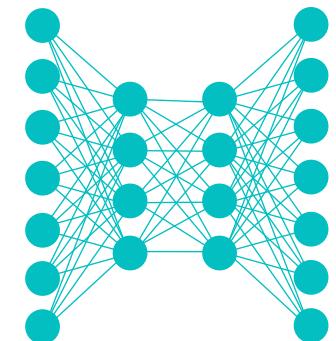


# Lecture Notes for **Neural Networks** **and Machine Learning**



Fully Convolutional Learning II:  
Object Detection



# Object Detection with YOLO



**Joseph Redmon** @pjreddie • Feb 20, 2020



"We shouldn't have to think about the societal impact of our work because it's hard and other people can do it for us" is a really bad argument.



**Roger Grosse** @RogerGrosse

Replying to @kevin\_zakkia and @hardmaru

To be clear, I don't think this is a positive step. Societal impacts of AI is a tough field, and there are researchers and organizations that study it professionally. Most authors do not have expertise in the area and won't do good enough scholarship to say something meaningful.



**Joseph Redmon**  
@pjreddie

I stopped doing CV research because I saw the impact my work was having. I loved the work but the military applications and privacy concerns eventually became impossible to ignore.



**Roger Grosse** @RogerGrosse

Replying to @skoularidou

What's an example of a situation where you think someone should decide not to submit their paper due to Broader Impacts reasons?

10:09 AM · Feb 20, 2020



related, but different approach

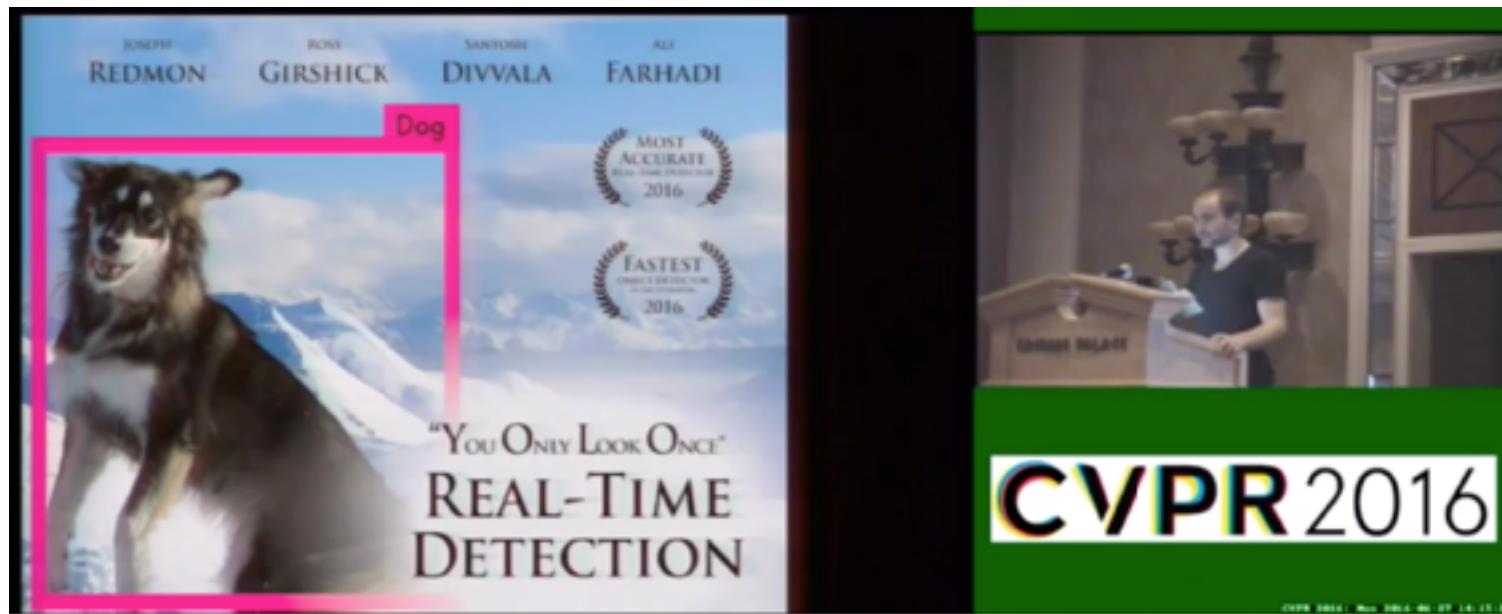


• e researchers started working on a



# Don't want to listen to me explain it?

- Check out Joseph Redmon's Talk at CVPR 2016:
  - <https://www.youtube.com/watch?v=NM6lrxy0bxS>
  - This is how you give a Technical presentation, mastery of presentation with technical depth



# 2018: Everybody has a Gimmick



- YOLO ~40-60 FPS
- Slightly more Accurate than **Faster R-CNN**

## YOLO9000: Better, Faster, Stronger

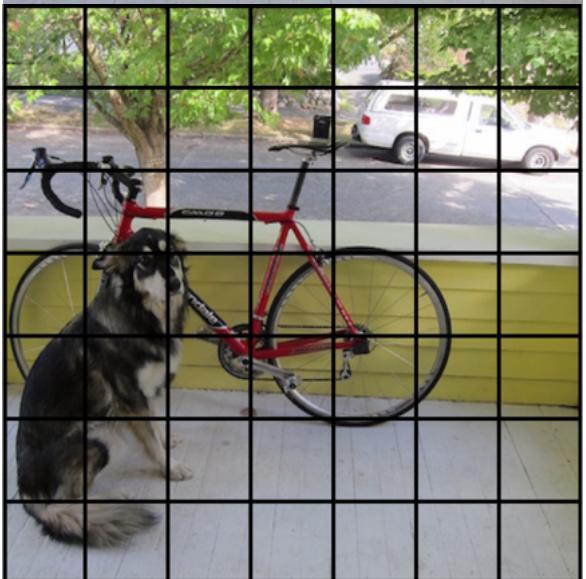
Joseph Redmon<sup>\*†</sup>, Ali Farhadi<sup>\*†</sup>

University of Washington<sup>\*</sup>, Allen Institute for AI<sup>†</sup>

<http://pjreddie.com/yolo9000/>

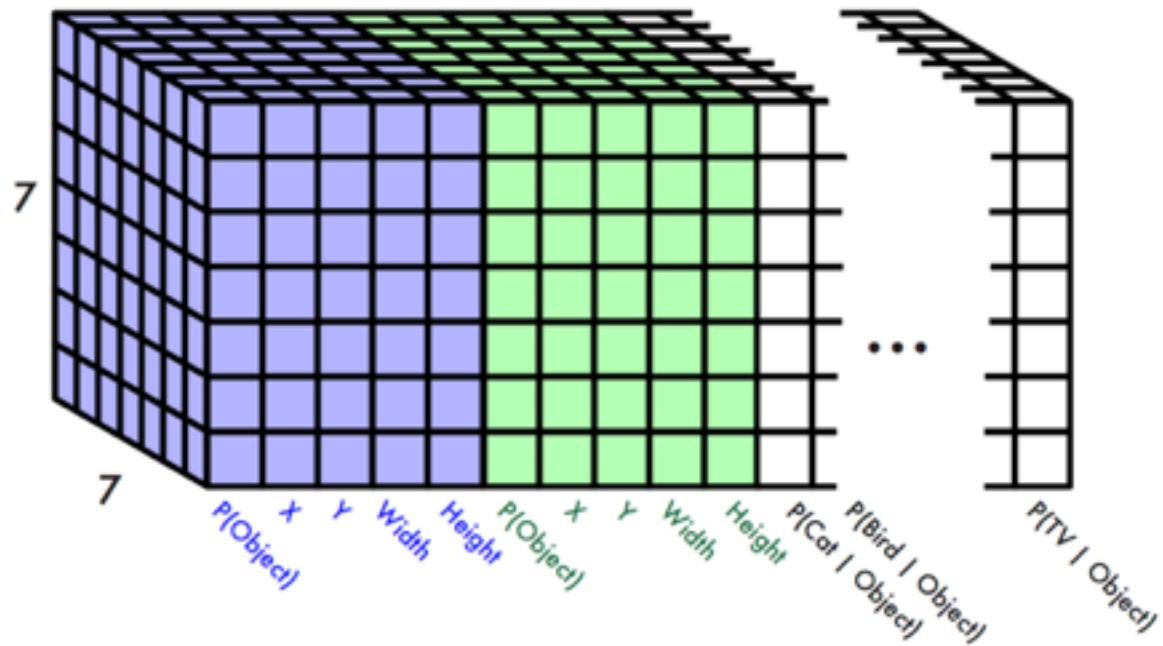


# The YOLO Output Tensor



**The Output Tensor**

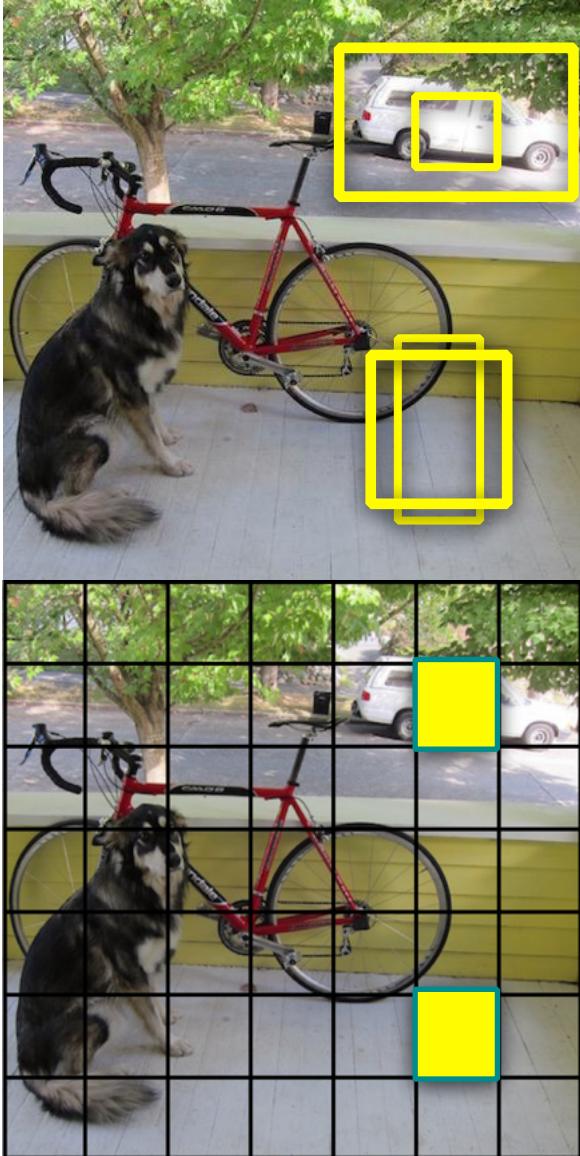
First Bounding Box      Second Bounding Box      Class Probabilities



Redmon and Farhadi, YOLO9000: Better, Faster, Stronger, 2016,  
December 25 — Merry Christmas?

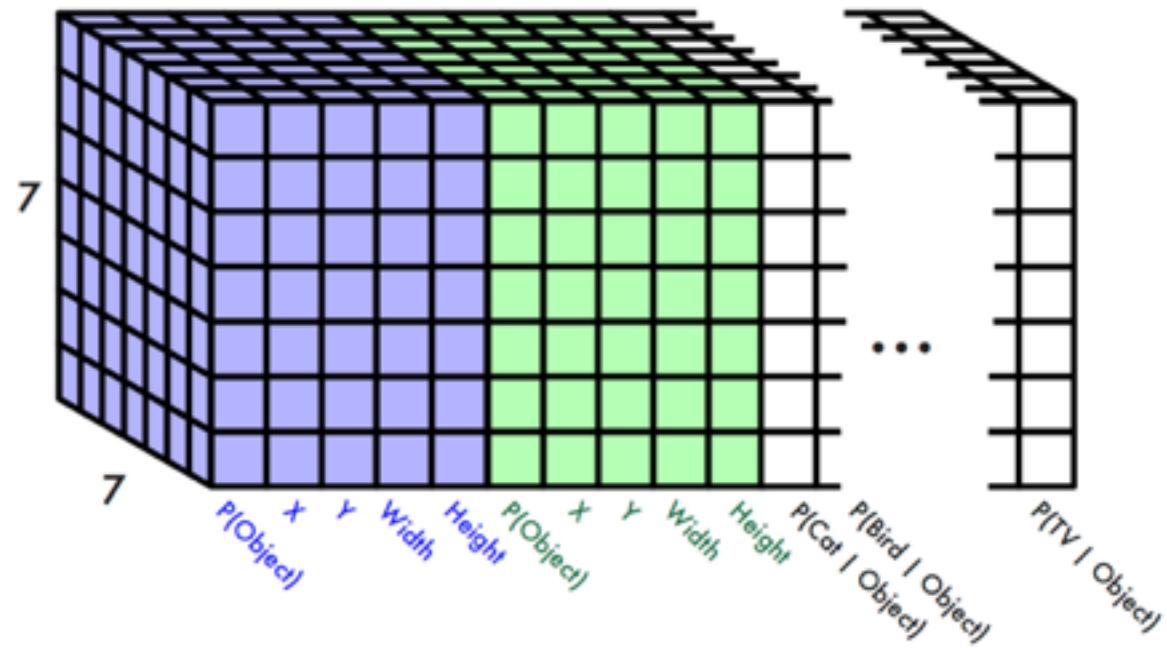


# The YOLO Output Tensor



**The Output Tensor**

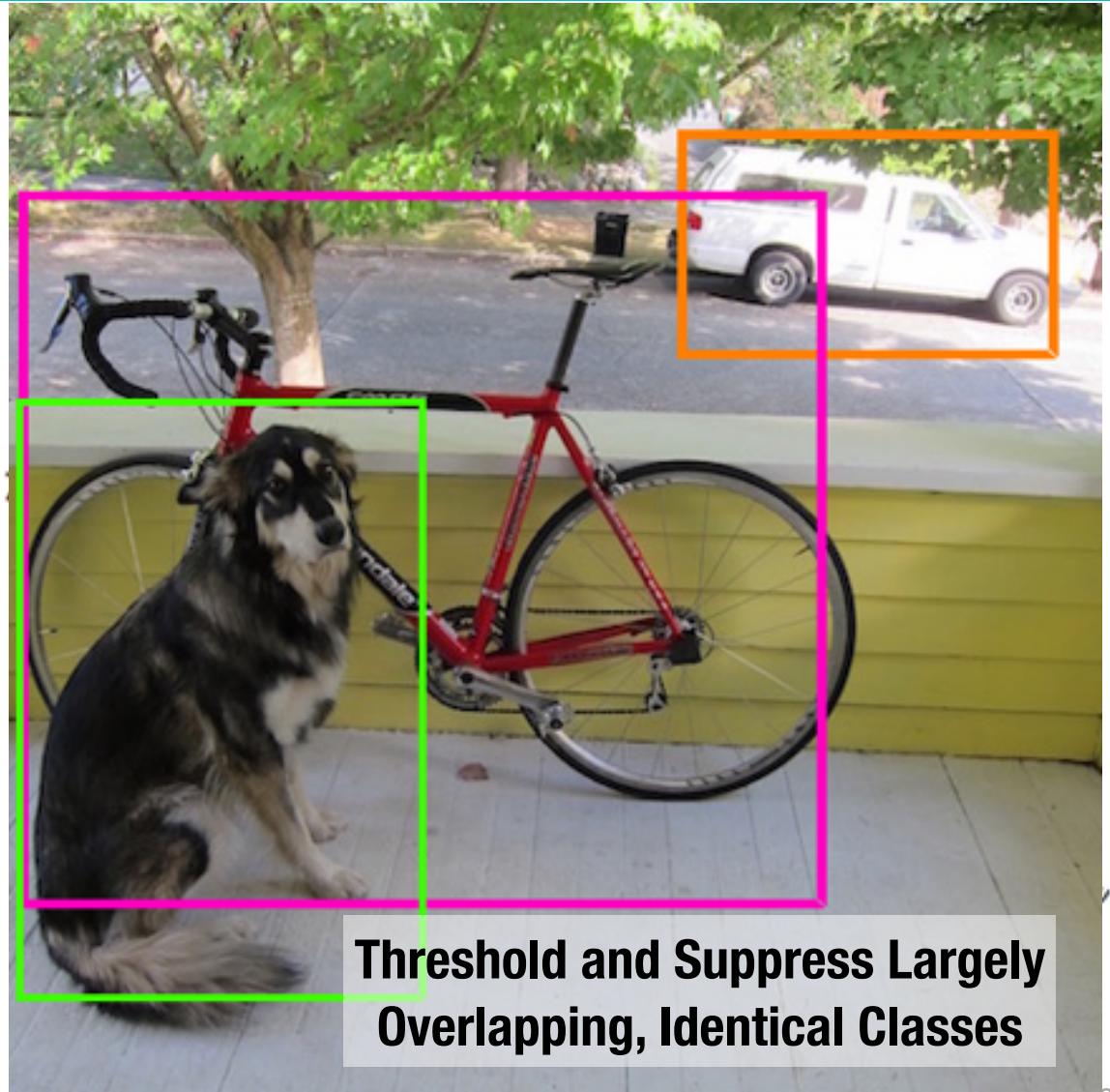
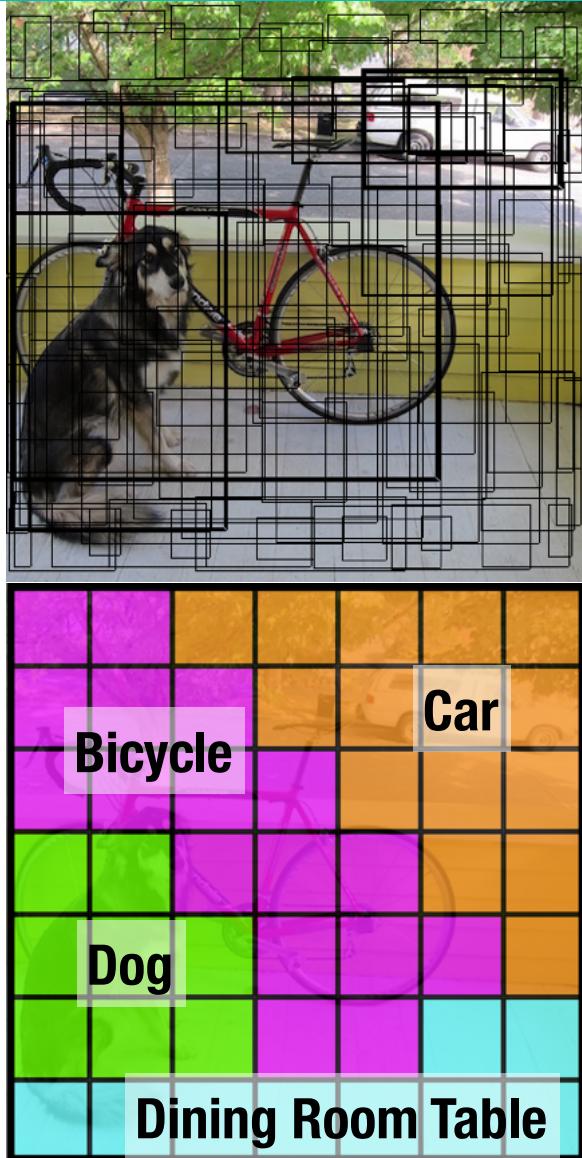
First Bounding Box      Second Bounding Box      Class Probabilities



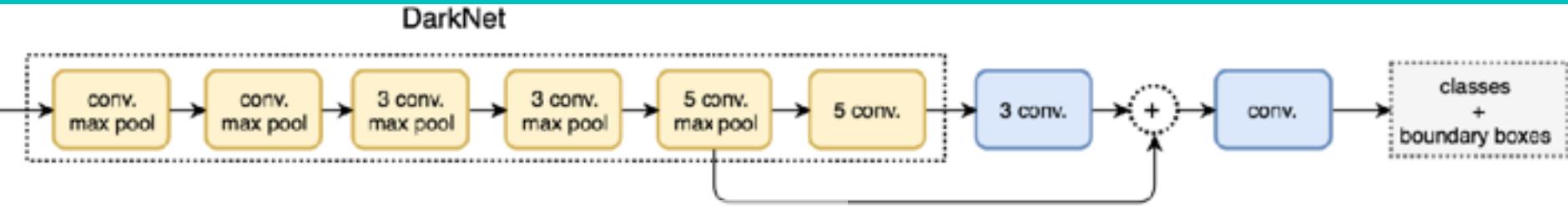
Redmon and Farhadi, YOLO9000: Better, Faster, Stronger, 2016,  
December 25 — Merry Christmas?



# The YOLO Output Tensor



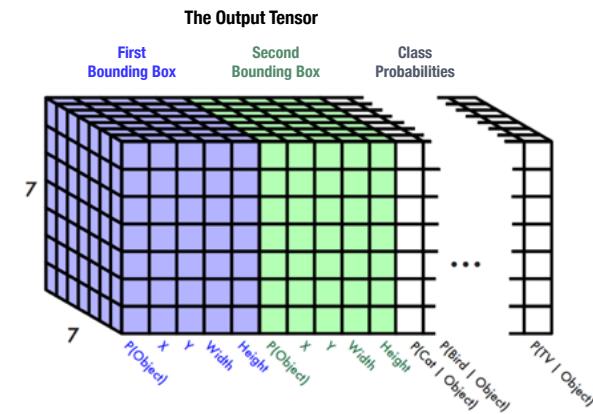
# The YOLO Architecture



**Trained from Traditional Image Dataset.  
Architecture usually: DarkNet on ImageNet**

	pjreddie guys one of my beehives died :-(
	guys one of my beehives died :-(
	SELU activation and yolo openimages
	GUYS I THINK MAYBE IT WAS BROKEN ON OPENCV IDK
	YO DAWG, I HEARD YOU LIKE LICENSES
	generate own license, totally legal :verified:

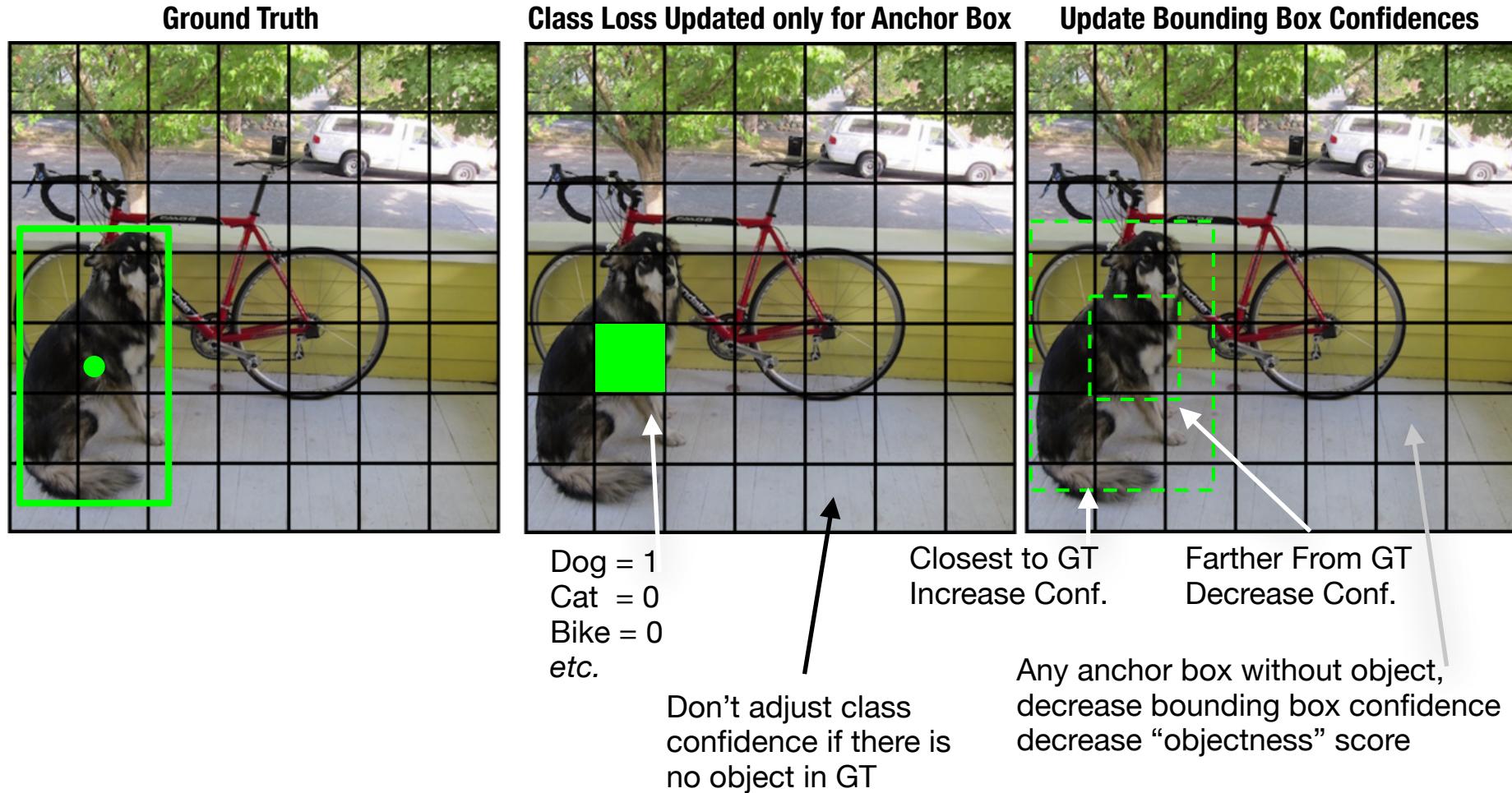
**Last layers:  
Trained on images  
with bounding boxes**



[https://medium.com/@jonathan\\_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088](https://medium.com/@jonathan_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088)

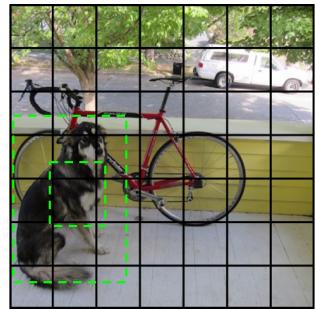


# Training the YOLO Architecture

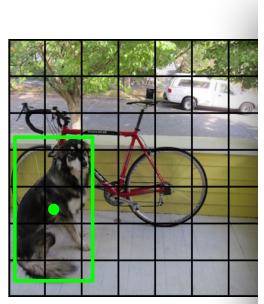


# The YOLO Loss Function

## Update Bounding Box



$$\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[ (x_i - \hat{x}_{ij})^2 + (y_i - \hat{y}_{ij})^2 \right] \\ + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[ (\sqrt{w_i} - \sqrt{\hat{w}_{ij}})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_{ij}})^2 \right]$$



$S \times S$  cells,  $i^{\text{th}}$  cell

$B$  boxes per cell,  $j^{\text{th}}$  box

$\mathbb{1}^{\text{obj}}$  indicator function, from GT

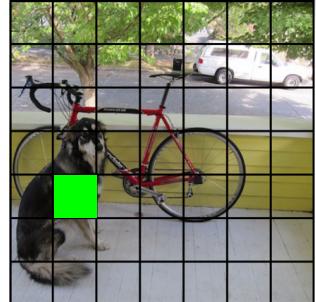
$\hat{C}$  is confidence per box

$\hat{p}(c)$  softmax output, per class

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} (\hat{C}_i - \hat{C}_{ij})^2$$

$$+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} (\hat{C}_i - \hat{C}_{ij})^2$$

## Class Loss



$$+ \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2$$

[https://medium.com/@jonathan\\_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088](https://medium.com/@jonathan_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088)

Localization Loss

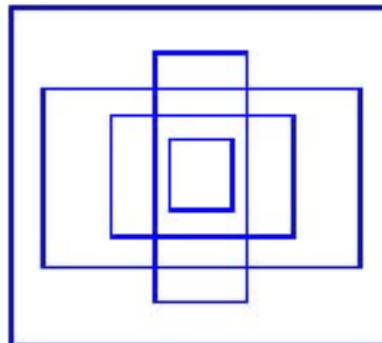
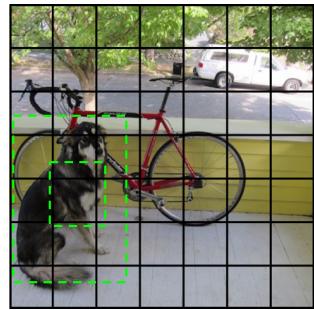
Object Detection Loss

Classification Loss

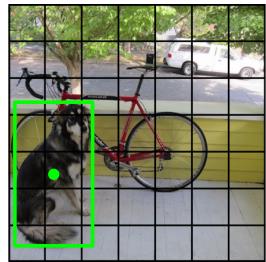


# Updated YOLO Localization (v4)

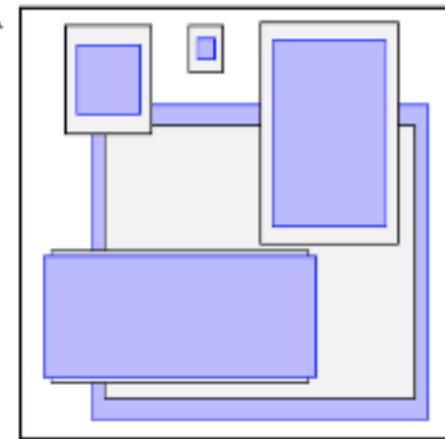
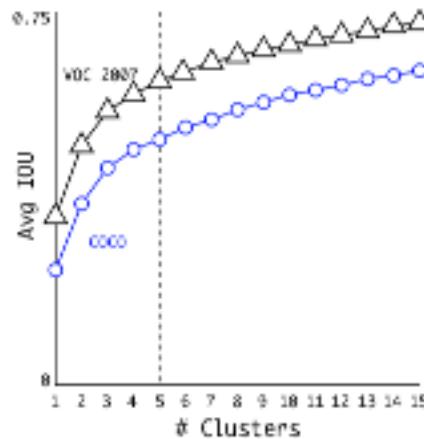
## Update Bounding Box



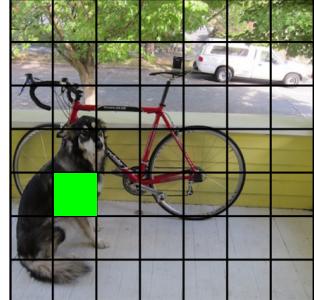
- Define 5 pre-defined box shapes (based on data)
- Regress  $x, y$  offset from cell center
- Bound  $x$  and  $y$  to the bounds of the cell
- And  $s$  scaling param of predefined shape



**“Good” Shape Priors  
Found via Clustering**



## Class Loss



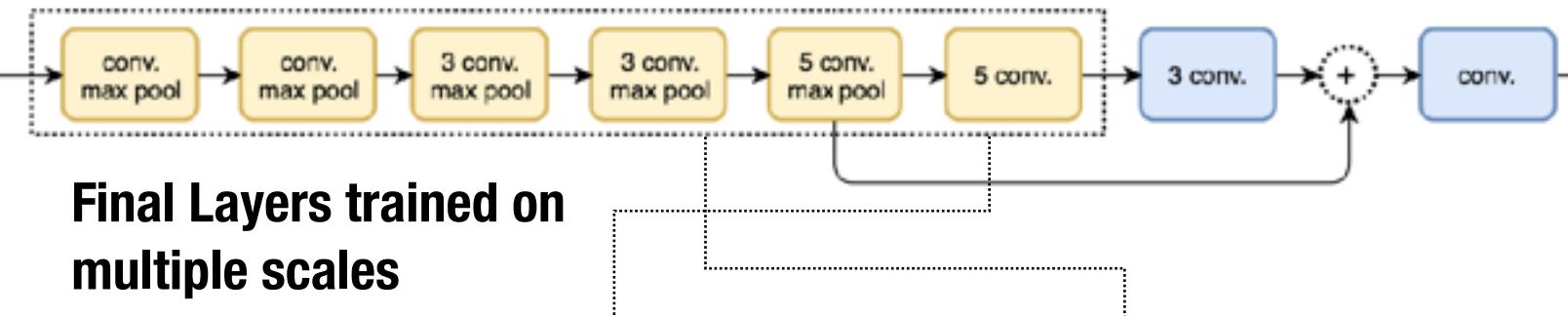
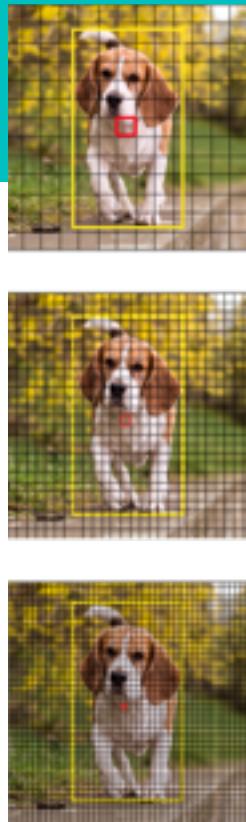
$$\begin{aligned} & + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left( C_i - \hat{C}_{ij} \right)^2 \\ & + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left( C_i - \hat{C}_{ij} \right)^2 + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} \left( p_i(c) - \hat{p}_i(c) \right)^2 \end{aligned}$$

[https://medium.com/@jonathan\\_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088](https://medium.com/@jonathan_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088)

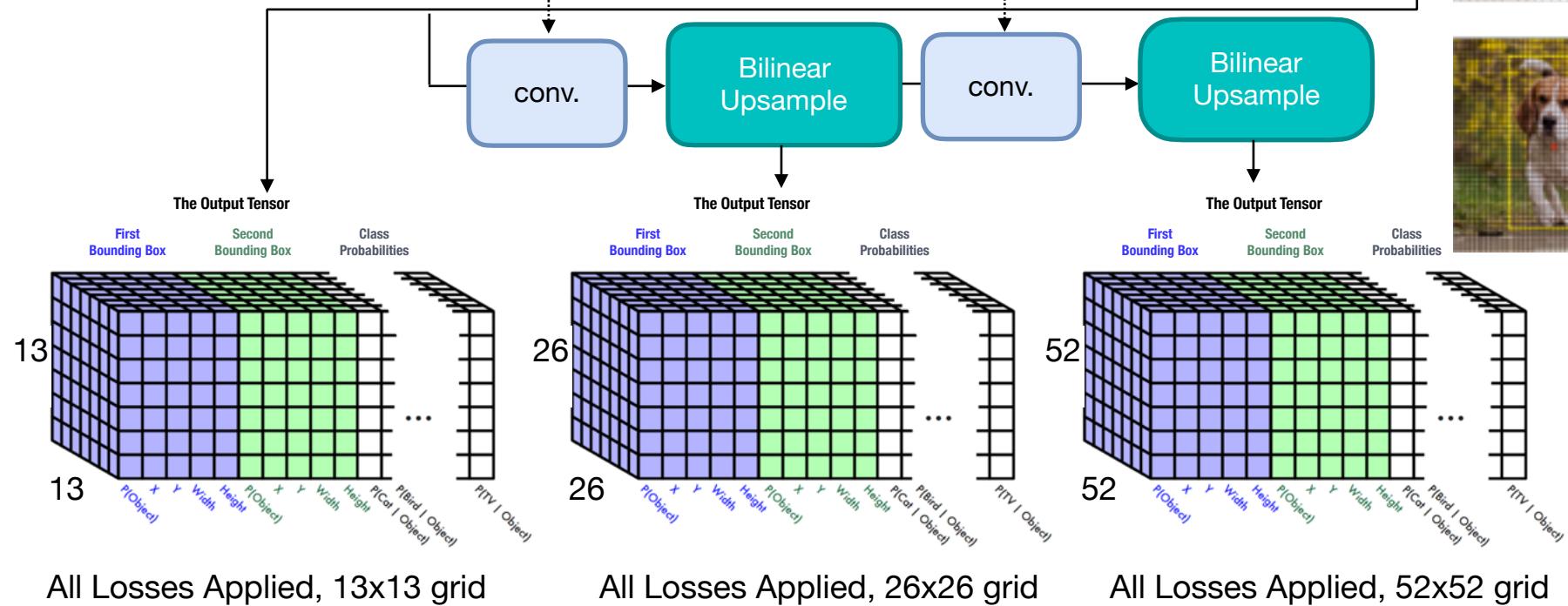
Classification Loss



# The YOLOv3 Architecture



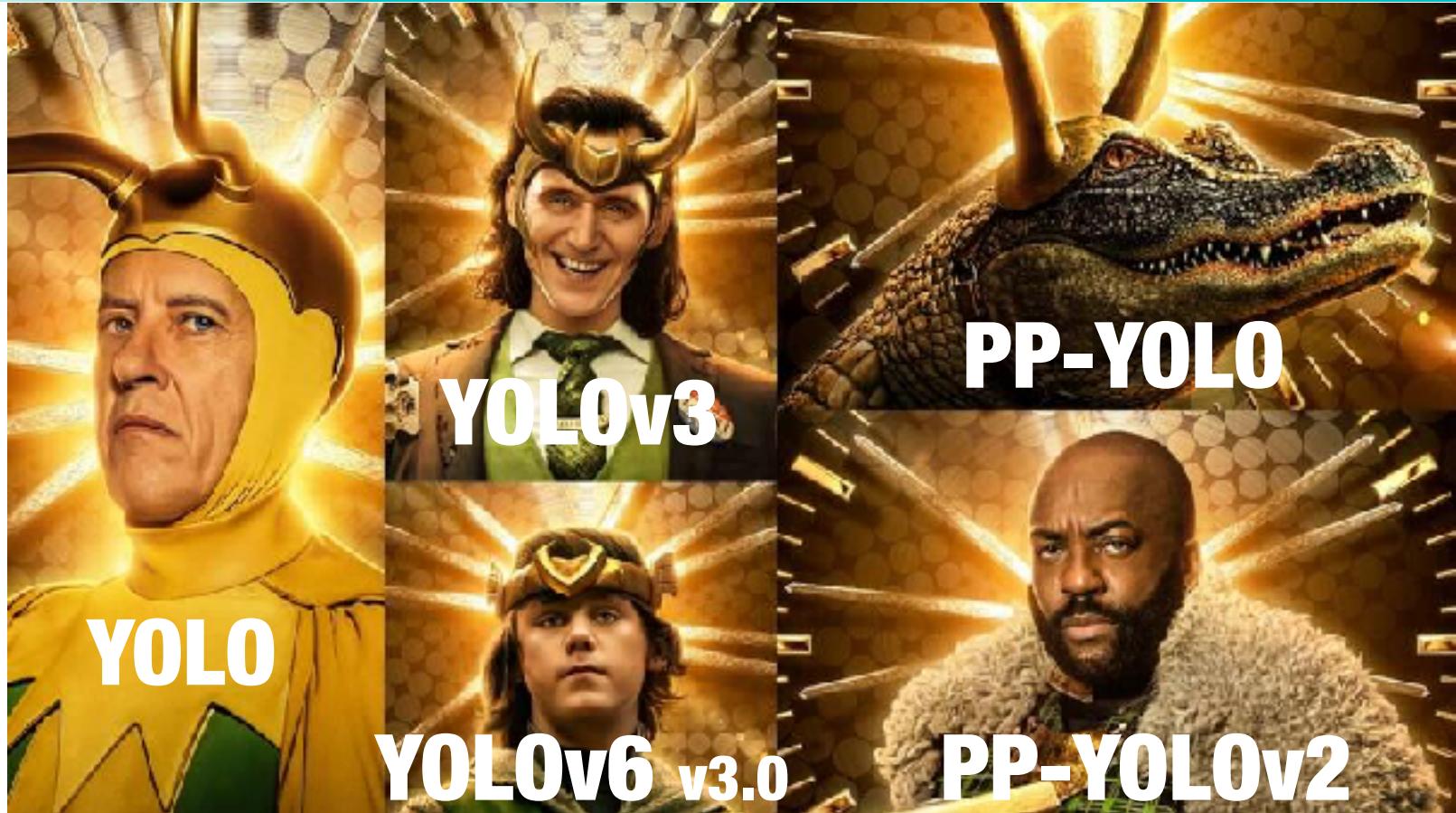
**Final Layers trained on  
multiple scales**



[https://medium.com/@jonathan\\_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088](https://medium.com/@jonathan_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088)

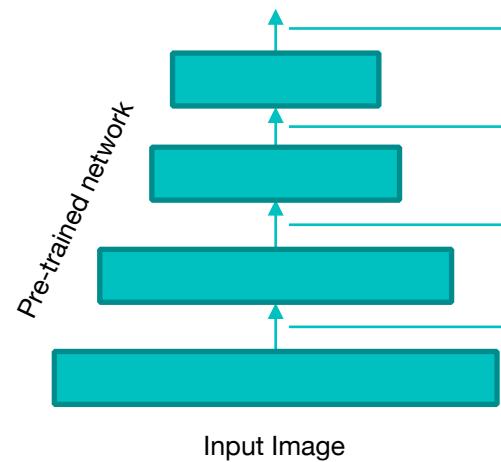


# More YOLO Variants

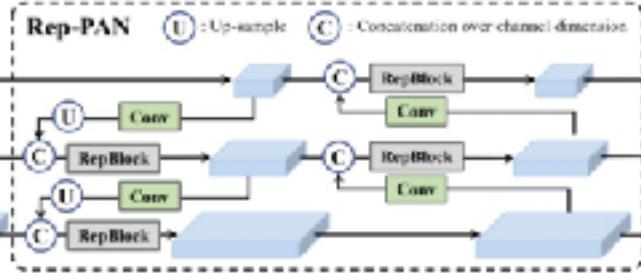


# YOLO Structures, In General

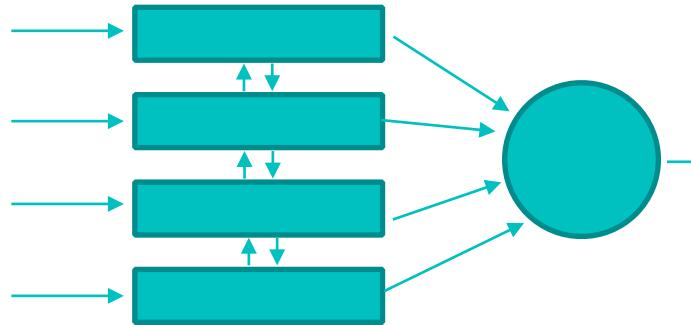
## Backbone



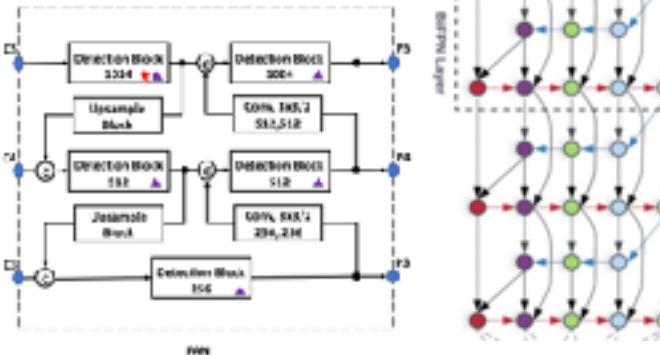
- YOLOv1-v4: Dark/ResNet
- Yv5: EfficientNet
- Yv6: EfficientNet-L2
- PP-Yv1-v2: ResNet50-vd
- Yv6 v3.0: RepVGG 😊



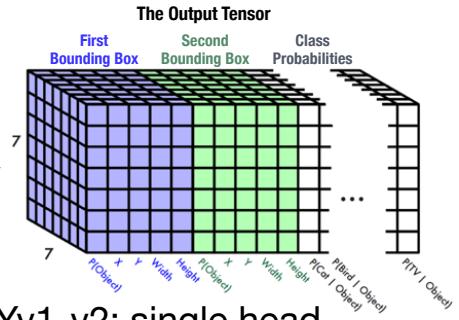
## Neck



- Yv1-v2: Transfer Learning, Single Layer
- Yv3: Upsampling, Multiple Layers
- Yv4: Cross Stage Partial layers (CSPNet). Use multiple layers from backbone with weighted concat
- PP-Y: Path Aggregation Network
- PP-Yv2: Multi-scale PAN



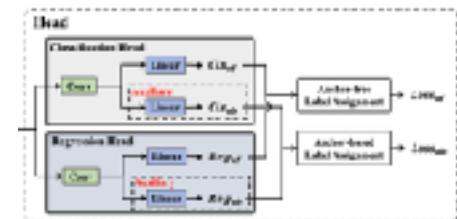
## Head(s)



- Yv1-v2: single head
- Yv3: Multi-resolution heads
- Yv4: Clustered anchor boxes and new gradient penalty
- PP-Yv2: IoU prediction loss
- Yv6-v8: Dense Anchor Boxes (anchors as aux. task)
- Yv5-8: Mosaic Augmentation

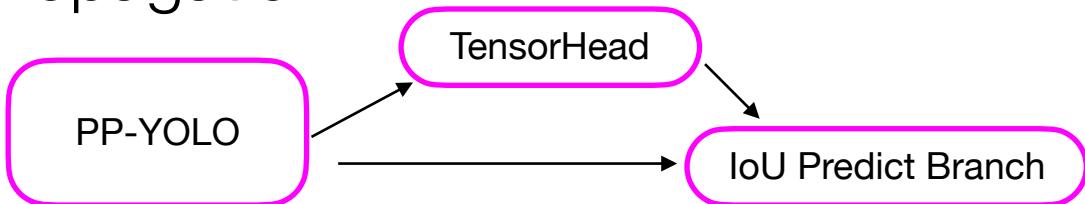
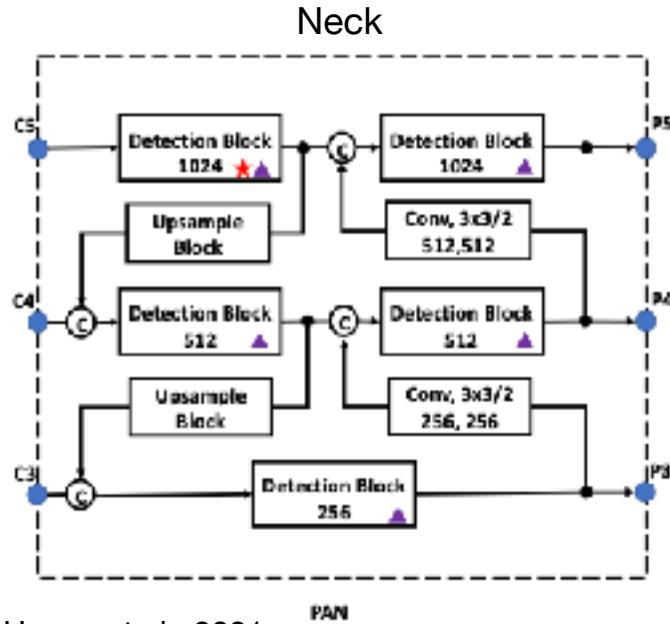
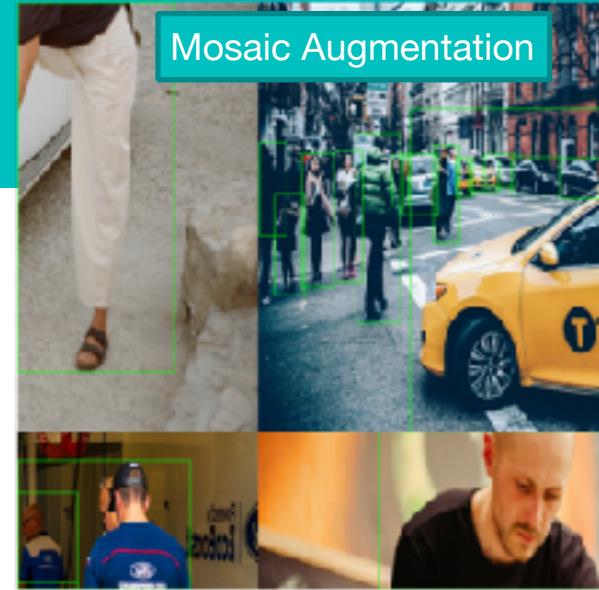
$$\mathcal{L}_{IoU} = -t_{IoU} \cdot \log(\sigma(p)) + (t_{IoU} - 1) \cdot \log(1 - \sigma(p))$$

IoU Prediction



# One Example: PP-YOLOv2

- Path Aggregation Network (PAN) for multi-scale processing
- Varied Input Sized Images and “Mish” activation
- “IoU Aware Branch” that uses the IoU prediction in the back propagation



$$\mathcal{L}_{IoU} = - t_{IoU} \cdot \log(\sigma(p)) + (t_{IoU} - 1) \cdot \log(1 - \sigma(p))$$

IoU between output of YOLO BBox and GT

$p$ =predicted IoU from branch

Main idea is to predict the IoU in a separate multi-task branch, which makes the network better at finding the correct bounding boxes



# PP-YOLOv2: Results

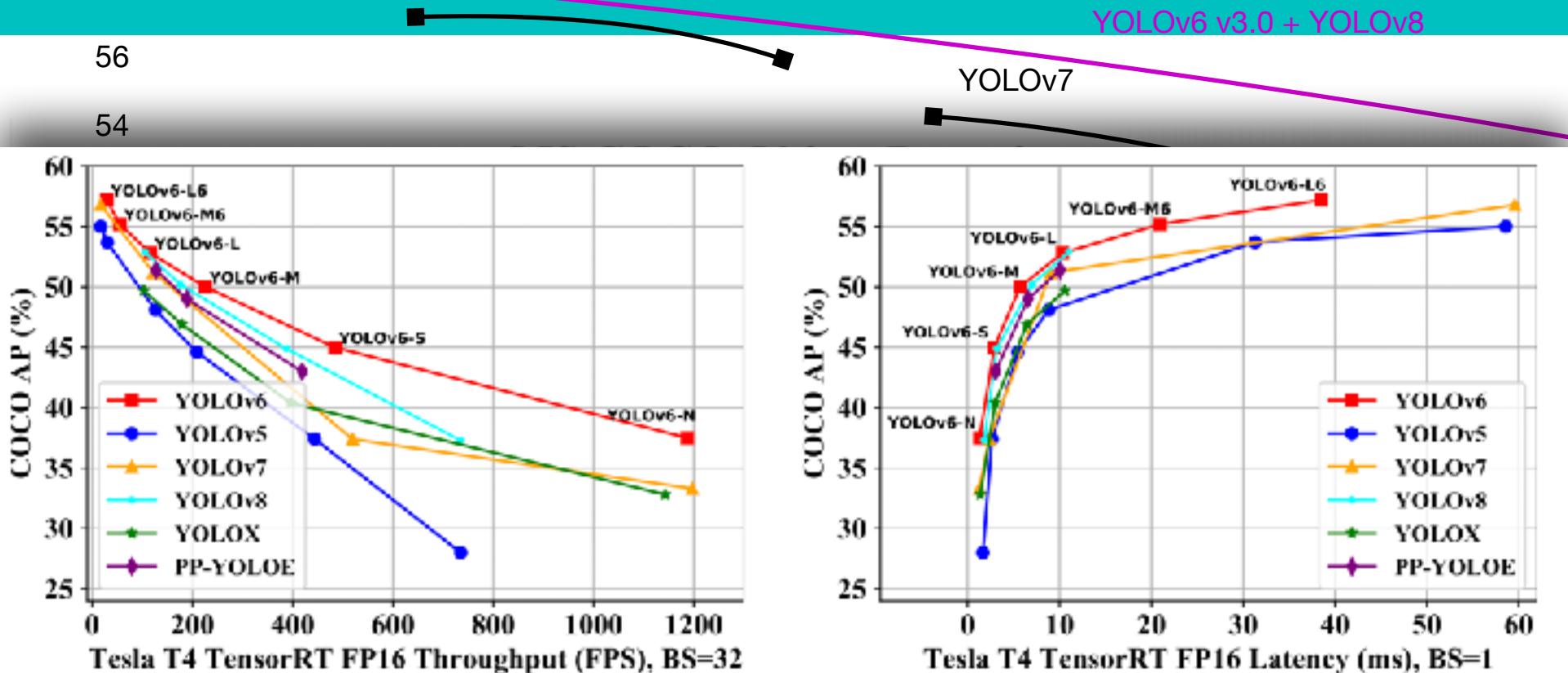


Figure 1: Comparison of state-of-the-art efficient object detectors. Both latency and throughput (at a batch size of 32) are given for a handy reference. All models are test with TensorRT 7.

