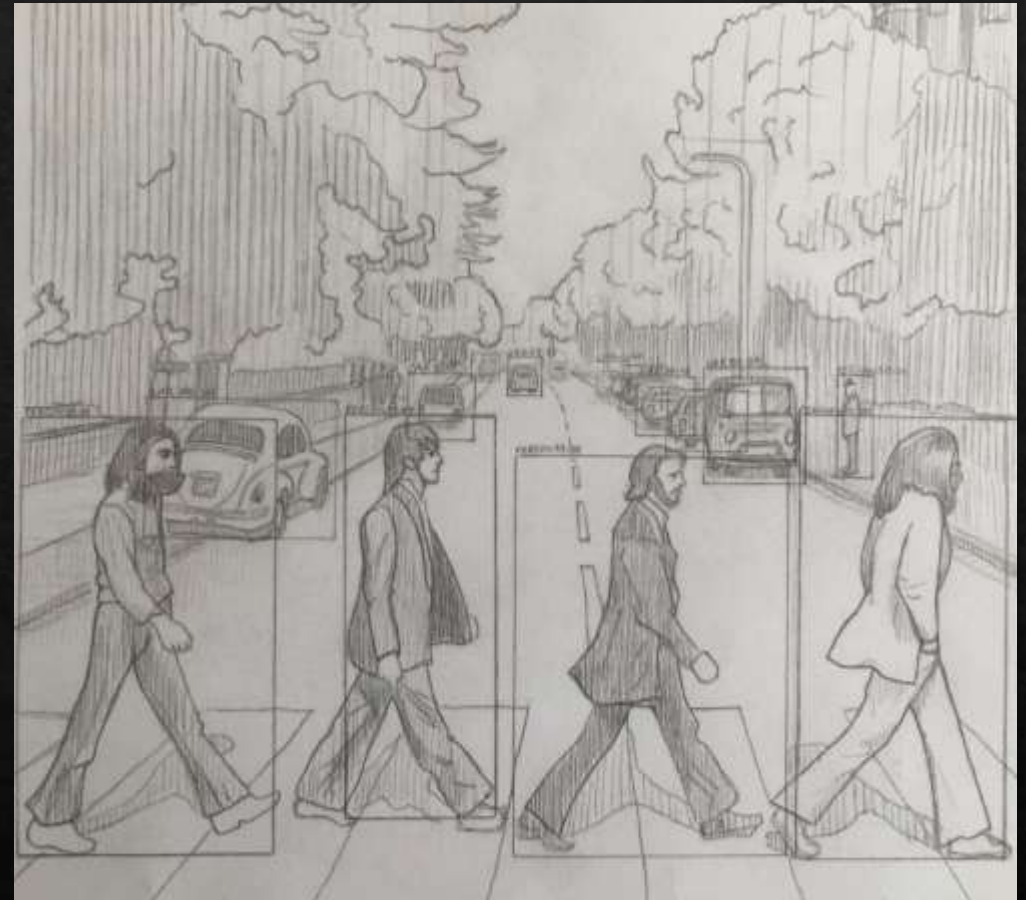
Moulay Taj Zakaria

Bahadi Ibrahim

El Khanoussi Redouan

# Détection D'objets

La détection d'objets est une technologie informatique liée à la vision par ordinateur et au traitement d'images qui traite de la détection d'instances d'objets sémantiques d'une certaine classe (tels que les humains, les bâtiments ou les voitures) dans les images et les vidéos numériques .La détection d'objets a des applications dans de nombreux domaines de la vision par ordinateur, y compris la récupération d'images et la vidéosurveillance.



# YOLO

YOLO abréviation de  
« You Only Look Once »  
l'un des algorithmes de  
détection d'objets les plus  
efficaces.





# Les cellules de grille (Grid cells)

Diviser une image en  
cellules Grille

$S \times S$ . C'est un concept  
unique en Yolo.

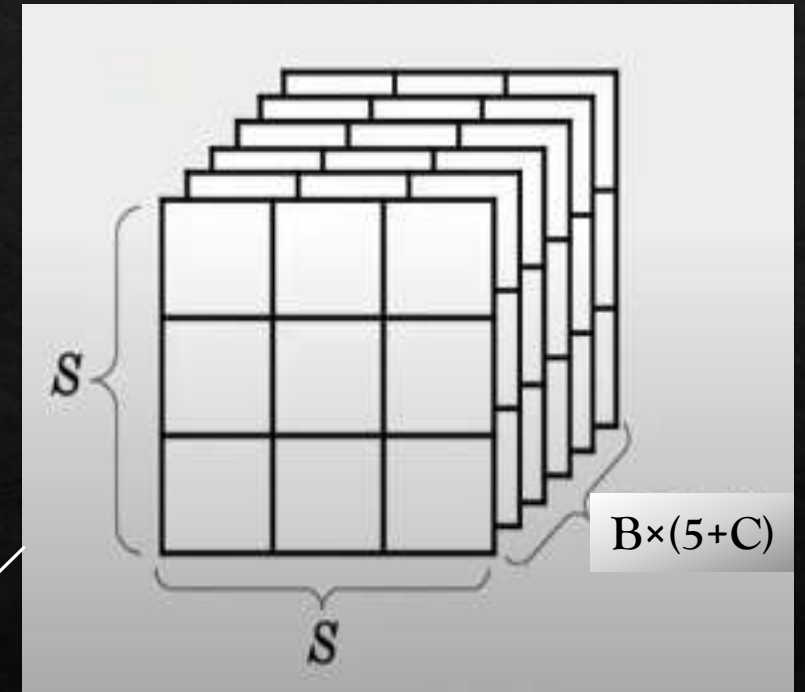


# Générer les boite d'ancre (Bounding Box)

Le Convnet prévoit un y pour chaque cellule, de sorte que la taille du tenseur de sortie devrait être :

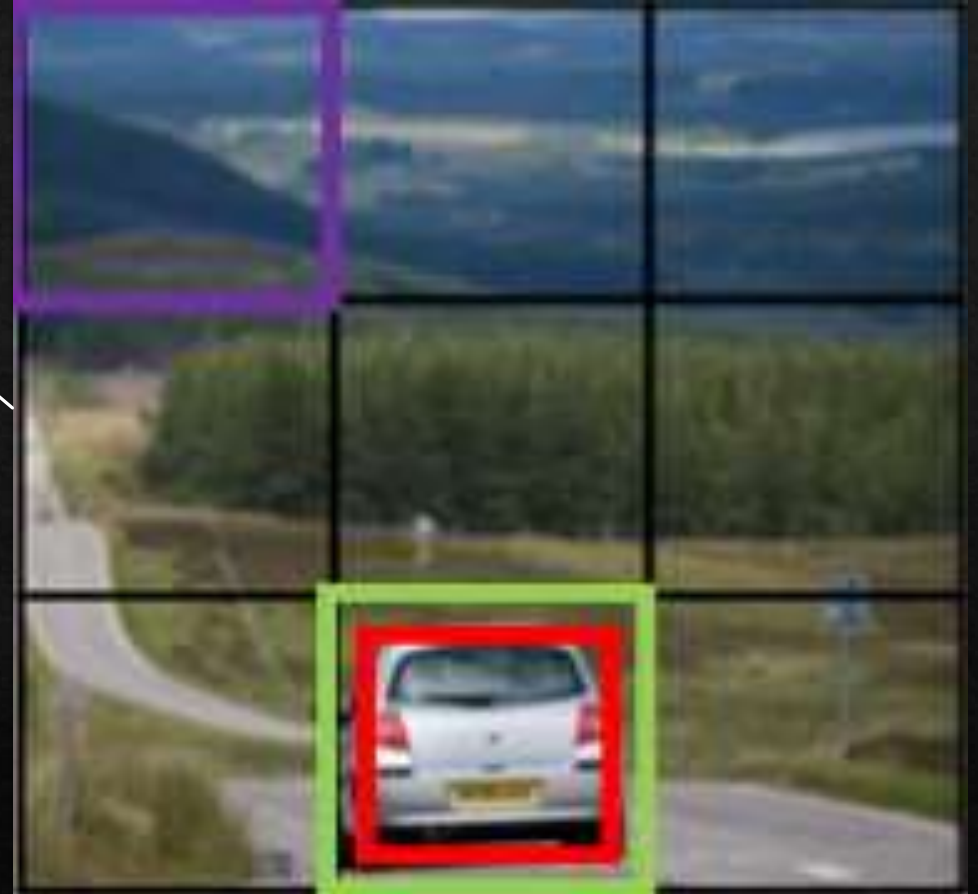
$$S \times S \times B \times (5+C).$$

- S dans ce cas égale à 3
- B c'est le nombres des boîtes d'ancre(Bounding Box).
- C le nombre des labels du classes.



$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

Exemple d'image divisé en  
3x3 avec 2 boîtes d'ancre,  
à savoir qu'on entraîne  
l'algorithme pour détecter  
3 objets donc vecteur de  
sortie y est  
 $3 \times 3 \times 2 \times (5+3)$ .  
Ou  $3 \times 3 \times 16$ .





1. Pedestrian
2. Car
3. Motorcycle



$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \\ p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

$$y = \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix}$$

$$y = \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$y = 3 \times 3 \times 16$$

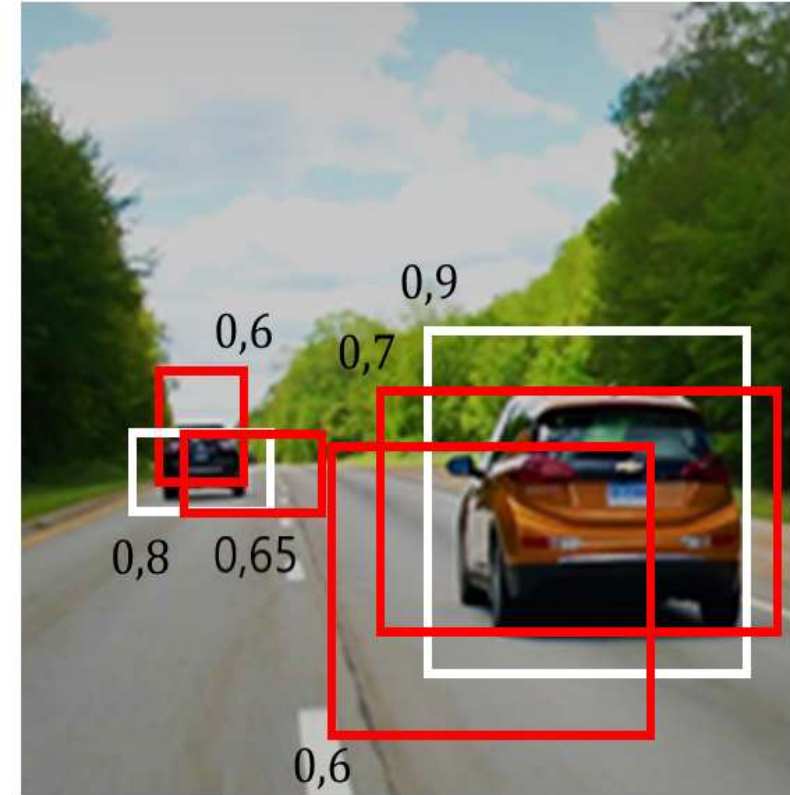
$$y = 3 \times 3 \times 2 \times 8$$

# anchors

# parameters

# Comment fonctionne une suppression non maximale ?

La non-max suppression nettoie ces multiples boîtes englobantes prévent pour le même objet.



19 × 19

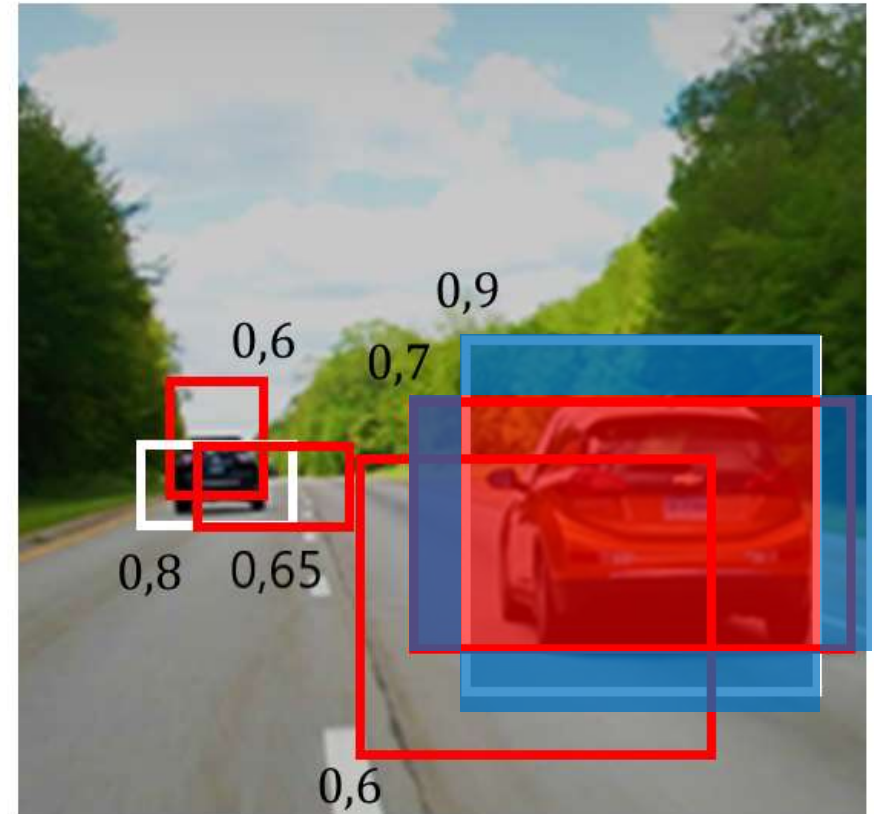
Jeter toutes les boîtes avec  $pc < 0,6$

Alors qu'il reste des cases :

- ◆ choisir la case avec la plus grande sortie  $pc$  qui, en tant que prédiction.
- ◆ éliminer toute boîte restante dont l'IoU est  $> 0,5$  avec la boîte produite à l'étape précédente.

$$\text{IoU (Intersection over Union)} = \frac{\text{taille de la zone rouge}}{\text{taille de la zone Bleue}}$$

Vrai si  $\text{IoU} \geq 0.3$



$19 \times 19$



