deeplearning.ai

Object Detection

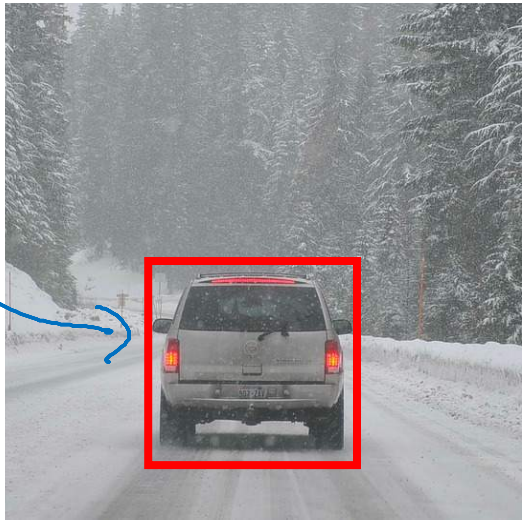
Object localization

What are localization and detection?

Image classification

"Car"

Classification with localization



"Car"

1 object

Detection

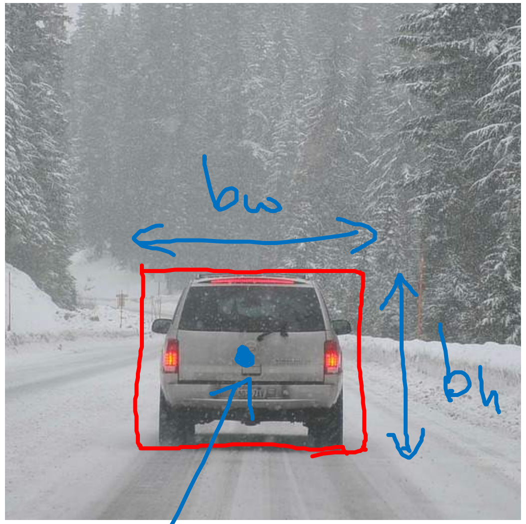
multiple objects

Classification with localization

(0,0)

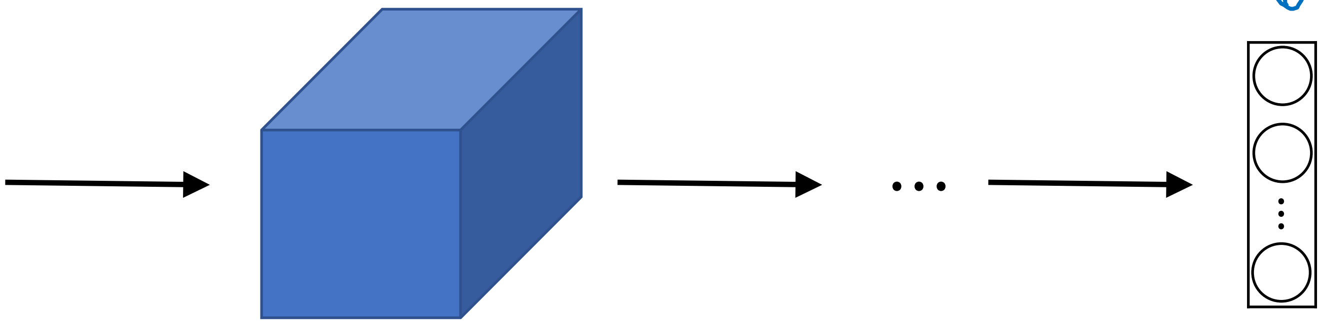
b_w

b_h



b_x, b_y

(1,1)



ConvNet

softmax (4)

b_x, b_y, b_h, b_w

bounding box

1 - pedestrian

2 - car

3 - motorcycle

4 - $\boxed{\text{background}}$

$b_x = 0.5$

$b_y = 0.7$

$b_h = 0.3$

$b_w = 0.4$

Defining the target label y

1 - pedestrian

2 - car

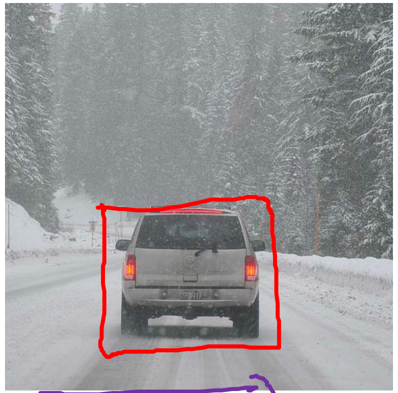
3 - motorcycle

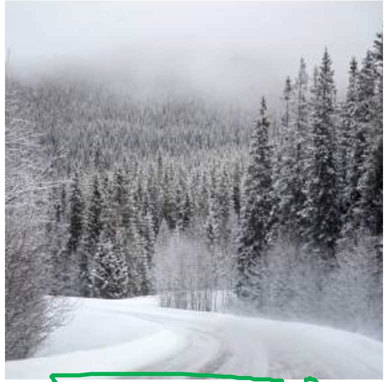
4 - background

P - Is any of the class object prese

Need to output b_x, b_y, b_h, b_w, class label (1-4)

$x =$





$$\mathcal{L}(\hat{y}, y) = \begin{cases} (\hat{y}_1 - y_1)^2 + (\hat{y}_2 - y_2)^2 + \dots + (\hat{y}_8 - y_8)^2 & \text{if } y_1 = 1 \\ (\hat{y}_1 - y_1)^2 & \text{if } y_1 = 0 \end{cases}$$

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

is there any object?

(x, y)

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

p_c

"don't care"

question mark d

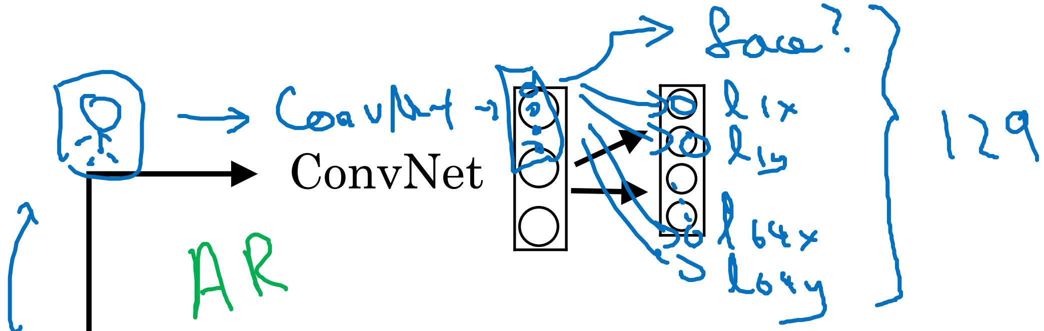
Andrew Ng

deeplearning.ai

Object Detection

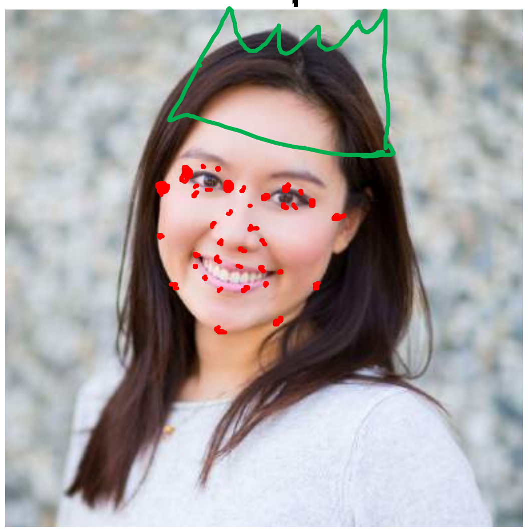
Landmark detection

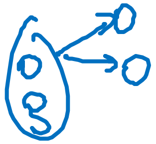
Landmark detection



If we want to detect

b_x, b_y, b_h, b_w





$$\left.\begin{matrix} l_{1x}, l_{1y}, \\ l_{2x}, l_{2y}, \\ l_{3x}, l_{3y}, \\ l_{4x}, l_{4y} \\ \vdots \\ l_{64x}, l_{64y} \end{matrix}\right\} X, y$$

$$\begin{matrix} l_{1x}, l_{1y}, \\ \vdots \\ l_{32x}, l_{32y} \end{matrix}$$

Andrew Ng

deeplearning.ai

Object Detection

Object detection

Car detection example

Training set:

x	y
	1
	1
	1
	0
	0

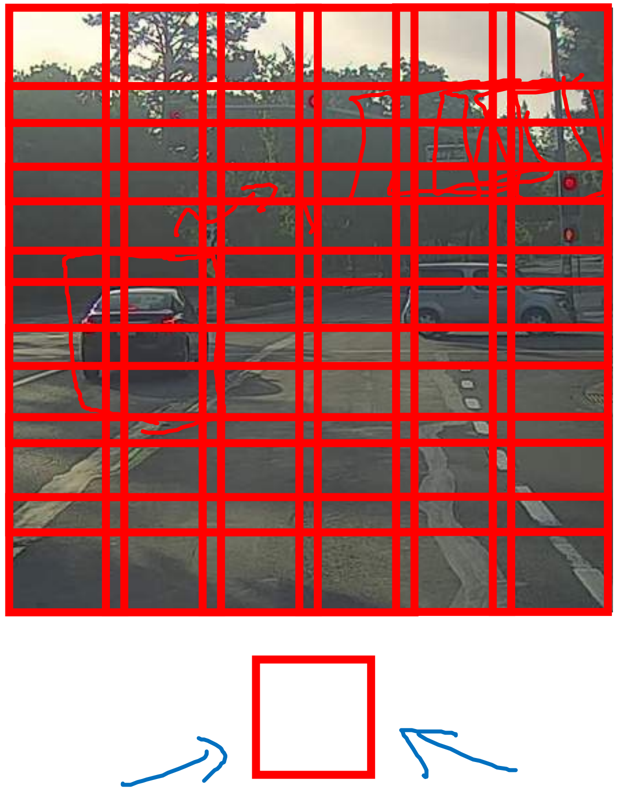
These are the closely cropped images of the dataset and

 → ConvNet → y

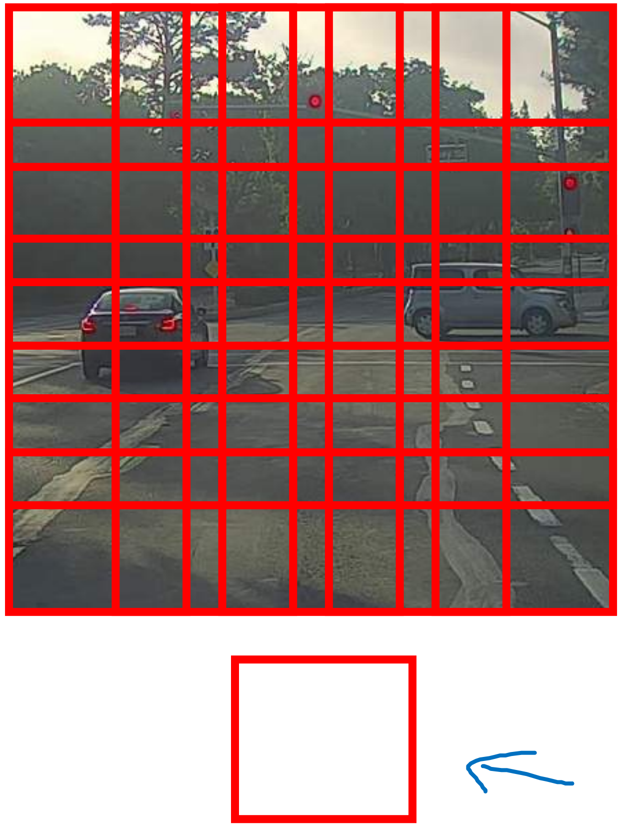
Andrew Ng

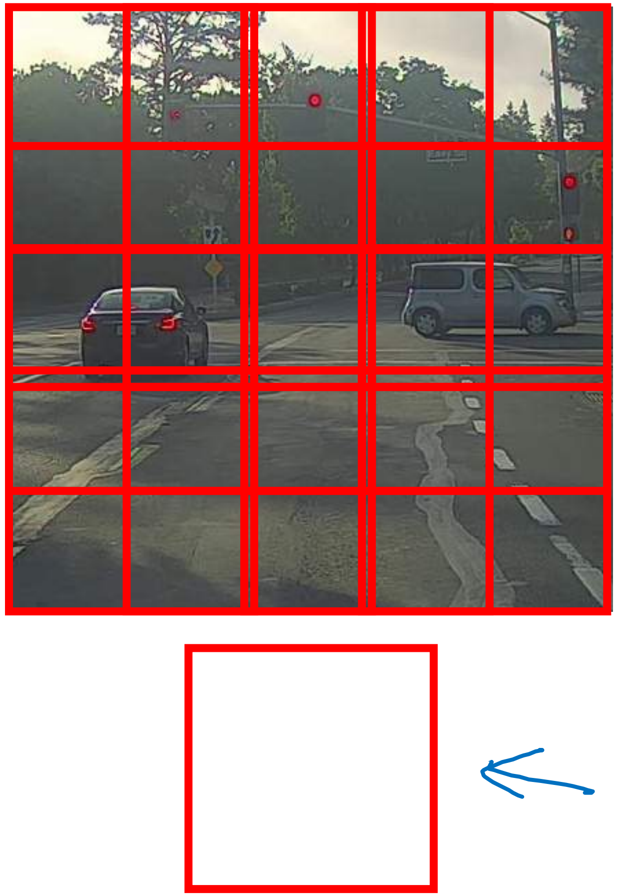
Sliding windows detection

ConvNet → 0



ConvNet





Computation cost

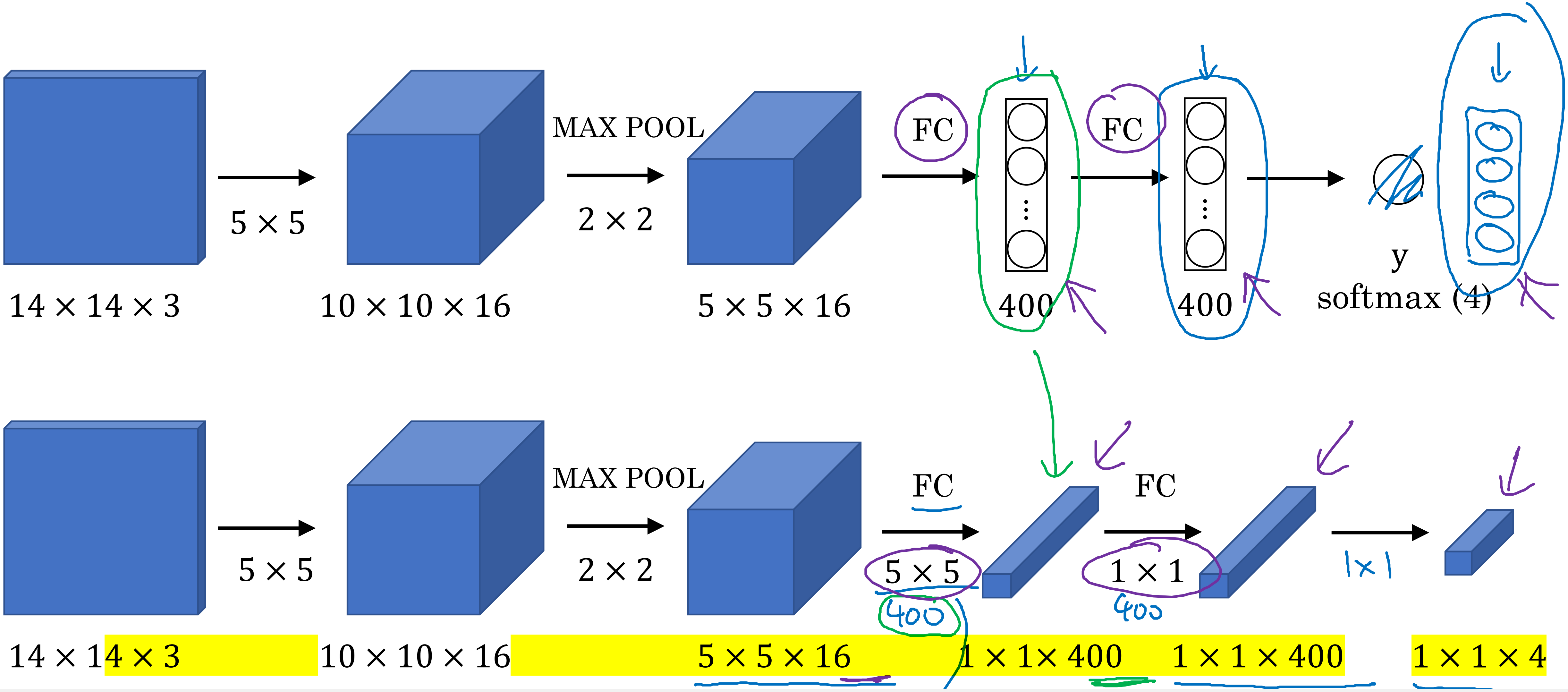
Andrew Ng

deeplearning.ai

Object Detection

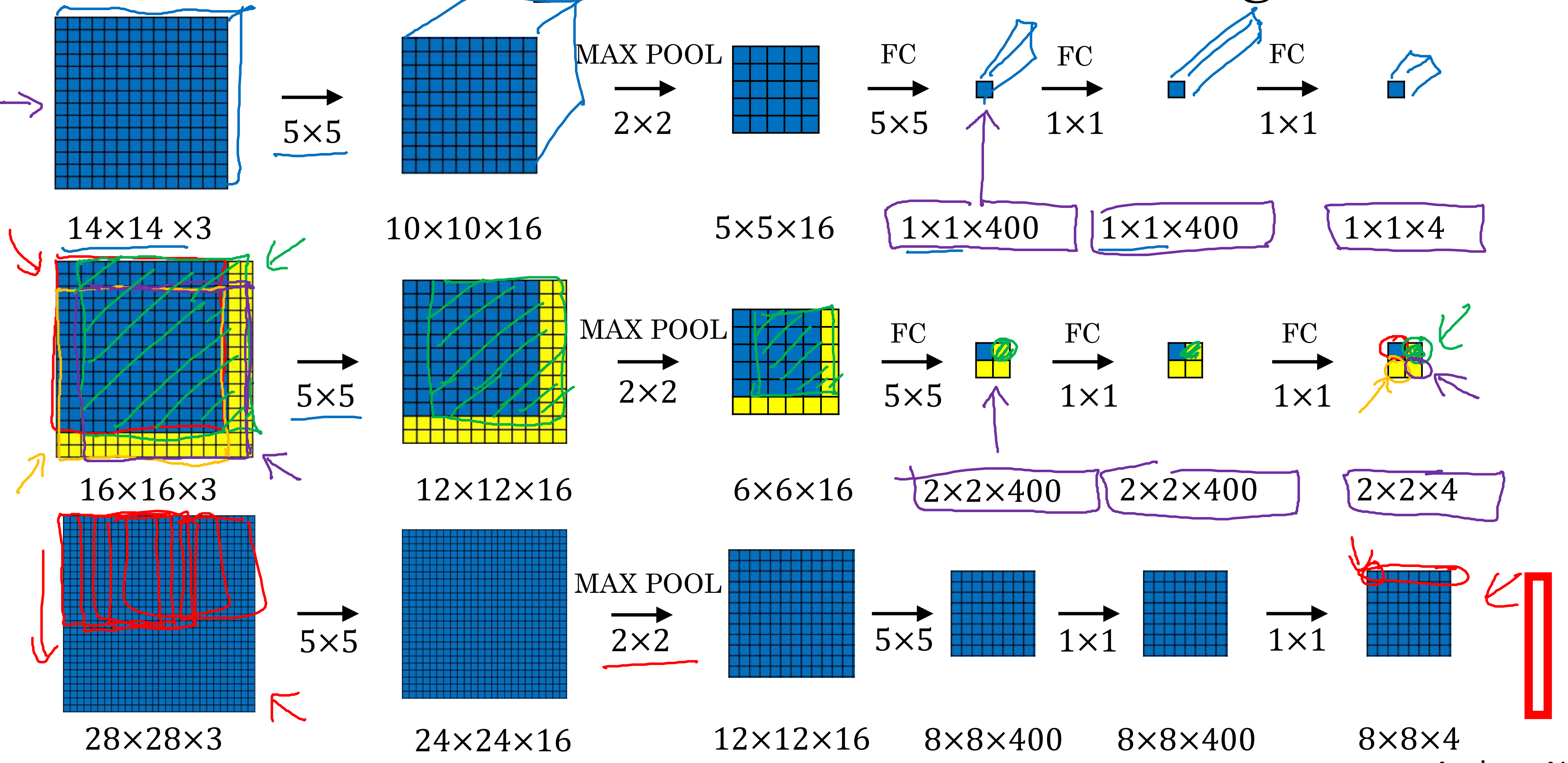
Convolutional implementation of sliding windows

Turning FC layer into convolutional layers



Below Slide : Let's say our input is 16x16x3 and our window is 12x12x3 then we will need to slide our window 4 times assuming stride of 2. As we see with different coloured windows here, there involves a lot of duplic

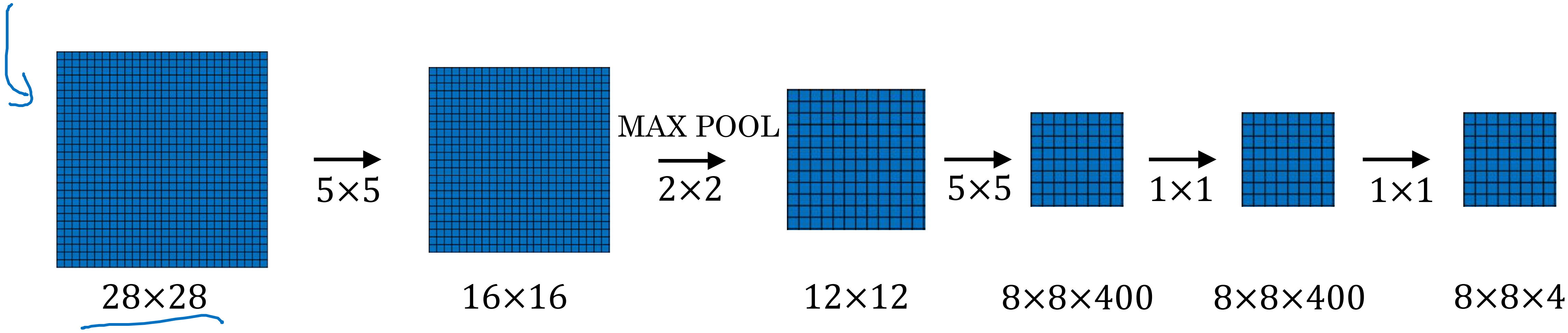
Convolution implementation of sliding windows

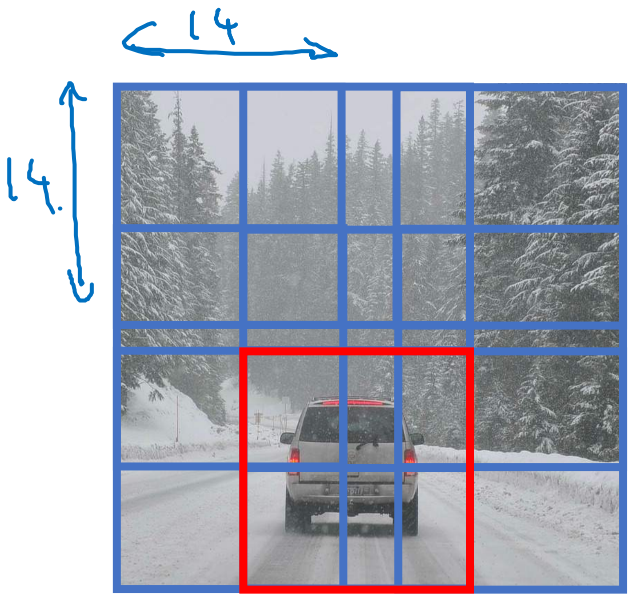


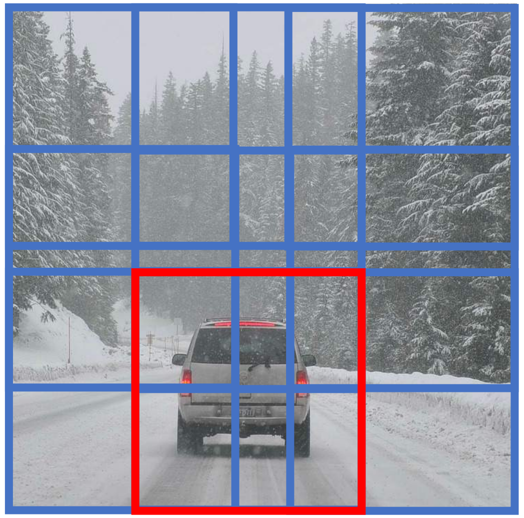
[Sermanet et al., 2014, OverFeat: Integrated recognition, localization and detection using convolutional networks]

Andrew Ng

Convolution implementation of sliding windows







Andrew Ng

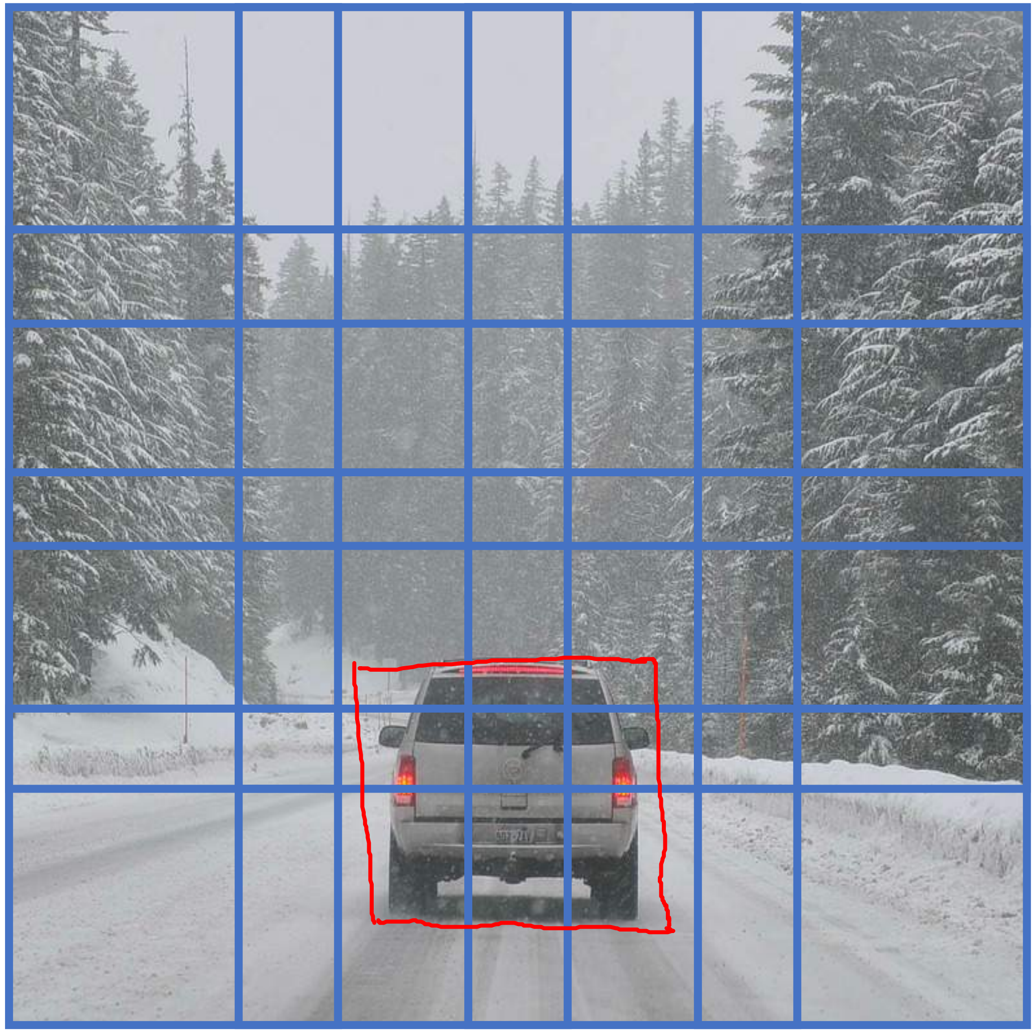
Sliding window approach may result in a rectangle such that none of the rectangle fits the object although it is there as shown below. So to overcome tha

deeplearning.ai

Object Detection

Bounding box predictions

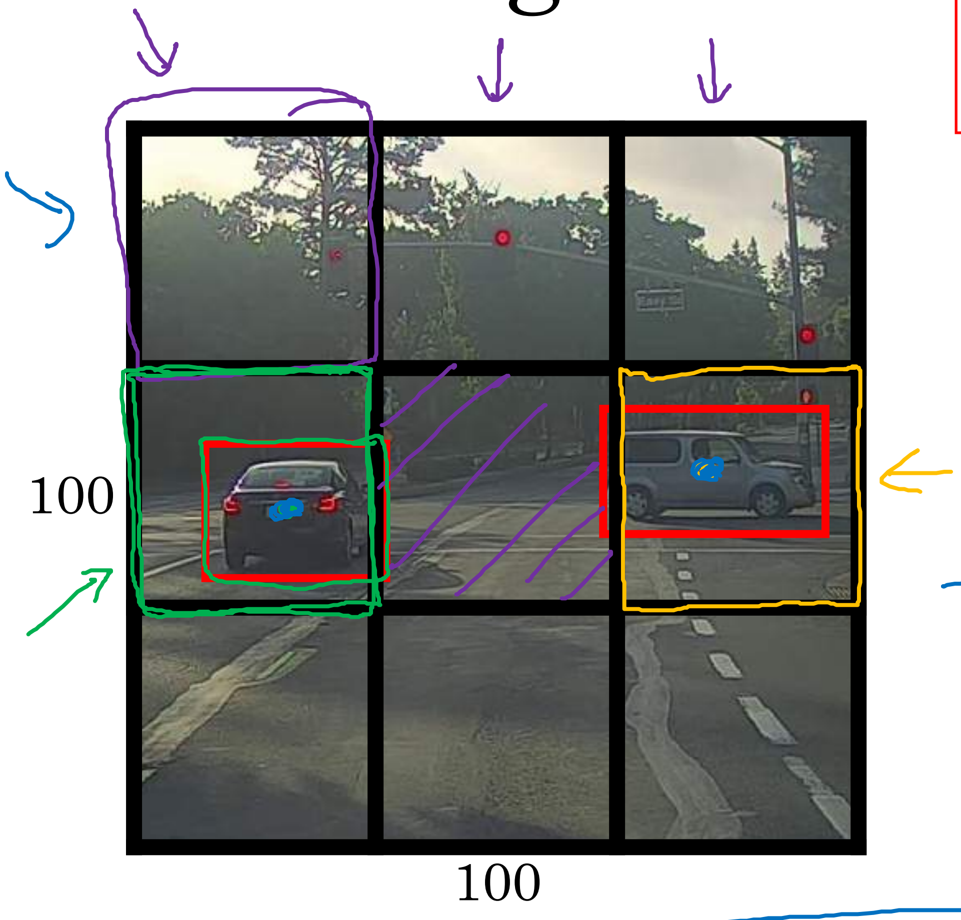
Output accurate bounding boxes



Andrew Ng

YOLO algorithm

We will get a 8x1 output for every square box , so in total we will get 3x3x8 output. We will train



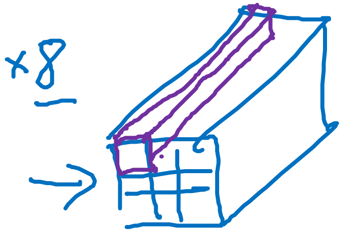
Labels for training

For each grid cell:

$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \quad \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix} \quad \begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Target output:

$3 \times 3 \times 8$



X (100x100x3) → CONV → MAXPOOL → .... → 3x3x8 y

361

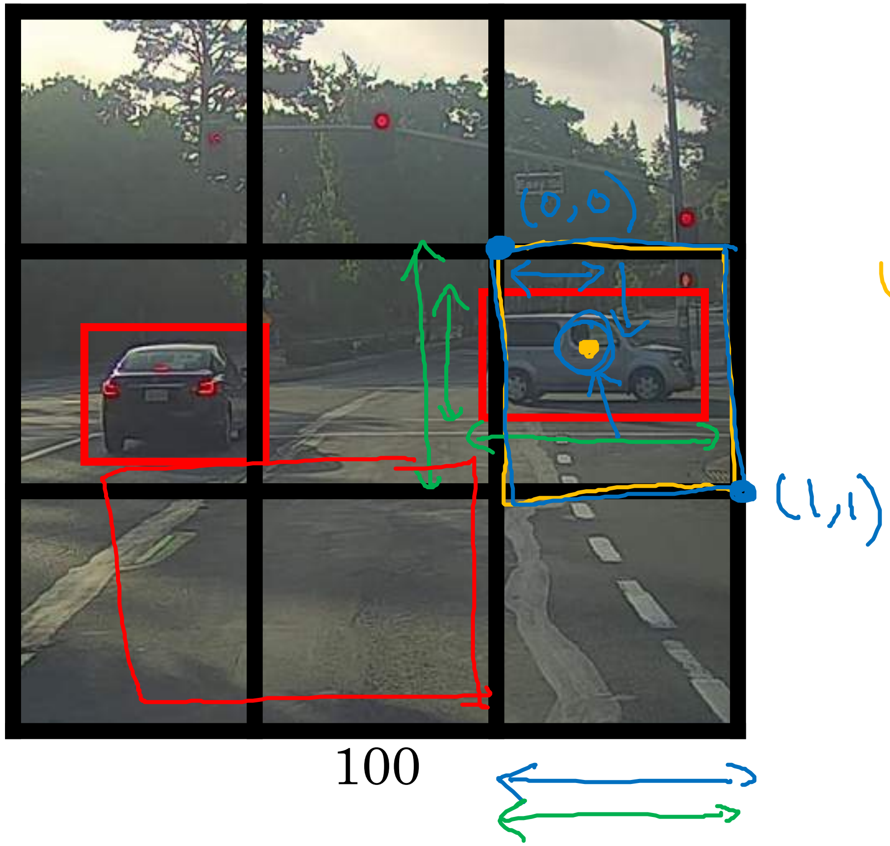
→ 19x19x8

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

Andrew Ng

Specify the bounding boxes

100



100

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

0.4
0.3 } between 0 and 1
0.9
0.5 } could be > 1

how to specify the box? We take upper left corner as 0,0 and right b

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

Andrew Ng

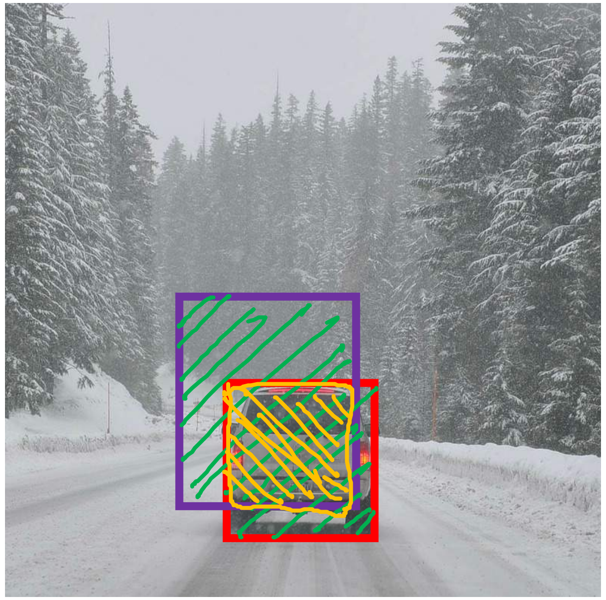
Used to determine how well is our algorithm is doing in detecting the objects. If red is actual and our model predicts purple box, then we take ratio of inte

deeplearning.ai

Object Detection

Intersection over union

Evaluating object localization



Intersection over Union (IoU)

$$= \frac{\text{size of}}{\text{size of}}$$

"Correct" if IoU $\geq \boxed{0.5}$

0.6

More generally, IoU is a measure of the overlap between two bounding boxes.

Andrew Ng

deeplearning.ai

Object Detection

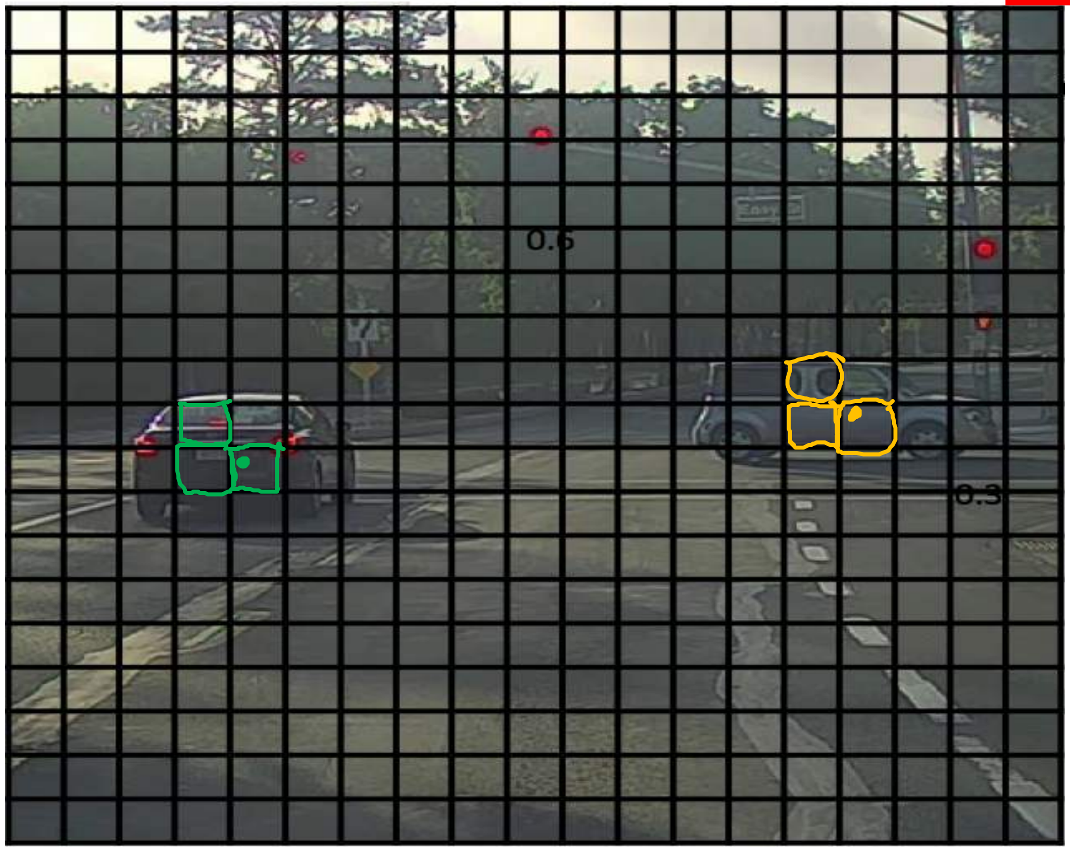
Non-max suppression

Non-max suppression example

To avoid detecting same object twice we

Andrew Ng

Non-max suppression example

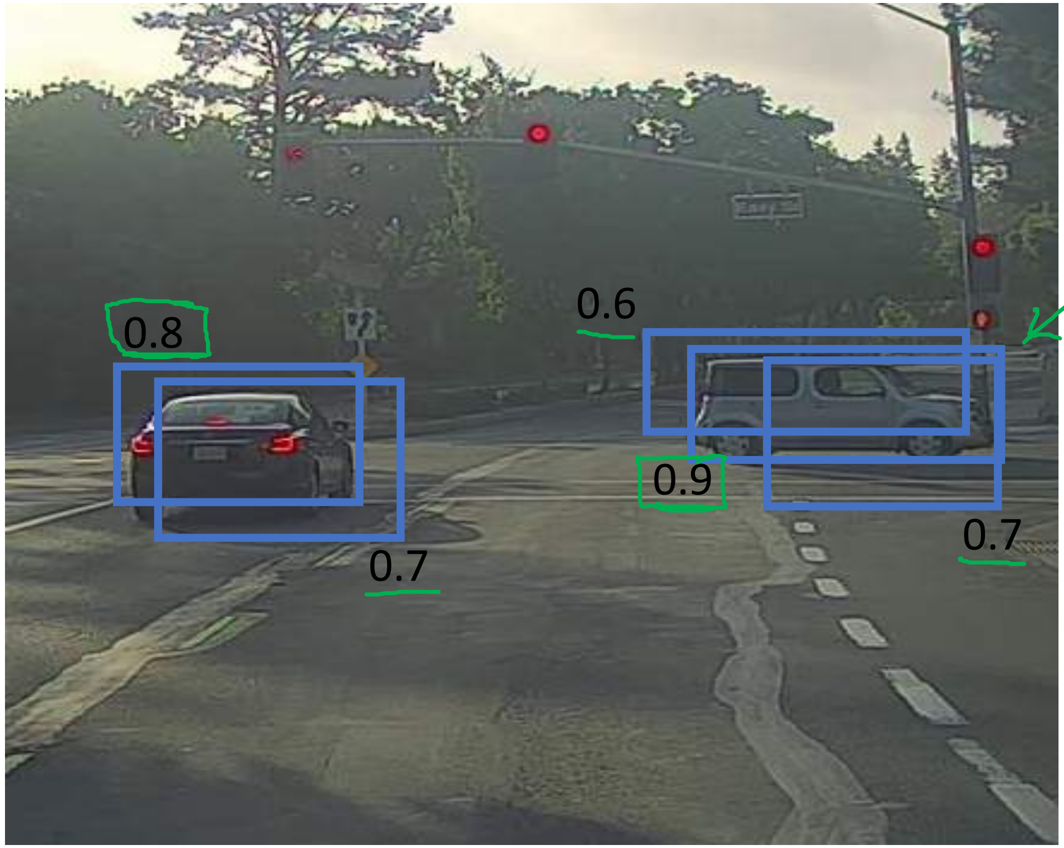


19x19

the real centre is at the marked position bu

Andrew Ng

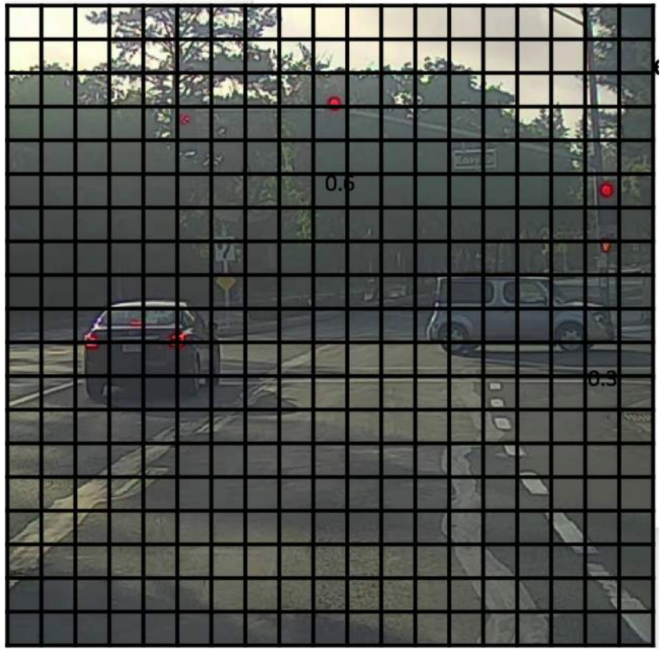
Non-max suppression example



p_c

Andrew Ng

Non-max suppression algorithm



19×19

Each output prediction is: $\begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \end{bmatrix}$

c_1
c_2
c_3

Discard all boxes with $p_c \leq 0.6$

While there are any remaining boxes:

- Pick the box with the largest p_c
  Output that as a prediction.
- Discard any remaining box with
  IoU ≥ 0.5 with the box output
  in the previous step

Jo pichle mein finally box chuna usse IOU nikalo au

Andrew Ng

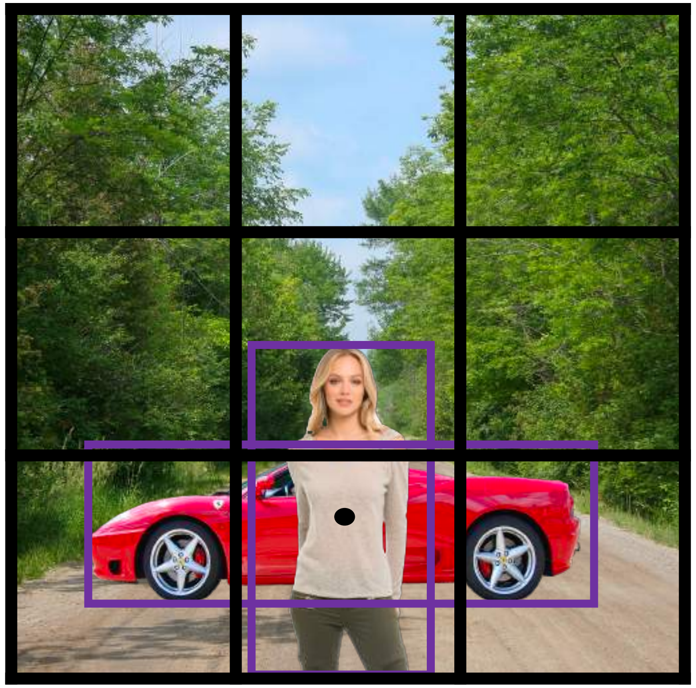
What if our square box contains more than 1 object. In that case anchor boxes helps. We define 2 anchor boxes of different orientation and for each sqar

deeplearning.ai

Object Detection

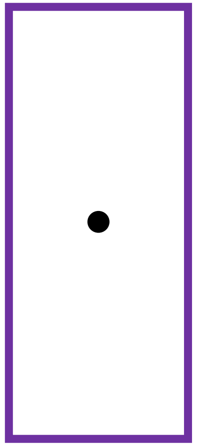
Anchor boxes

Overlapping objects:

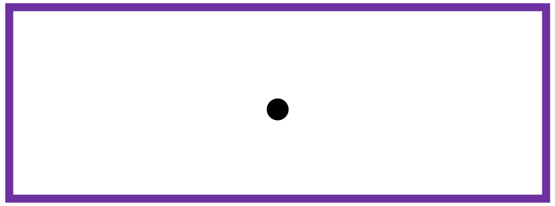


$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

Anchor box 1:



Anchor box 2:



$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \\ p_c \\ b_x \\ \vdots \\ c_3 \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix}\right\} \text{anchor box 1} \\ \left.\begin{matrix} \\ \\ \\ \\ \end{matrix}\right\} \text{anchor box 2} \end{matrix}$$

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

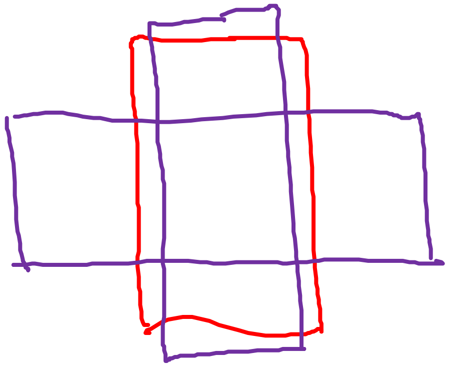
Anchor box algorithm

Previously:

Each object in training image is assigned to grid cell that contains that object's midpoint.

Output y:

$3 \times 3 \times 8$



With two anchor boxes:

Each object in training image is assigned to grid cell that contains object's midpoint and anchor box for the grid cell with highest IoU.

(grid cell, anchor box)

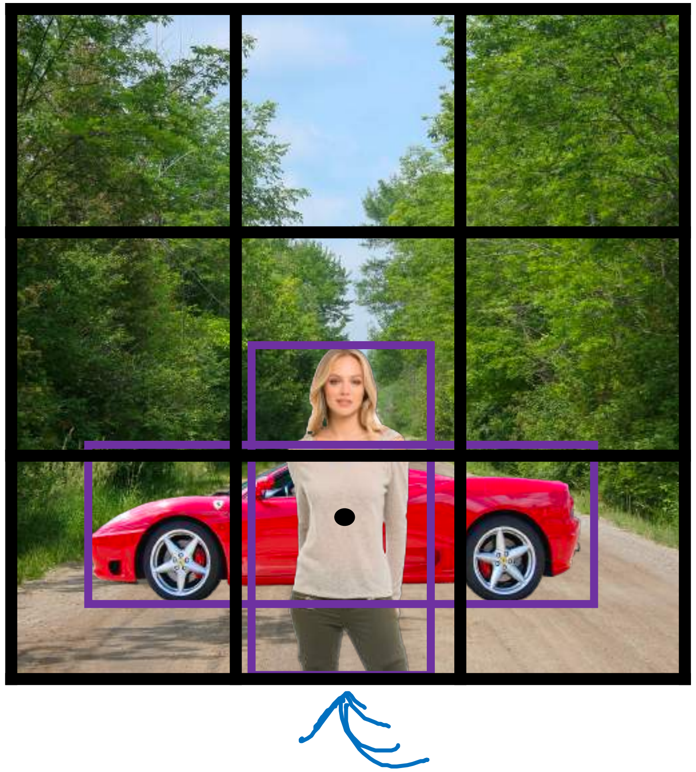
Output y:

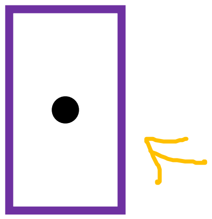
$3 \times 3 \times 16$

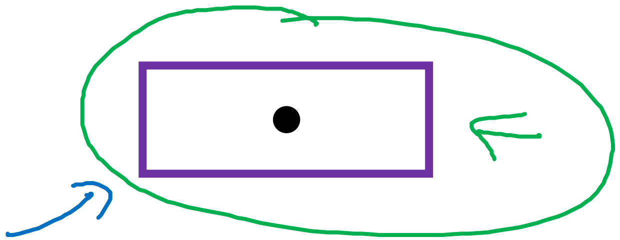
$3 \times 3 \times 2 \times 8$

Andrew Ng

Anchor box example



Anchor box 1: 

Anchor box 2: 

car only?

$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \\ p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \quad \begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 1 \\ 0 \\ 0 \\ 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix}\right\} \text{anchor box 1} \\ \left.\begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix}\right\} \text{anchor box 2} \end{matrix}$$

Andrew Ng

deeplearning.ai

Object Detection

Putting it together: YOLO algorithm

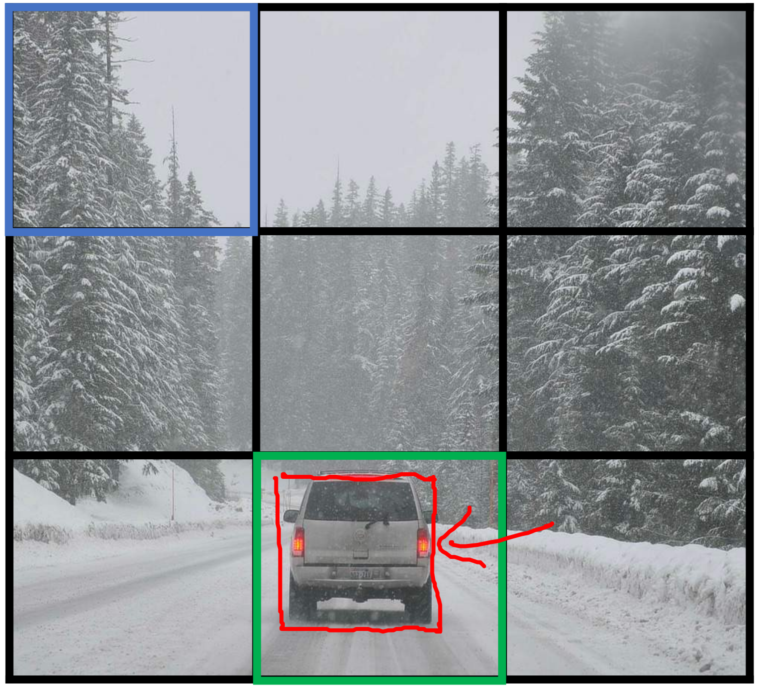
Training

Ye to bas ek hai ese hi 3x3 honge har ek square k

1 - pedestrian

2 - car

3 - motorcycle



If we are using 2 anchor boxes then our ouptut

$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \\ p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \quad \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix} \quad \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

1 2

3 × 3 × 16

y is 3×3×2×8

19×19×16

19×19×40

#anchors

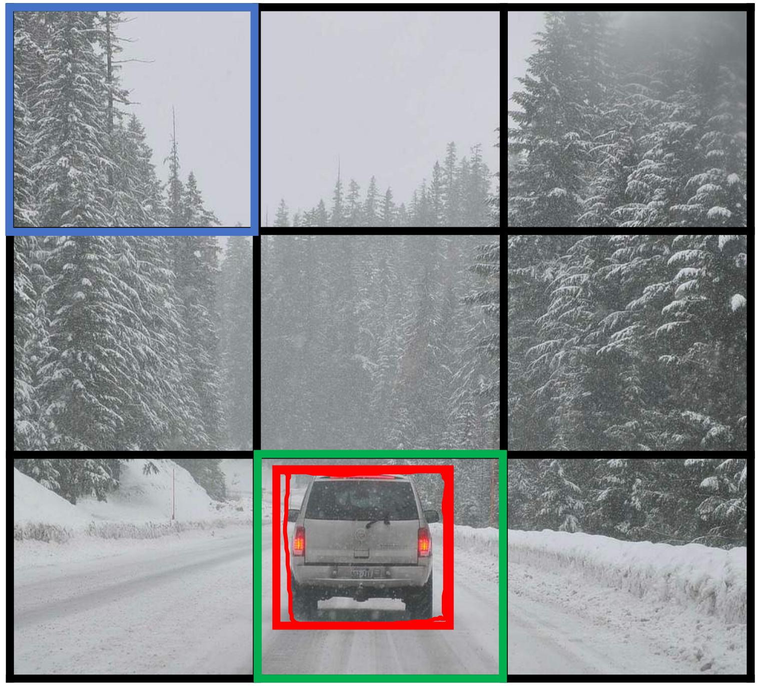
5 + #classes

100×100×3 → ConvNet → 3×3×16

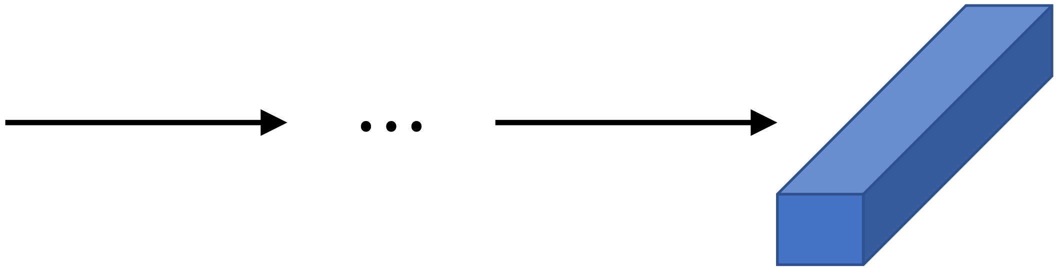
Andrew Ng

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

Making predictions



our model take picture and convert it to 3x3x2x8 which will be used

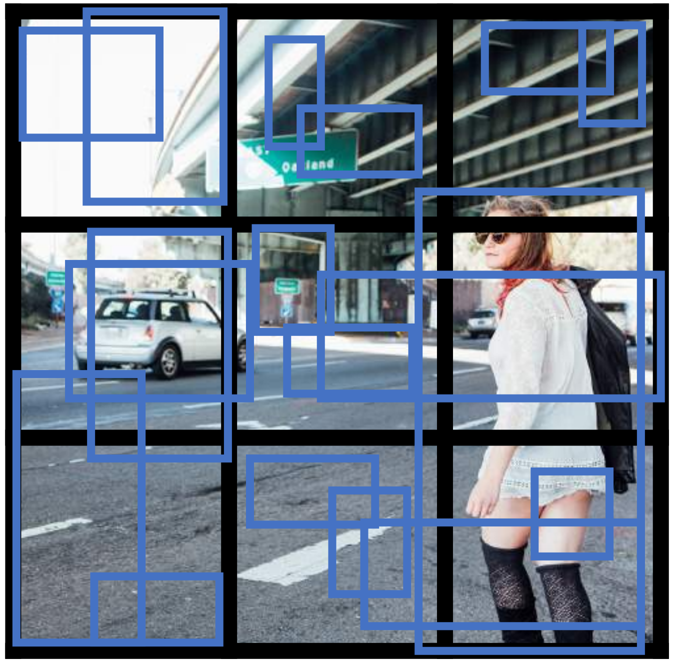


$3 \times 3 \times 2 \times 8$

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

Andrew Ng

Outputting the non-max supressed outputs



- For each grid call, 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.

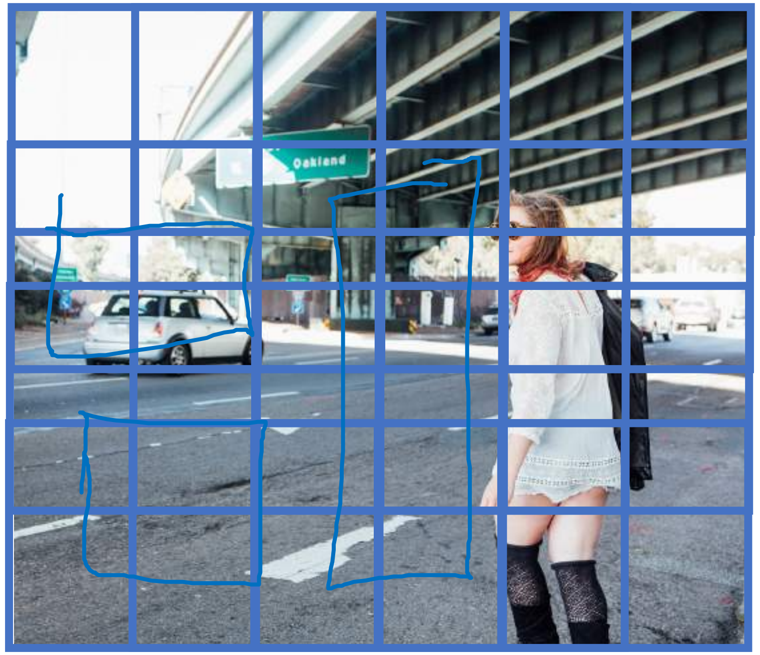
Instead of running sliding window everywhere where there are clearly no object we run our algo only on some regions of interest where objects can be

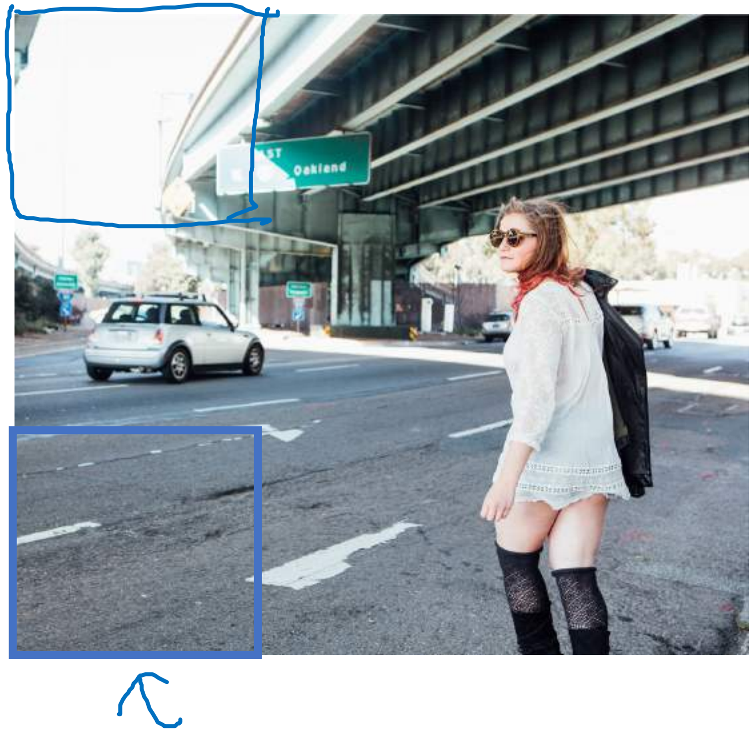
deeplearning.ai

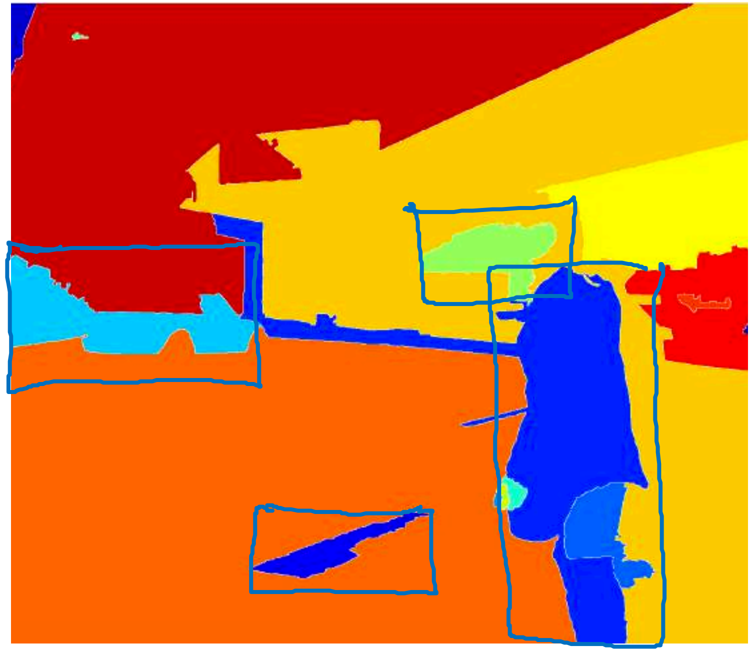
Object Detection

Region proposals (Optional)

Region proposal: R-CNN







Segmentation algorithm

~2,000

[Girshik et. al, 2013, Rich feature hierarchies for accurate object detection and semantic segmentation] Andrew Ng

Faster algorithms

R-CNN: Propose regions. Classify proposed regions one at a time. Output label + bounding box.

Fast R-CNN: Propose regions. Use convolution implementation of sliding windows to classify all the proposed regions.

Faster R-CNN: Use convolutional network to propose regions.

[Girshik et. al, 2013. Rich feature hierarchies for accurate object detection and semantic segmentation]
[Girshik, 2015. Fast R-CNN]
[Ren et. al, 2016. Faster R-CNN: Towards real-time object detection with region proposal networks]

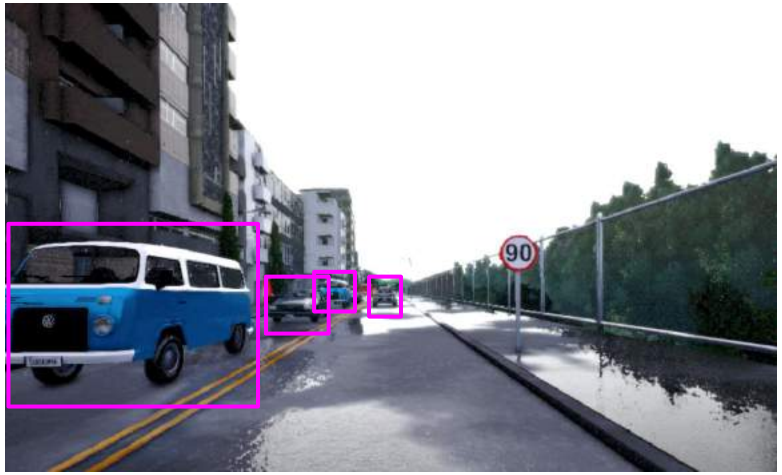
deeplearning.ai

Convolutional Neural Networks

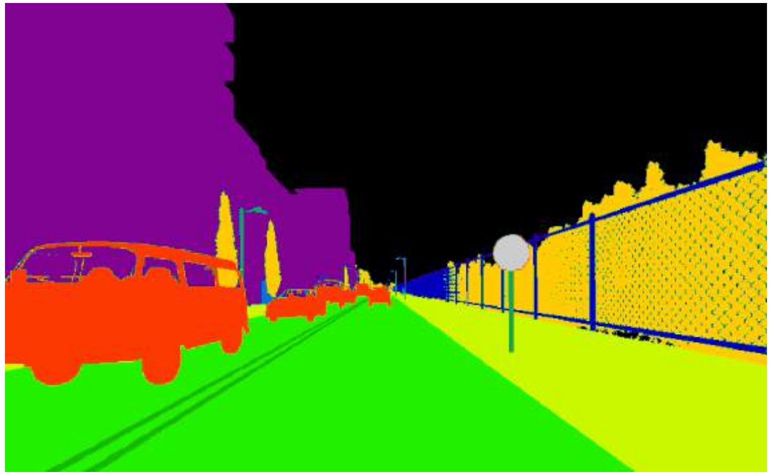
Semantic segmentation with U-Net

Object Detection vs. Semantic Segmentation

Input image



Object Detection

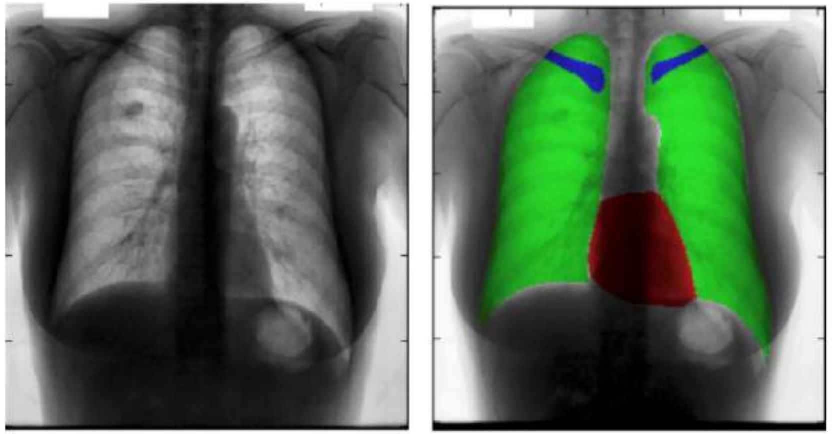


Semantic Segmentation

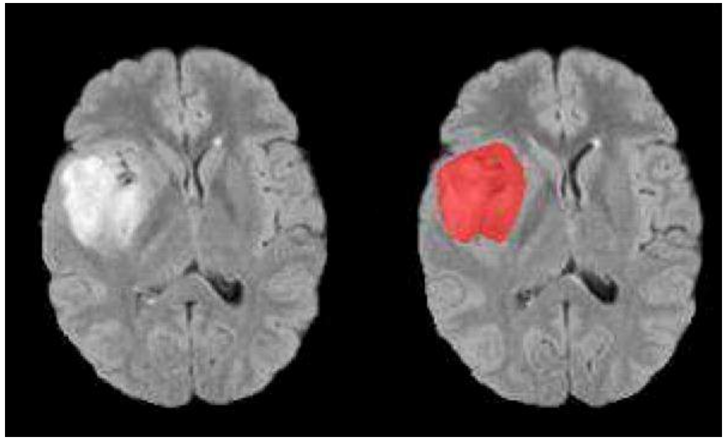
what segmentatic does it tries to find out what each pixel is doing. Where, for example, rather than detectingthe road and trying to drawa bour

Andrew Ng

Motivation for U-Net



Chest X-Ray



Brain MRI

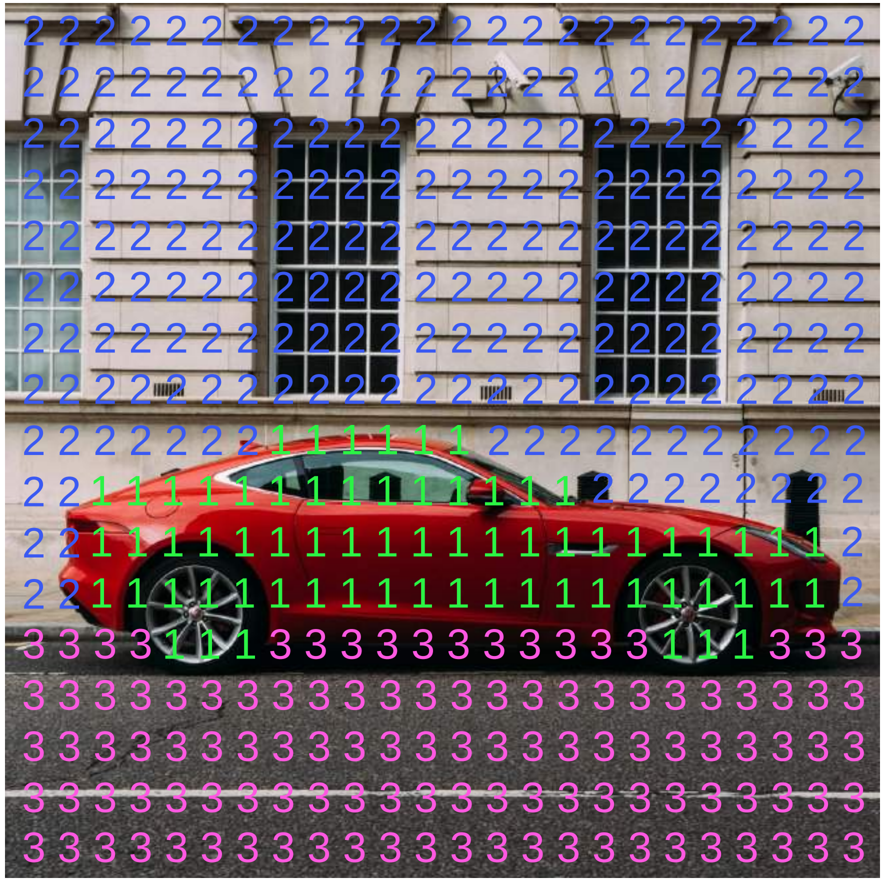
[Novikov et al., 2017, Fully Convolutional Architectures for Multi-Class Segmentation in Chest Radiographs]
[Dong et al., 2017, Automatic Brain Tumor Detection and Segmentation Using U-Net Based Fully Convolutional Networks]

Per-pixel class labels

We want 0/1 for every pixel.

1. Car
0. Not Car

Per-pixel class labels

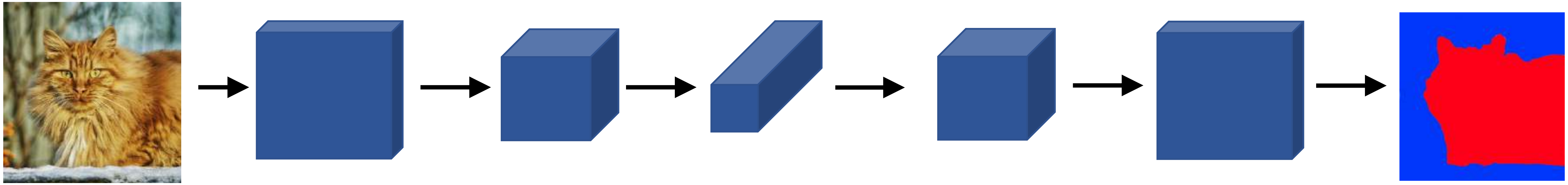


1. Car
2. Building
3. Road

2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2
2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2
2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2
2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2
3	3	3	3	1	1	1	3	3	3	3	3	3	3	3	3	3	3	1	1	1	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3

Segmentation Map

Deep Learning for Semantic Segmentation



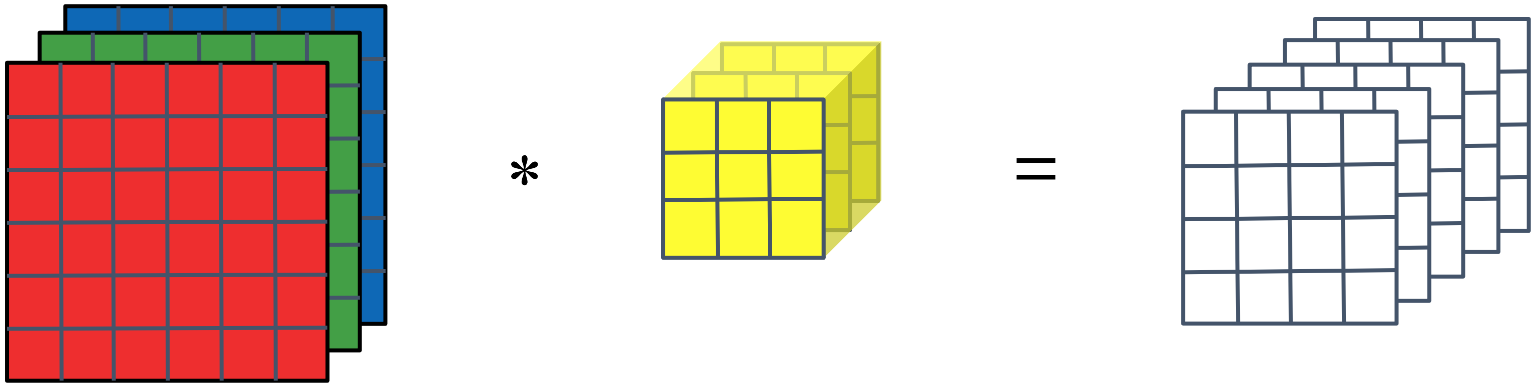
Input image, . one key point in semantic segmentation is that as we progress len and width goes decreasing and depth keeps on increasing. So

Andrew Ng

Transpose convolution helps us to make the input image blow up to higher shape and dimension. like blowing up 2x2 image to a 4x4 image.

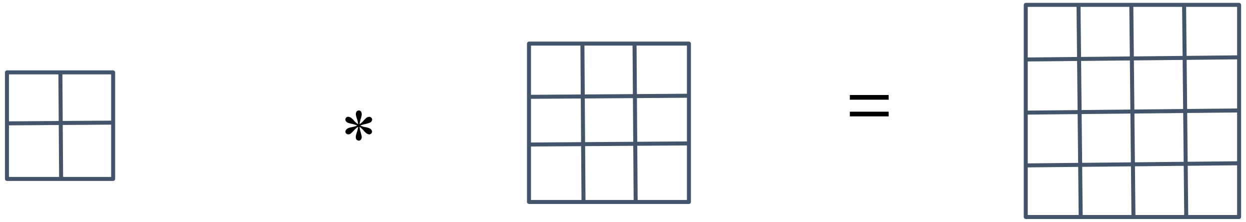
Transpose Convolution

Normal Convolution



Transpose Convolution

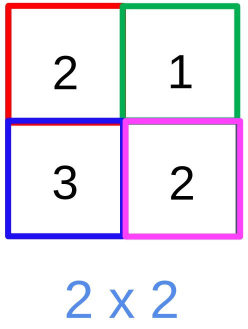
Output bigger than the input.

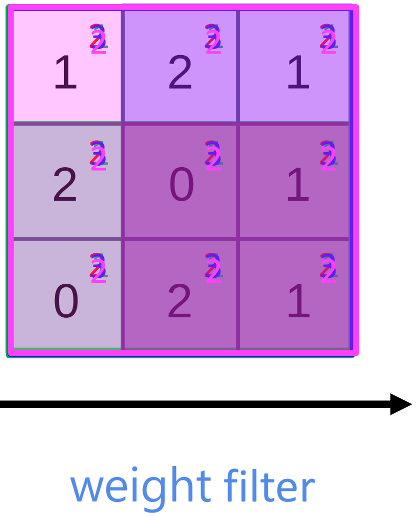


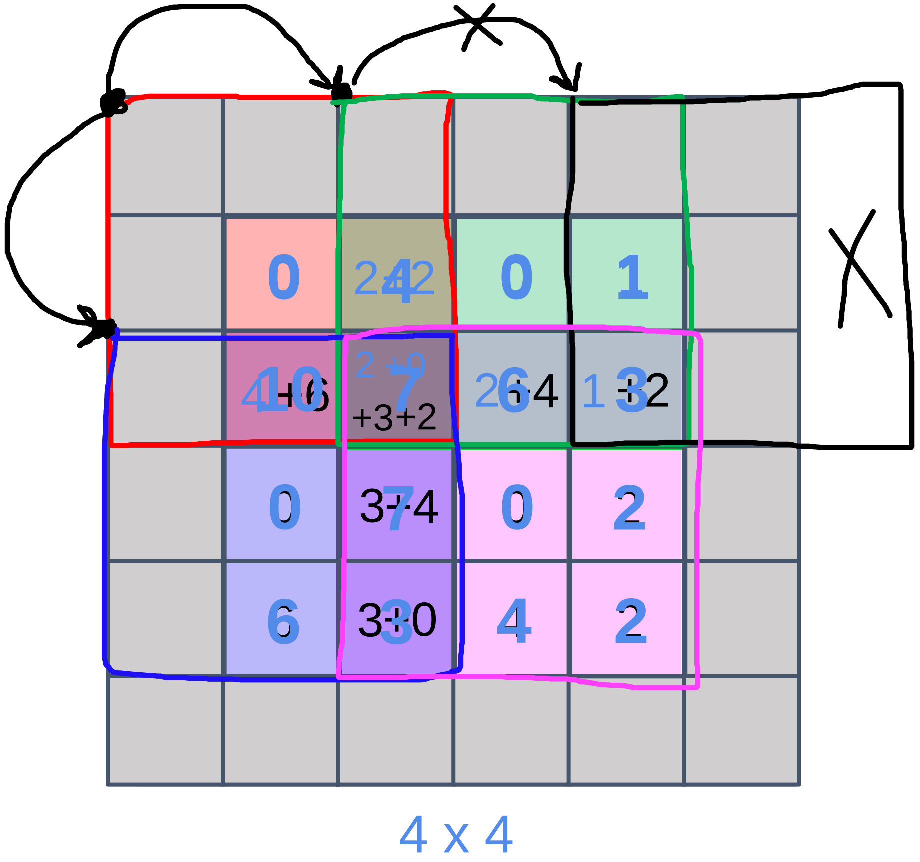
Andrew Ng

In the regular convolution, you would take the filter and place it on top of the inputs and then multiply and sum up. In the transpose convolution, instead o

Transpose Convolution





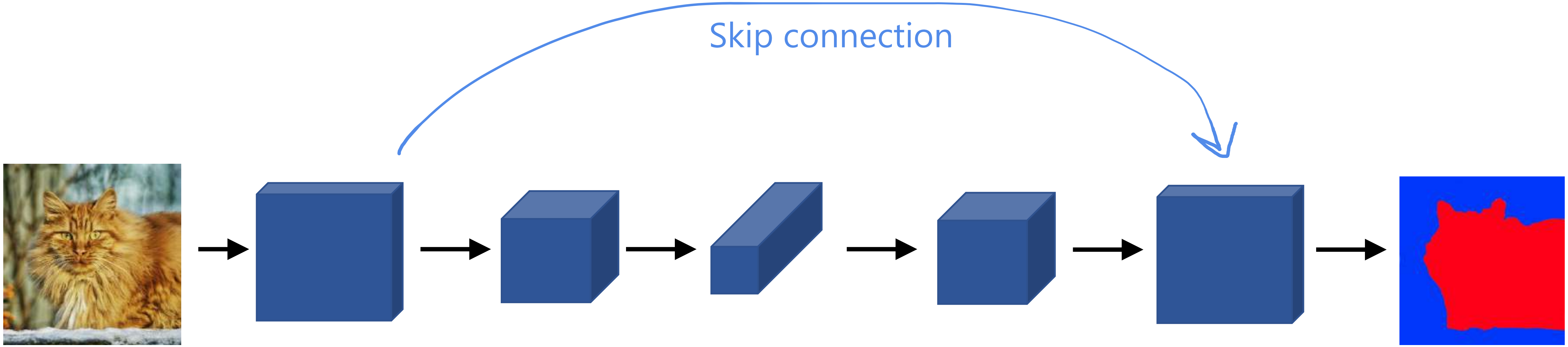


filter f x f = 3 x 3 padding p = 1 stride s = 2

Padding applied to the out

Andrew Ng

Deep Learning for Semantic Segmentation



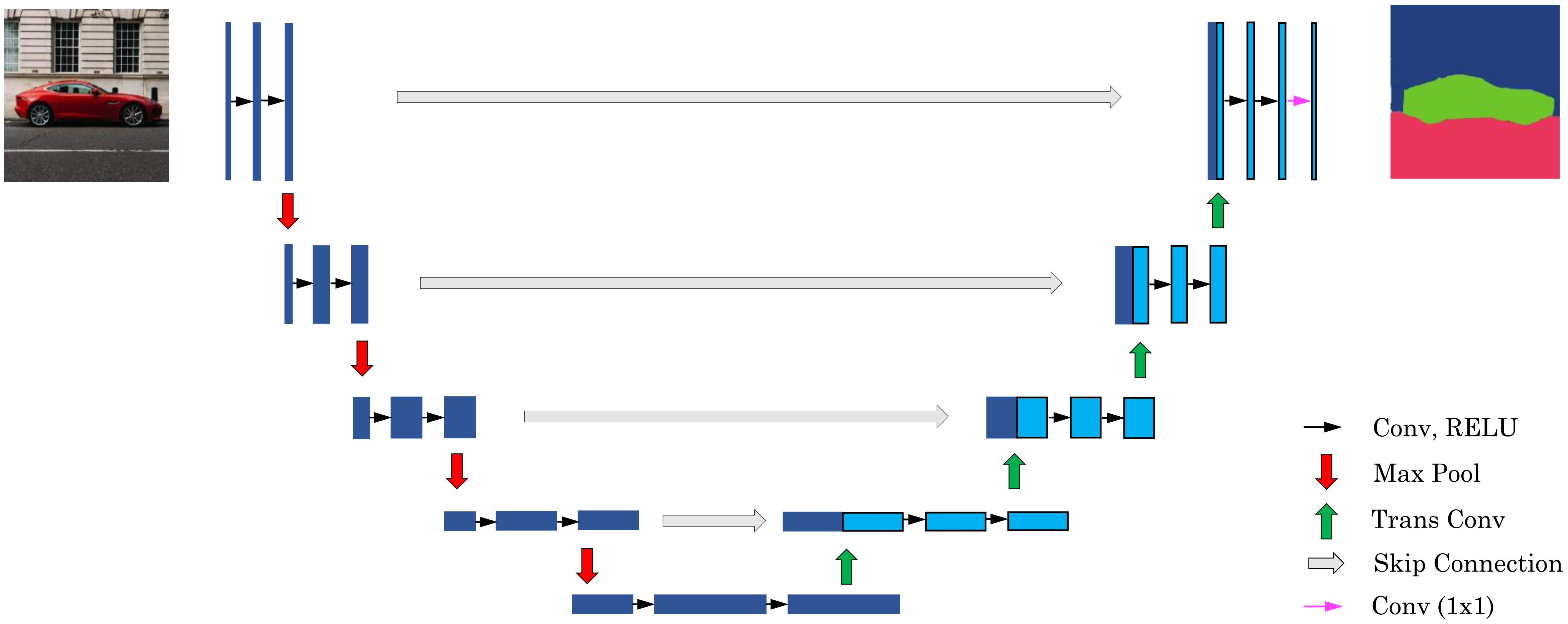
For the first part we will be using the norn

Second part uses trasnspose convultioin to blov

Andrew Ng

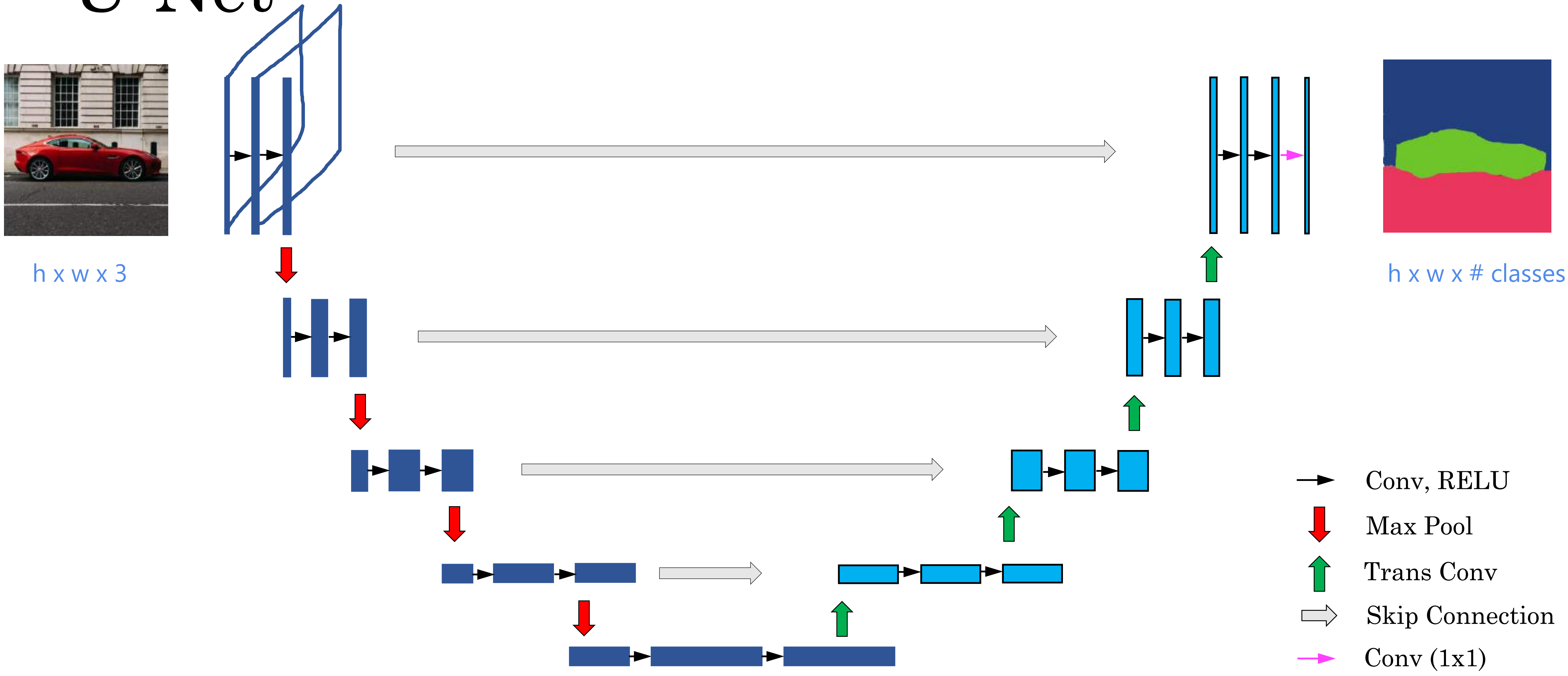
U-Net

See it looks like a U and that's why callled u-net.



[Ronneberger et al., 2015, U-Net: Convolutional Networks for Biomedical Image Segmentation]

U-Net



[Ronneberger et al., 2015, U-Net: Convolutional Networks for Biomedical Image Segmentation]

Andrew Ng