



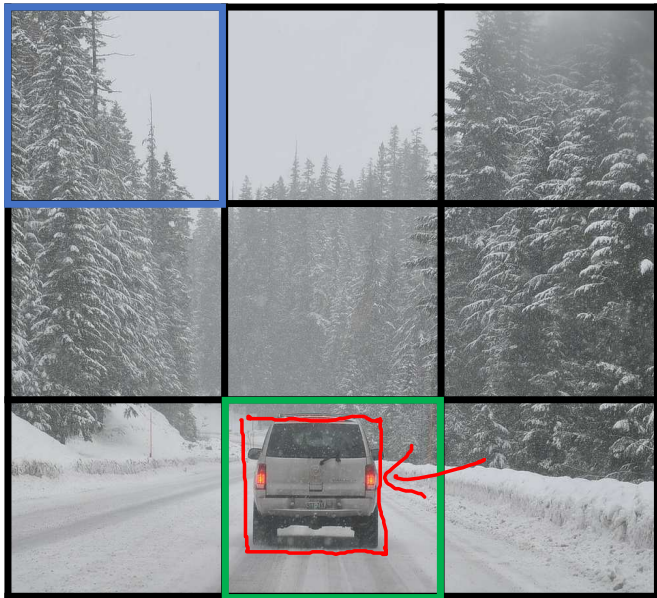
deeplearning.ai

Object Detection

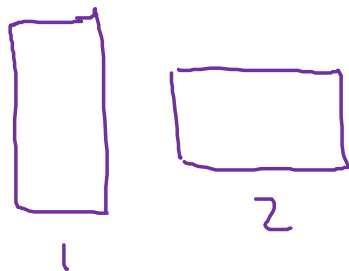
Putting it together:
YOLO algorithm

Training

- 1 - pedestrian
- 2 - car ←
- 3 - motorcycle



$y =$



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

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

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

$3 \times 3 \times 16$

y is $3 \times 3 \times 2 \times 8$

$19 \times 19 \times 16$

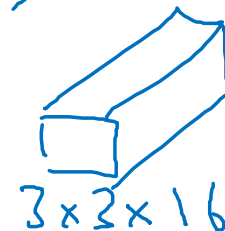
$19 \times 19 \times 40$

#anchors

$5 + \#classes$



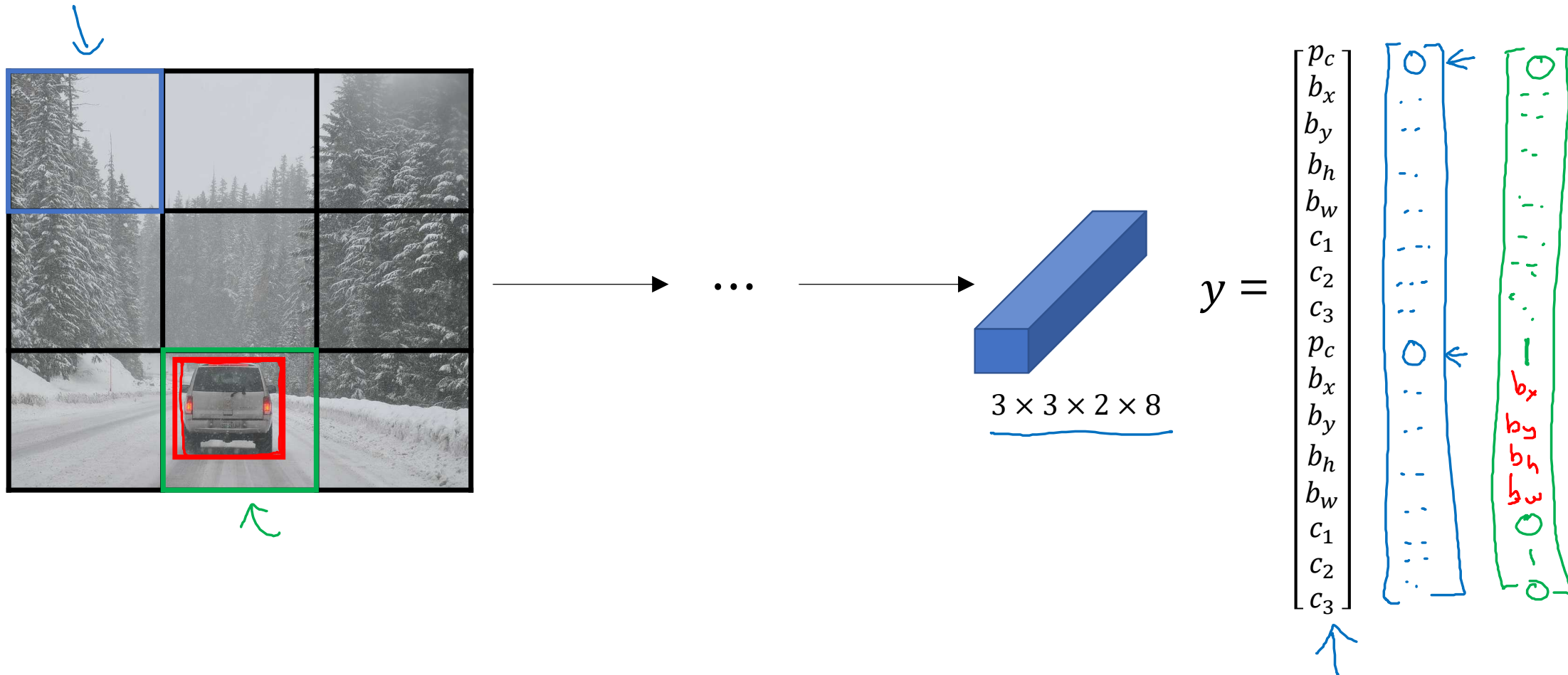
→ ConvNet →



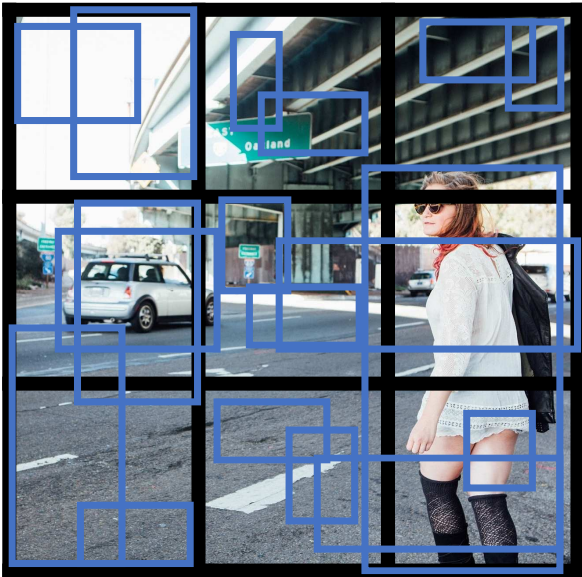
Andrew Ng

[Redmon et al., 2015, You Only Look Once: Unified real-time object detection]

Making predictions



Outputting the non-max suppressed outputs



- For each grid cell, get 2 predicted bounding boxes.
- Get rid of low probability predictions.
- For each class (pedestrian, car, motorcycle) use non-max suppression to generate final predictions.