



UFOP

Universidade Federal
de Ouro Preto

Seminário da Disciplina:
Redes Neurais e Aprendizado em Profundidade



YOLOv3: An Incremental Improvement (Uma melhoria Incremental)

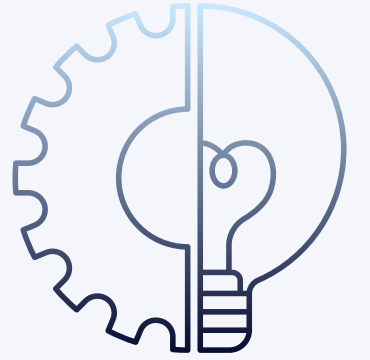
Joseph Redmon e Ali Farhadi (2018)
University of Washington

Halliday Gauss Costa dos Santos
Vinicius de Paula Silva

Ouro Preto (Outubro-2022)



Organização do Trabalho



01

Introdução

02

Desenvolvimento

03

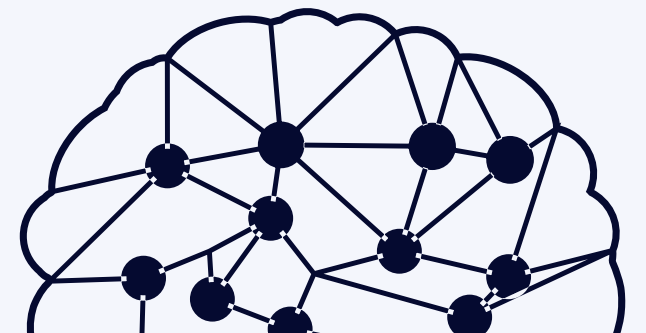
Ideias que não funcionaram

04

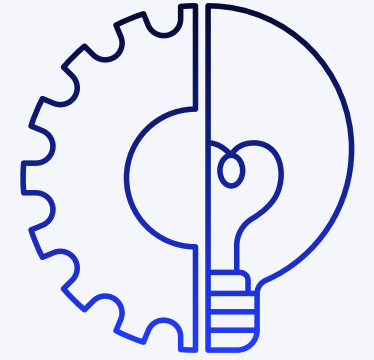
Conclusão

05

Referências



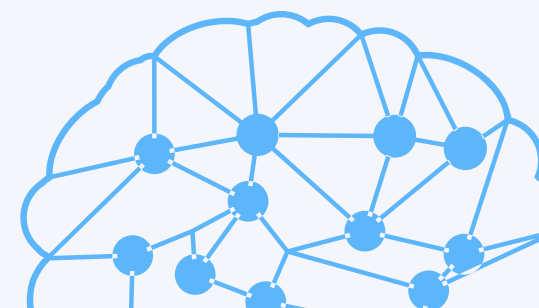
Introdução - Conceitos - YOLO



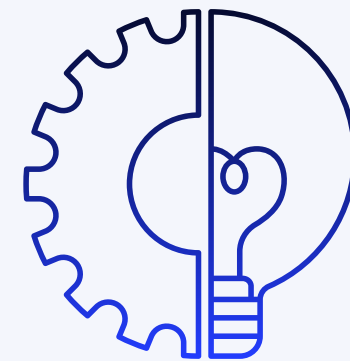
Desenvolvido por Joseph Redmon e Ali Farhadi em 2015 (Autores do Artigo).



Veio como uma abordagem totalmente nova sendo capaz de obter uma precisão igual ou superior ao dos outros métodos de detecção de objetos da época.

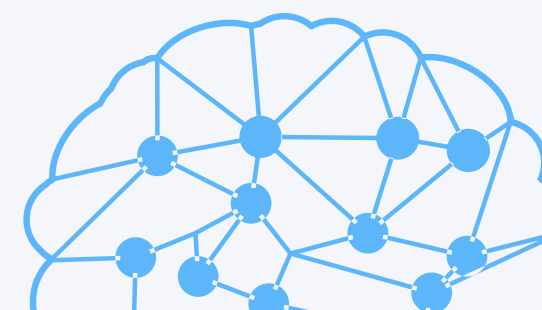


Introdução - Conceitos - YOLO

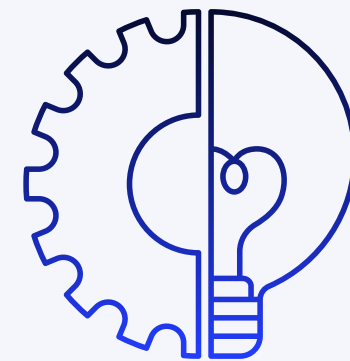


■ **YOLO (You Only Look Once – “Você só olha uma vez”)** é um método de detecção de objetos de passada única (single pass) que utiliza uma rede neural convolucional como extrator de características (features). Diferente da R-CNN ou Faster R-CNN.

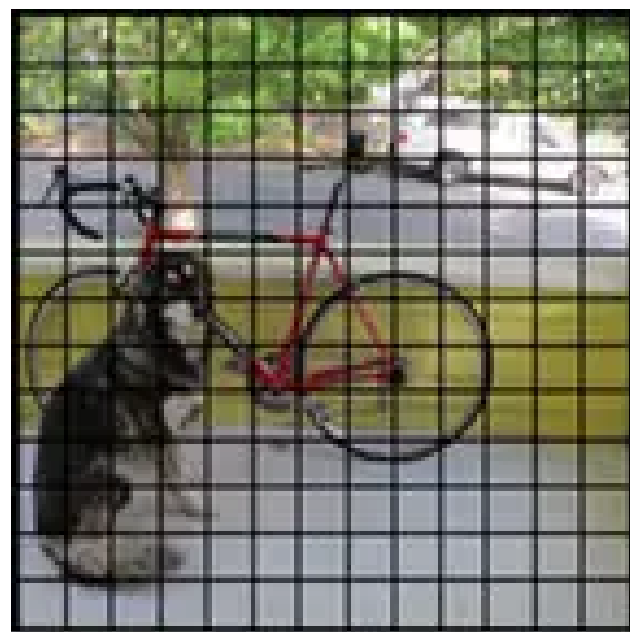
■ **Maior velocidade na detecção sem perdas grandes de acurácia.**



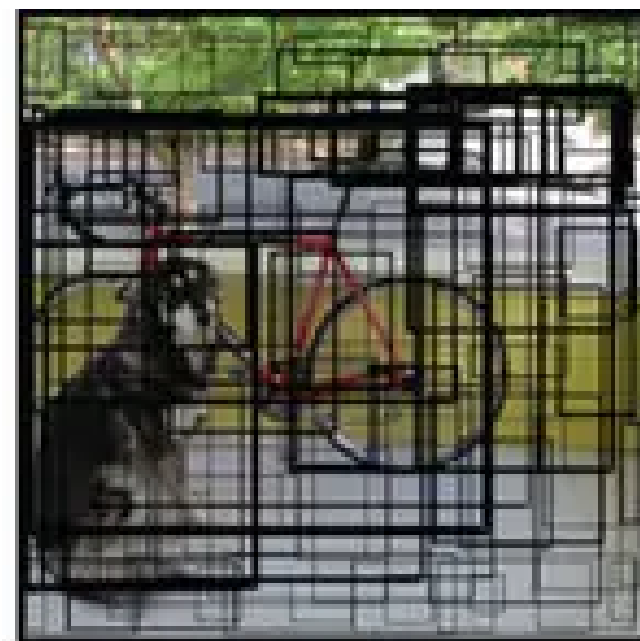
Introdução - Conceitos - YOLO



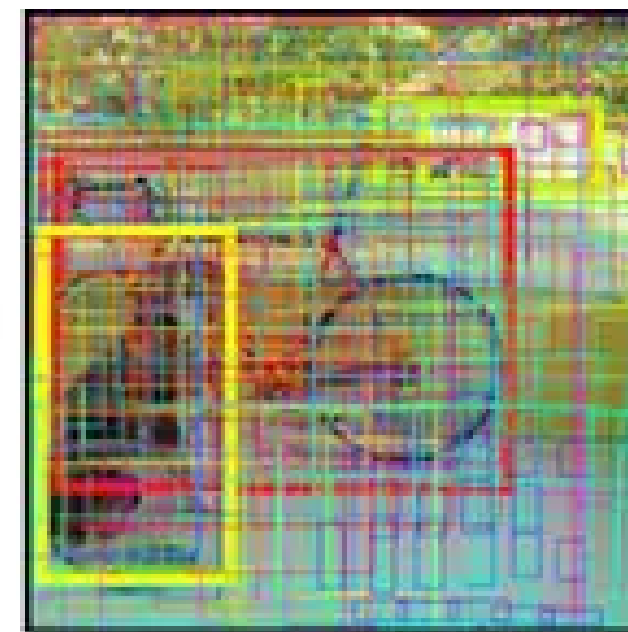
Funcionamento



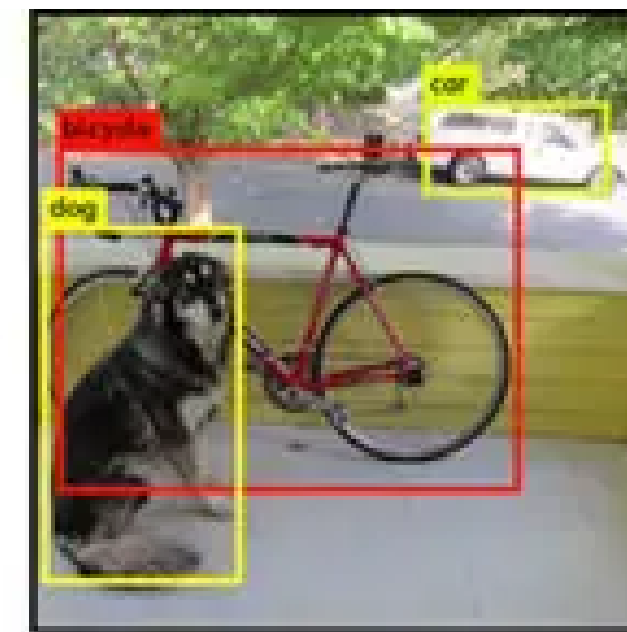
Dividir a imagem em um grid de $S \times S$ células.



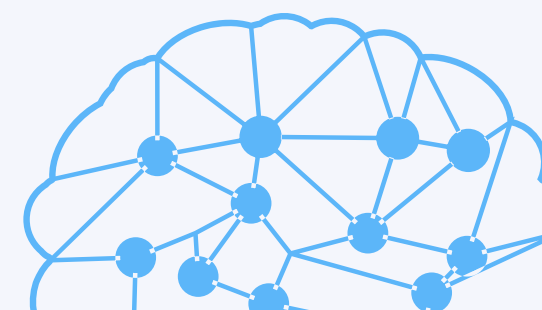
Cada célula prediz várias caixas delimitadoras retornando o valor de confiança que tenha um objeto.



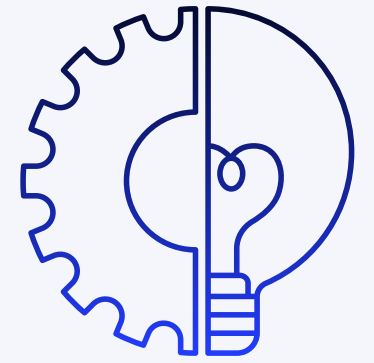
Cada caixa faz a previsão de uma classe. É retornado o valor da probabilidade (combinação entre confiança da caixa e predição de um objeto específico).



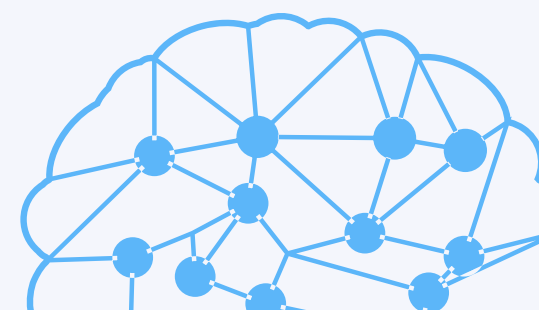
Considera apenas as caixas cuja pontuação final seja maior que um limiar.



Introdução - Conceitos - Yolo

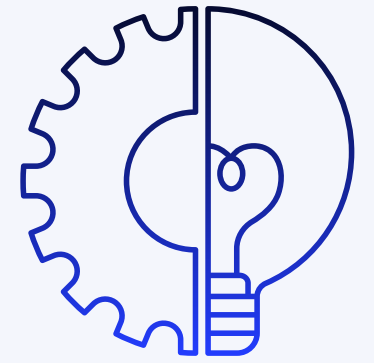


 **YOLO utiliza uma rede neural profunda, cuja arquitetura é chamada de Darknet.**





Introdução - Conceitos - Outros Modelos



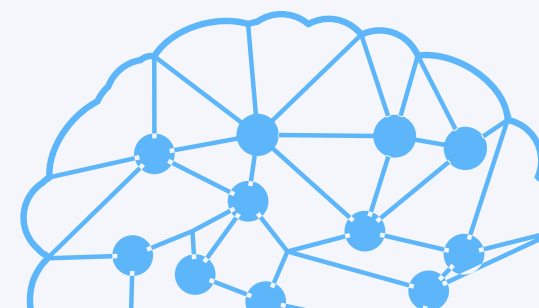
Modelos citados:



RetinaNet (ResNet e ResNext).



SSD (ResNet).



Introdução - Conceitos - Métricas

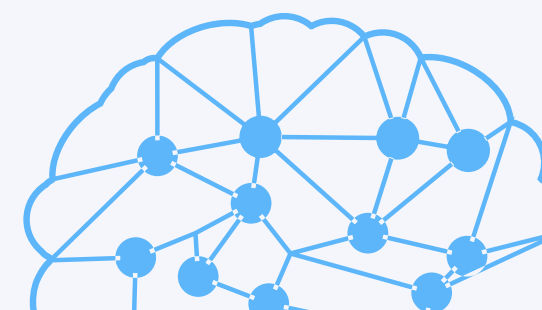
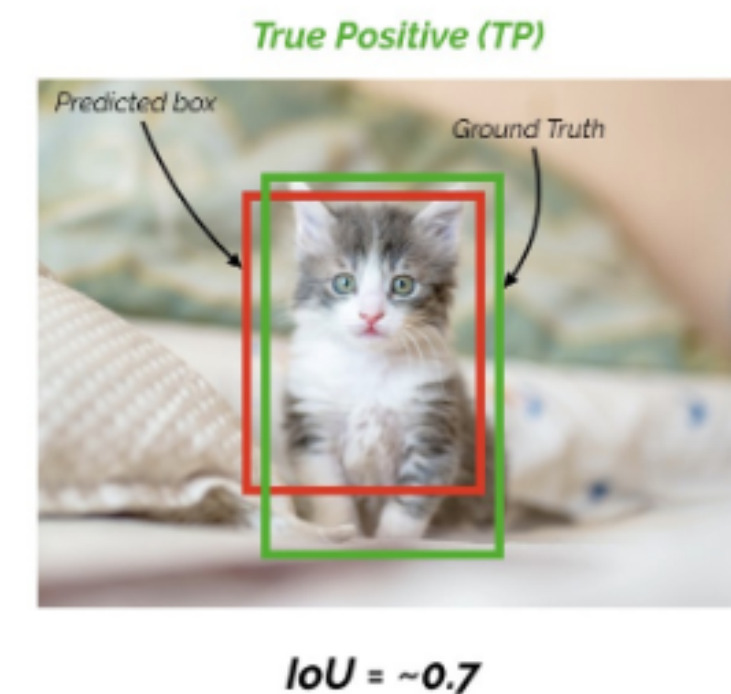
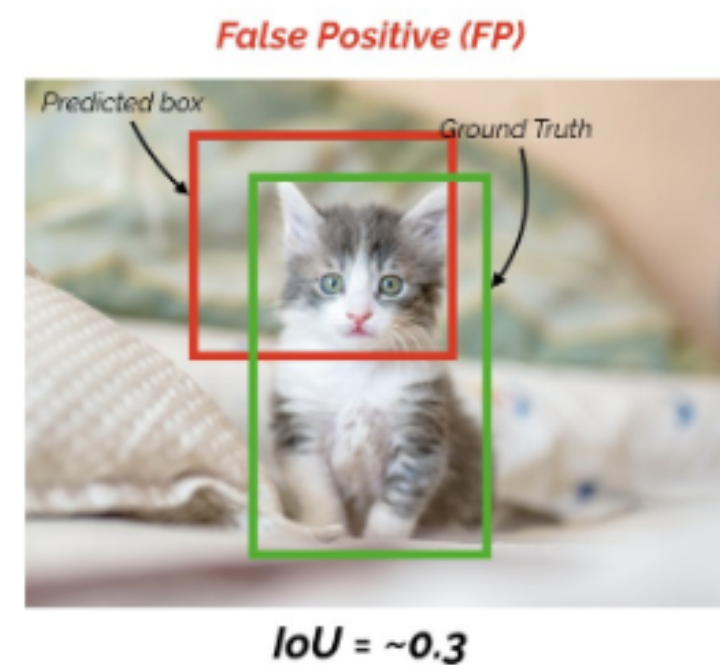


■ Precisão Média Média (mAP)

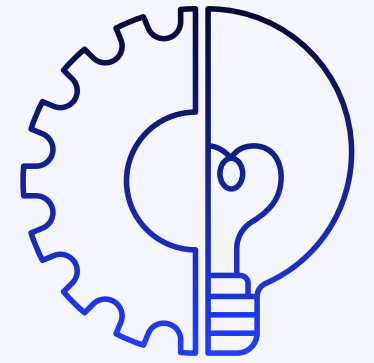
■ Importante para avaliar o desempenho dos modelos de detecção de objetos

■ Baseadas em: Matriz de confusão, Intersecção sobre União (IoU), Recall e Precisão.

If IoU threshold = 0.5



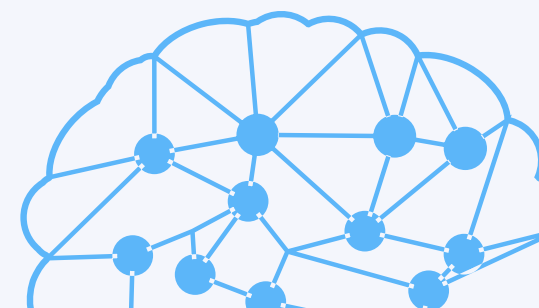
Introdução - Conceitos - Métricas

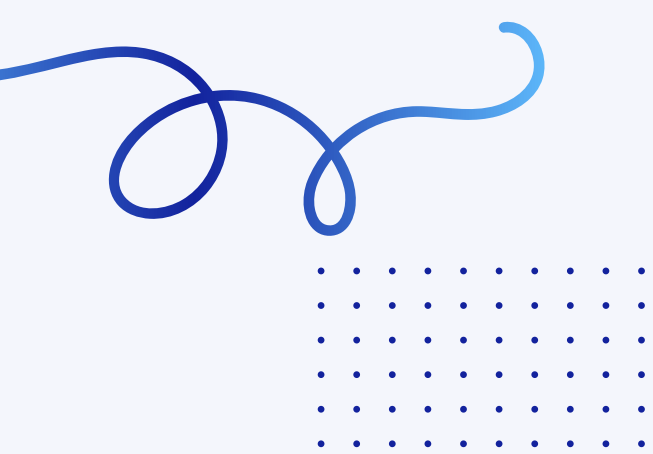


Calculando o mAP:

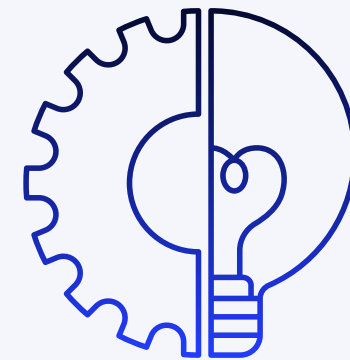
O mAP é calculado encontrando a Precisão média (AP) para cada classe e, em seguida, a média de várias classes.

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^N \text{AP}_i$$





Introdução - Conceitos - Métricas



Calculando o AP:



Gere as pontuações de previsão usando o modelo.



Converta as pontuações de previsão em rótulos de classe.



Calcule a matriz de confusão – TP, FP, TN, FN.



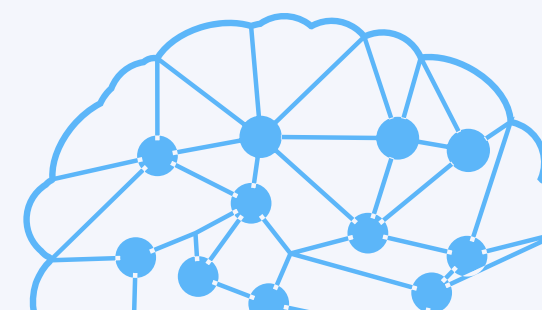
Calcule as métricas de precisão e recall.



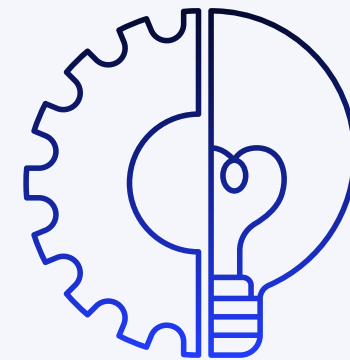
Calcule a área sob a curva de precisão-recall.



Meça a precisão média.



Introdução - Conceitos - Notações



■ **AP: AP com IoU de 50%, 95% e 5%.**

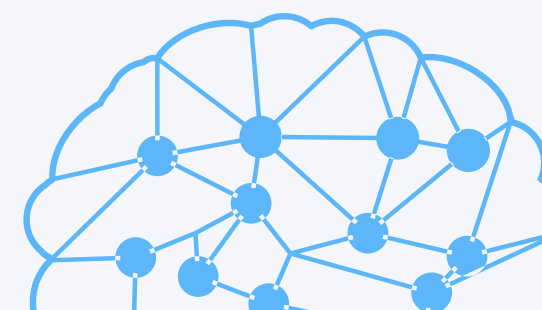
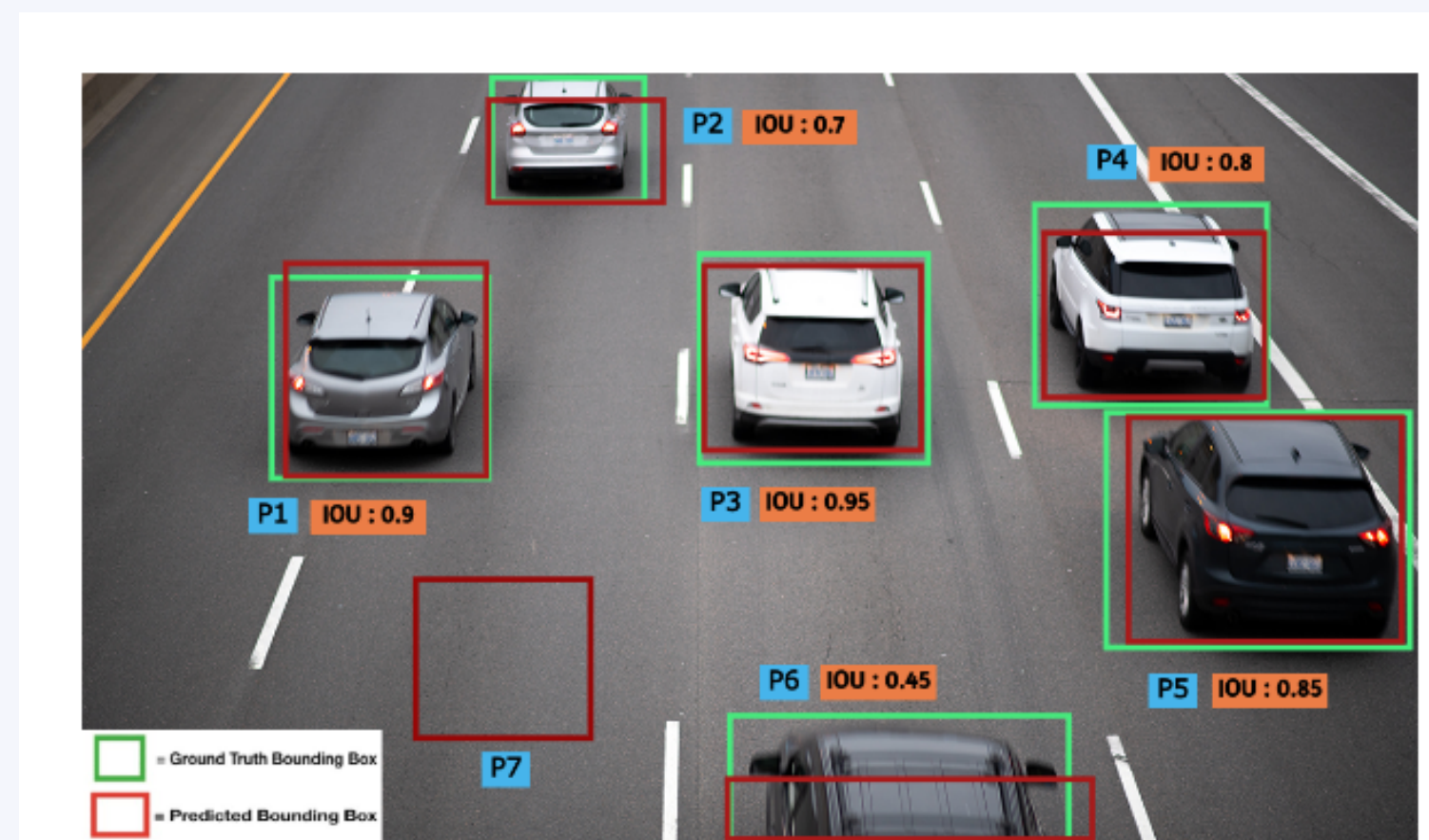
■ **AP50: AP com IoU de 50%.**

■ **AP75: AP com IoU de 75%.**

■ **APS: AP para objetos pequenos.**

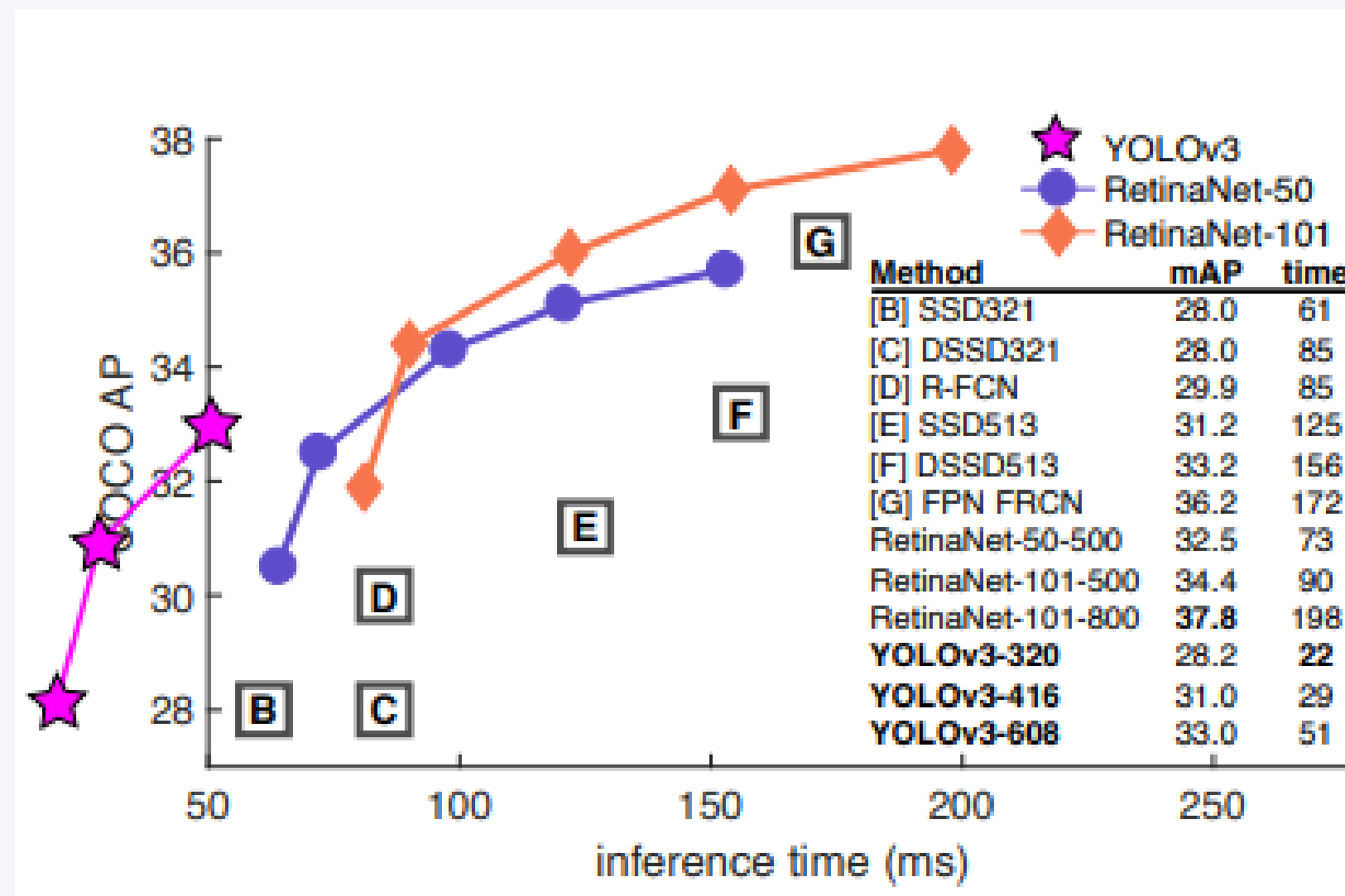
■ **APM: AP para objetos de tamanho médio.**

■ **APL: AP para objetos grandes.**

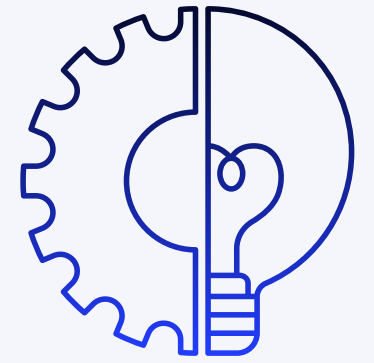


Introdução - Objetivo

Obter melhorias no YOLO.



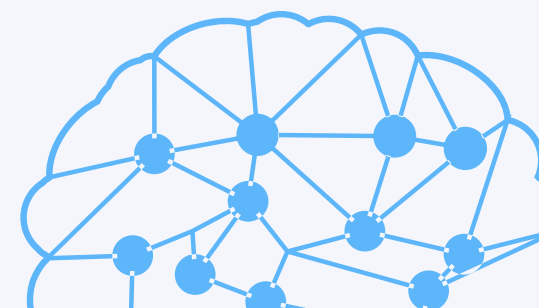
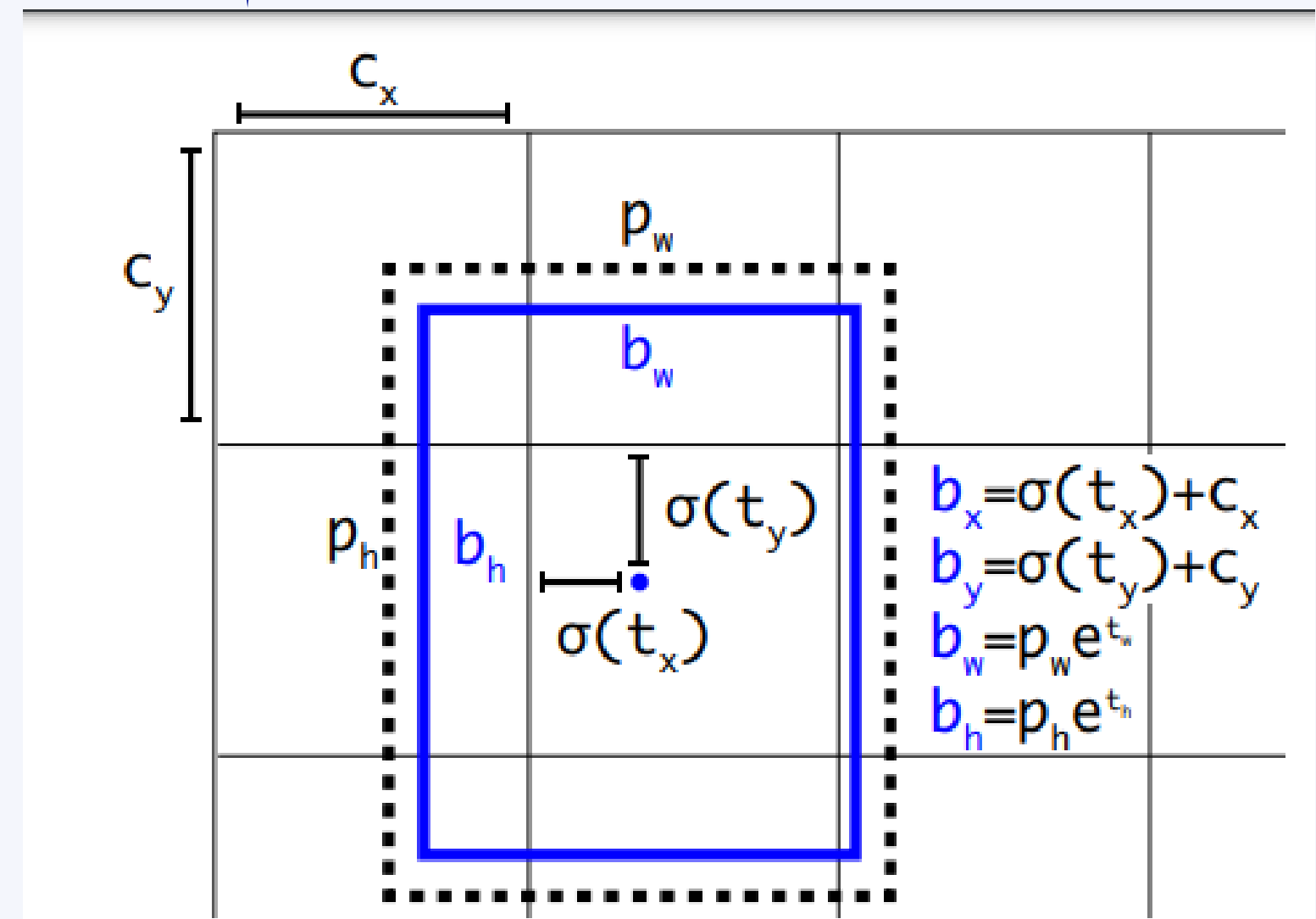
Desenvolvimento - Predição da caixa delimitadora



Assim como o YOLO9000, o sistema prevê caixas delimitadoras usando *anchor boxes*.

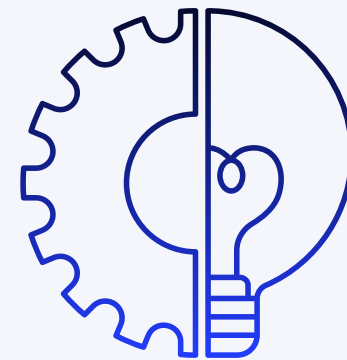
A rede prevê 4 coordenadas para cada caixa delimitadora, t_x , t_y , t_w , t_h . Também é utilizado o deslocamento da célula c_x e c_y .

É previsto a largura e a altura da caixa como deslocamentos dos centróides do cluster. É predito as coordenadas do centro da caixa em relação ao local de aplicação do filtro usando uma função sigmóide.

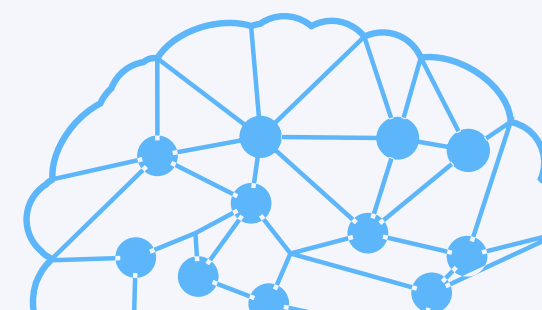




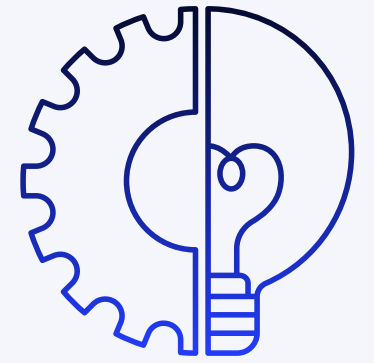
Desenvolvimento - Predição de classe



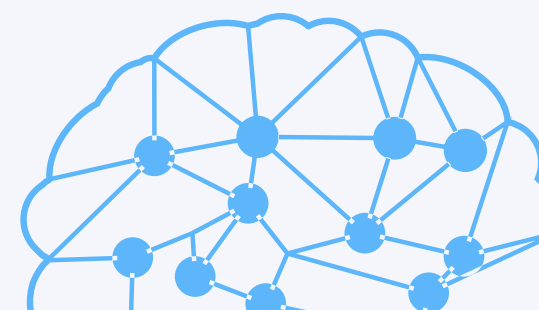
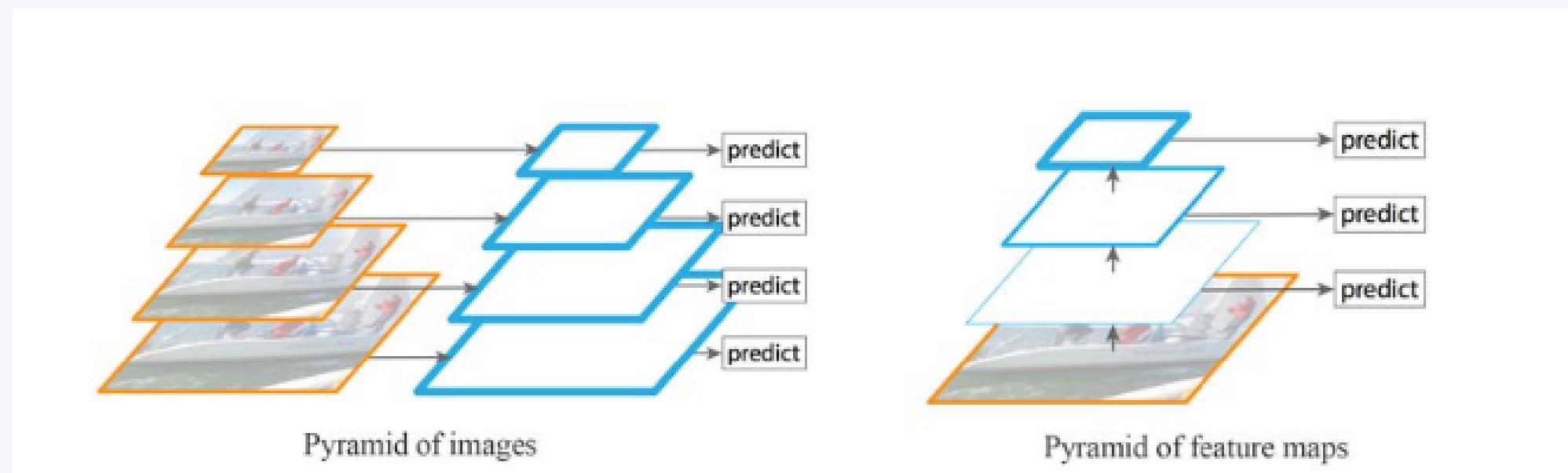
- **Dentro de cada caixa delimitadora será predito as classes que podem aparecer usando classificação multirrotulo.**
- **Ao invés de usar softmax foi utilizado classificadores logísticos independentes, para ganhar desempenho.**
- **A loss utilizado no treinamento foi: binary cross-entropy**



Desenvolvimento - Predições utilizando escalas

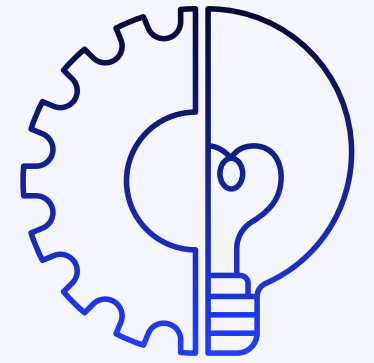


- **YOLOv3 prevê caixas em 3 escalas diferentes.**
- **O sistema extrai características dessas escalas usando um conceito similar para caracterizar redes de pirâmide.**

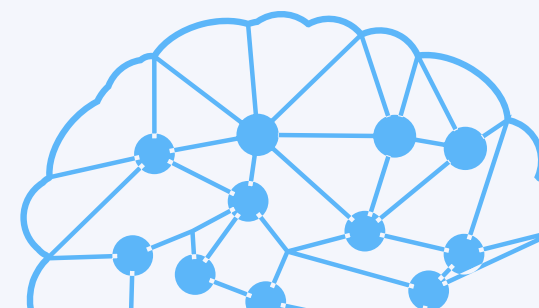




Desenvolvimento - Extrator de Características



- Criada uma nova rede baseada na mistura da Darknet-19 da YOLOv2, com diversas camadas convolucionais de 3X3 e 1X1 em sequência.
- Rede mais larga com 53 camadas convolução, chamada de Darknet-53.

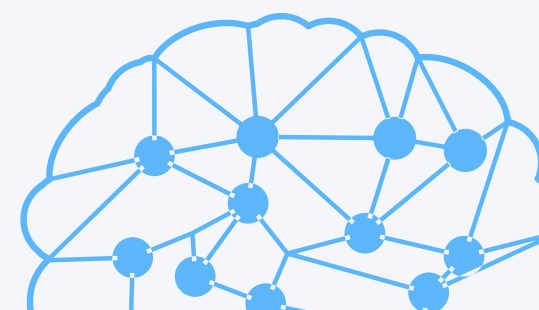


Desenvolvimento - Extrator de Características



	Type	Filters	Size	Output
1x	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
2x	Convolutional	128	3 × 3 / 2	64 × 64
	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
8x	Convolutional	256	3 × 3 / 2	32 × 32
	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8x	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4x	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Backbone	Top-1	Top-5	Bn Ops	BFLOP/s	FPS
Darknet-19 [15]	74.1	91.8	7.29	1246	171
ResNet-101[5]	77.1	93.7	19.7	1039	53
ResNet-152 [5]	77.6	93.8	29.4	1090	37
Darknet-53	77.2	93.8	18.7	1457	78

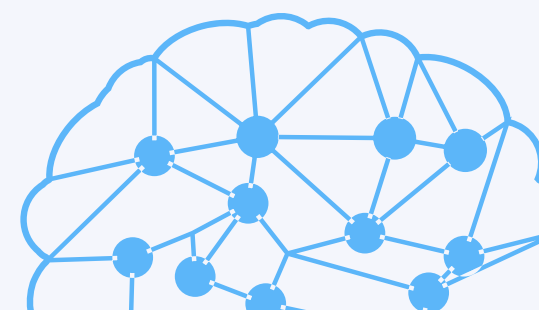
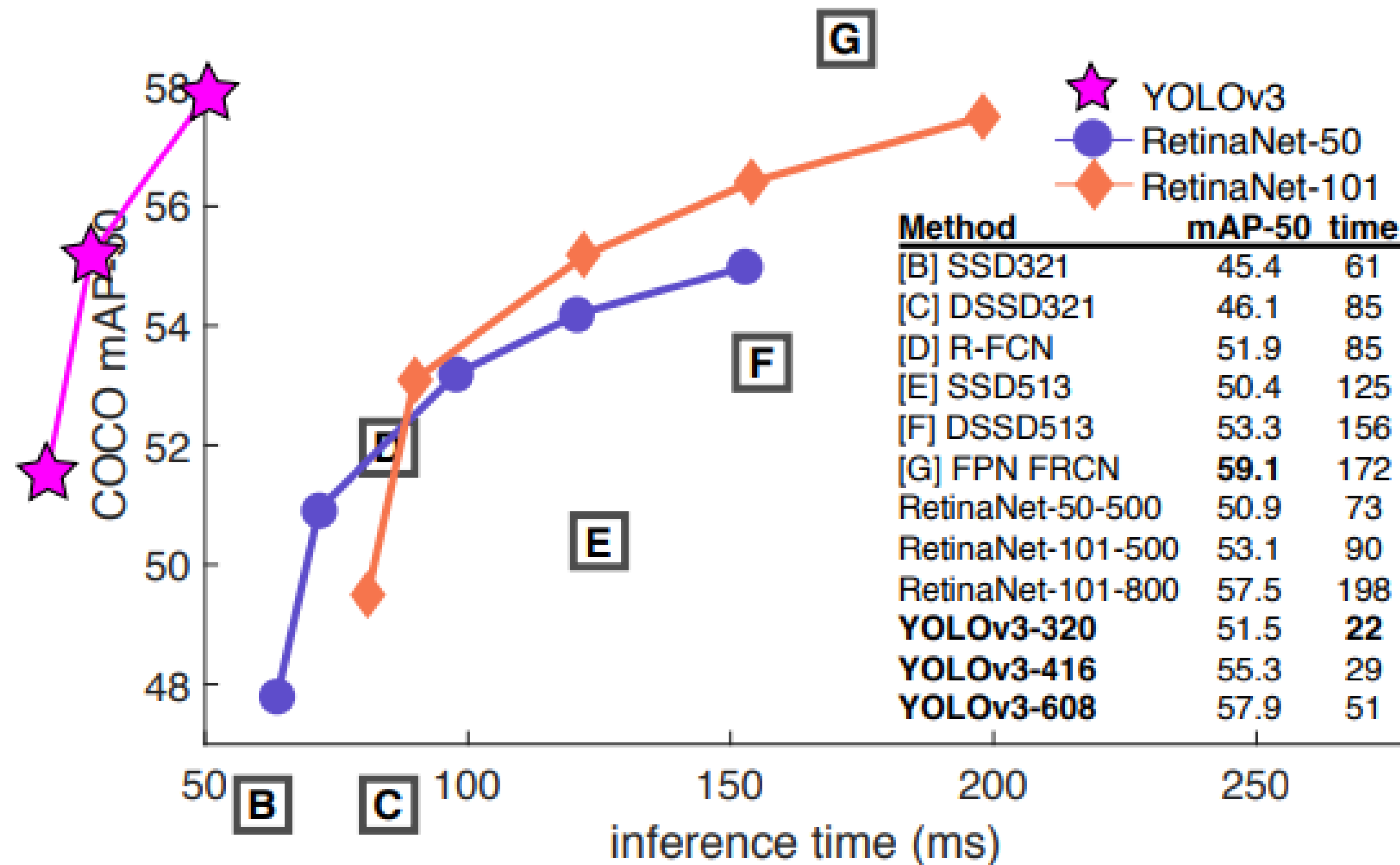
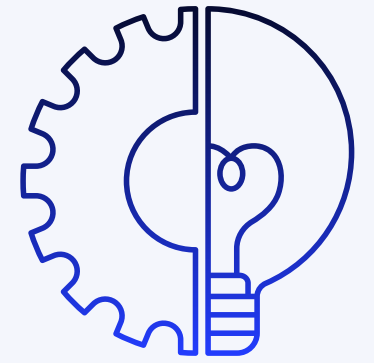


Desenvolvimento - Resultados

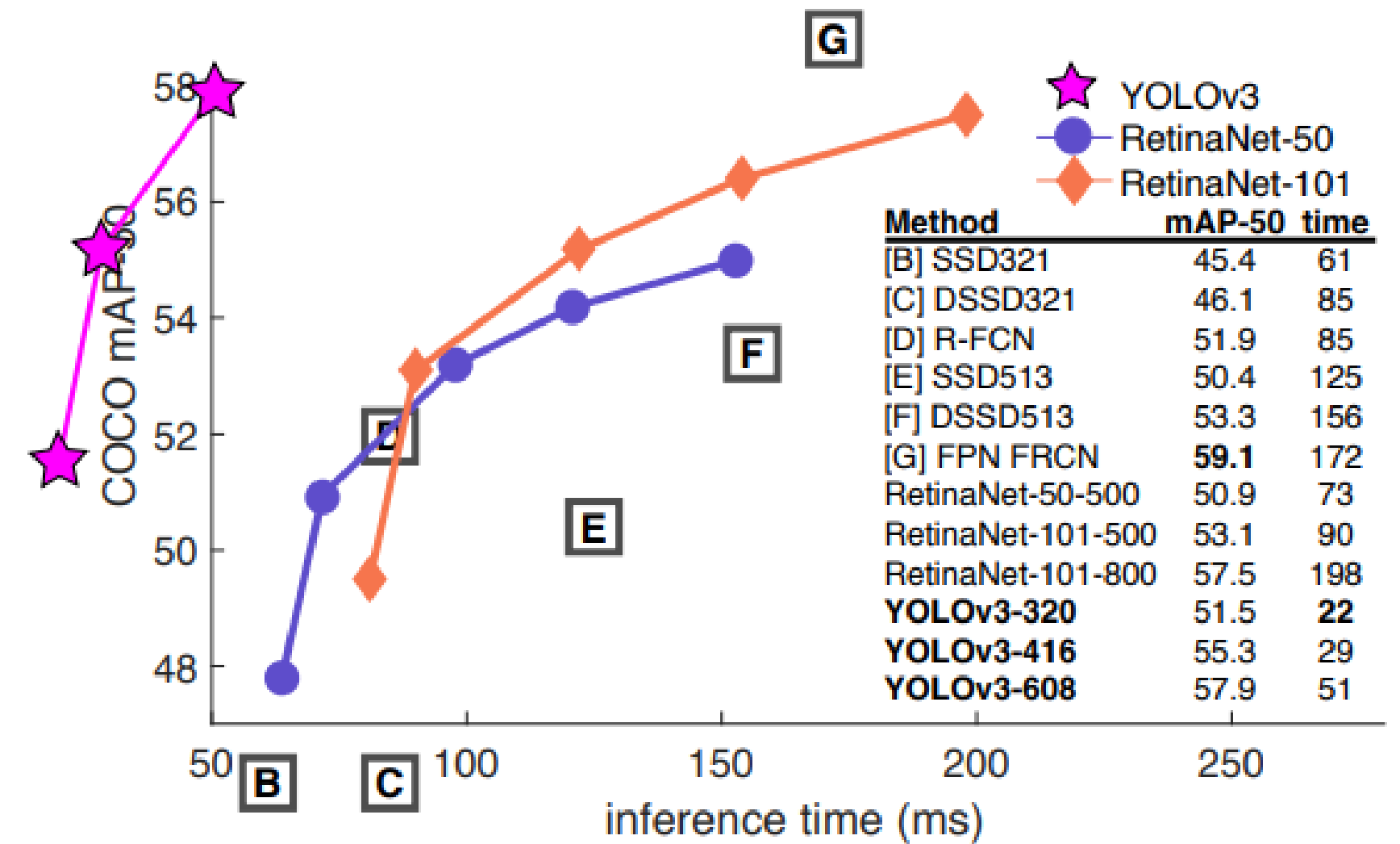
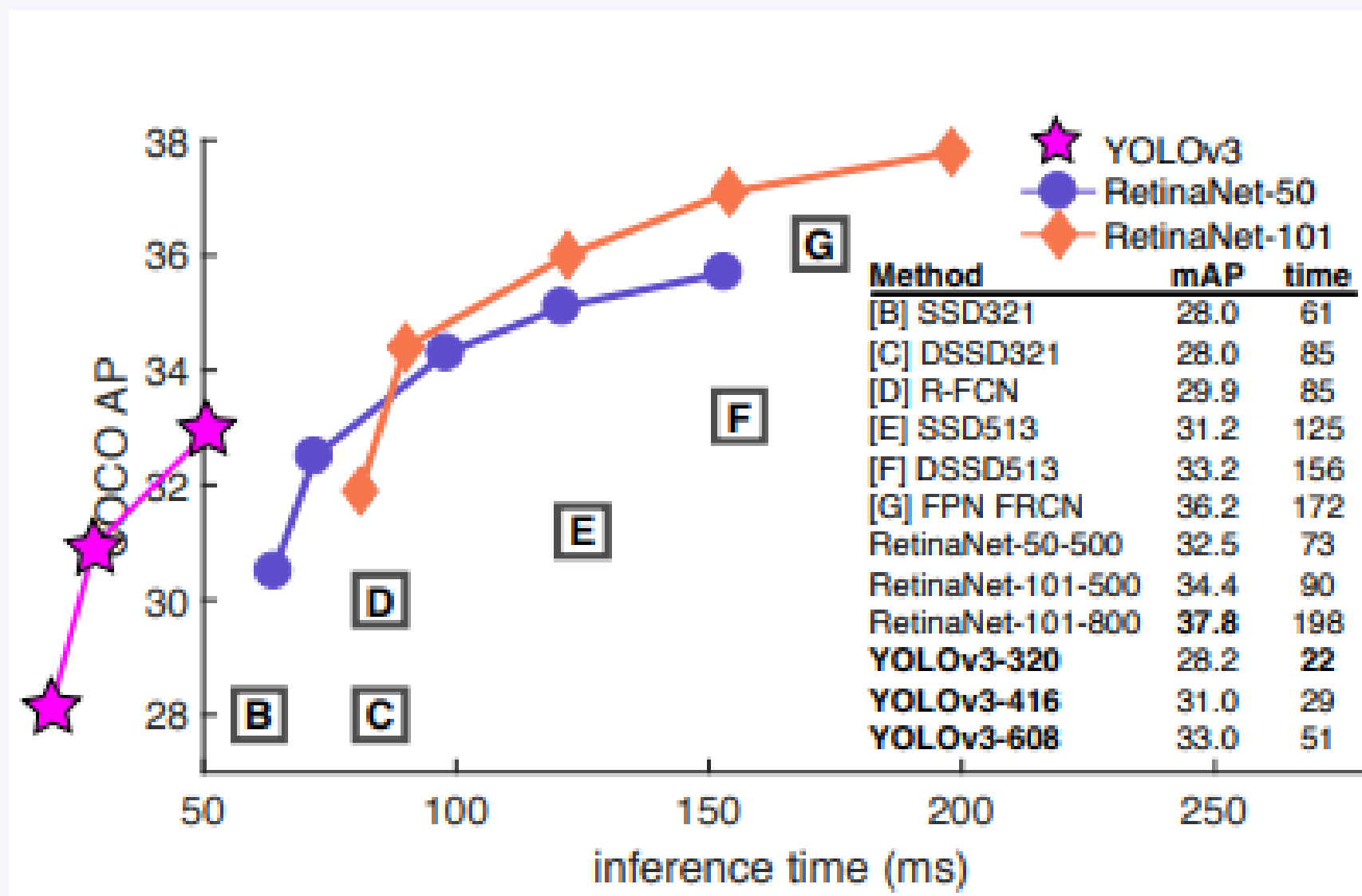


	backbone	AP	AP ₅₀	AP ₇₅	AP _S	AP _M	AP _L
<i>Two-stage methods</i>							
Faster R-CNN+++ [5]	ResNet-101-C4	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN w FPN [8]	ResNet-101-FPN	36.2	59.1	39.0	18.2	39.0	48.2
Faster R-CNN by G-RMI [6]	Inception-ResNet-v2 [21]	34.7	55.5	36.7	13.5	38.1	52.0
Faster R-CNN w TDM [20]	Inception-ResNet-v2-TDM	36.8	57.7	39.2	16.2	39.8	52.1
<i>One-stage methods</i>							
YOLOv2 [15]	DarkNet-19 [15]	21.6	44.0	19.2	5.0	22.4	35.5
SSD513 [11, 3]	ResNet-101-SSD	31.2	50.4	33.3	10.2	34.5	49.8
DSSD513 [3]	ResNet-101-DSSD	33.2	53.3	35.2	13.0	35.4	51.1
RetinaNet [9]	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
RetinaNet [9]	ResNeXt-101-FPN	40.8	61.1	44.1	24.1	44.2	51.2
YOLOv3 608 × 608	Darknet-53	33.0	57.9	34.4	18.3	35.4	41.9

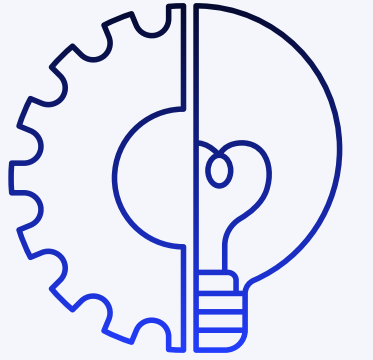
Desenvolvimento - Resultados



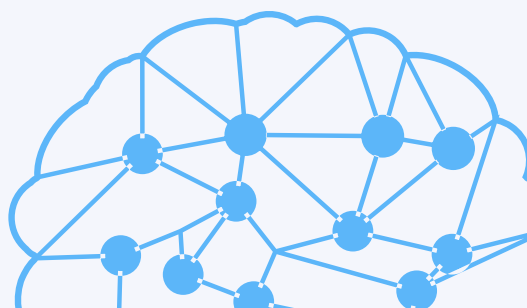
Desenvolvimento - Resultados - Comparação



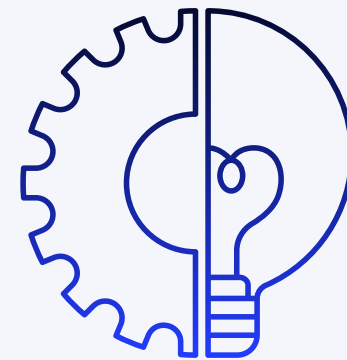
Ideias que não funcionaram



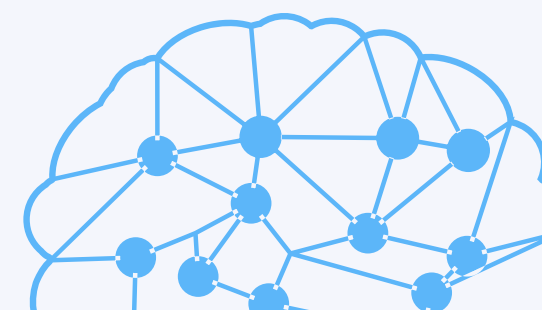
- ***Anchor box* x, y , para previsões de deslocamento:**
- **Prever deslocamento x, y como múltiplo da altura e largura da caixa usando uma ativação linear.**
- **Diminuiu a estabilidade do modelo.**



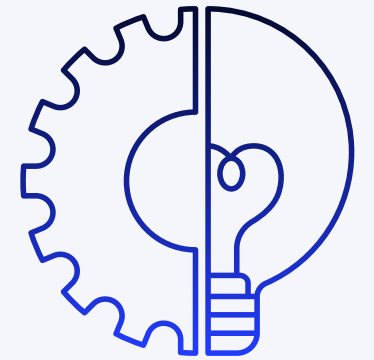
Ideias que não funcionaram




- **Predição linear de x , y , ao invés de predições logísticas:**
- **Ativação linear para prever diretamente o deslocamento x , y .**
- **Diminuiu o mAP.**



Ideias que não funcionaram



Focal Loss:

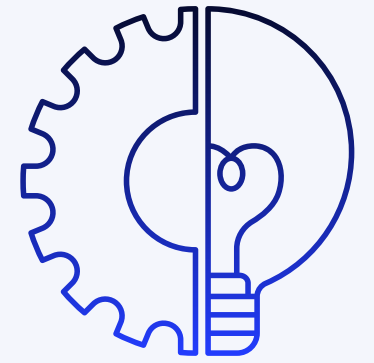
 **Aprimoramento da Cross-Entropy Loss e é introduzida para lidar com o problema de desequilíbrio de classe com modelos de detecção de objeto de estágio único.**

 **Usada na RetinaNet.**

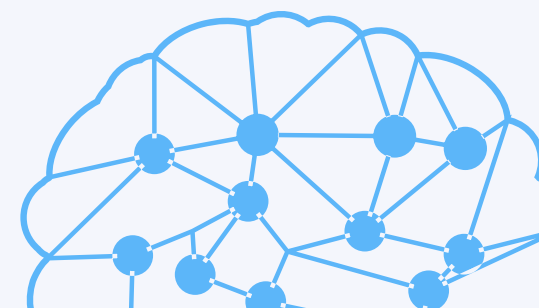
 **Diminiu em 2 pontos o mAP.**



Conclusão

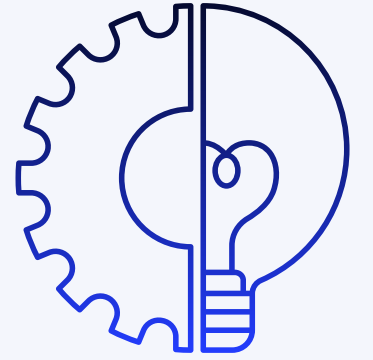


- **Versão do YOLO: YOLOv3 ainda rápida e mais precisa.**
- **Em 320X320 é executado em 22 ms a 28,2 mAP, tão preciso quanto o SSD, mas três vezes mais rápido**
- **Considerando a métrica mAP de .5 IOU, YOLOv3 é muito boa, atingindo 57,9 AP50 em 51 ms em uma Titan X, comparado a 57,5 AP50 em 198 ms por uma RetinaNet, desempenho semelhante, mas 3,8 vezes mais rápido.**
- **Não tão boa entre 0,5 e 0,95 de IOU.**





Referências



Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." arXiv preprint arXiv:1804.02767 (2018).

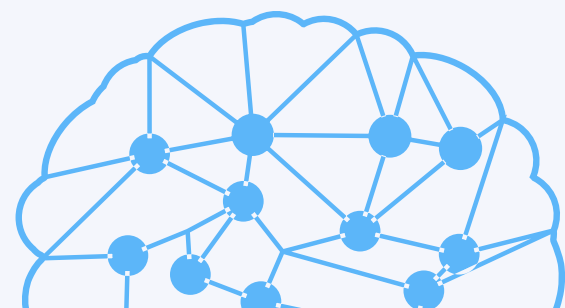
ALVES, G. Detecção de Objetos com YOLO – Uma abordagem moderna. Disponível em: <<https://iaexpert.academy/2020/10/13/deteccao-de-objetos-com-yolo-uma-abordagem-moderna/>>. Acesso em: 18 out. 2022.

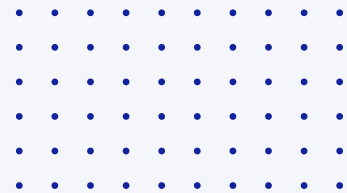
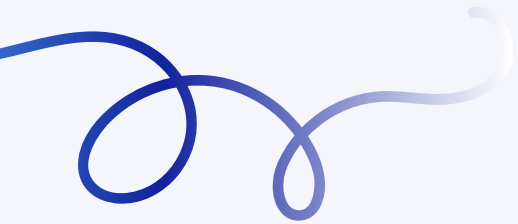
SHAH, D. Mean Average Precision (mAP) explained: Everything you need to know. V7labs.comV7, , 7 out. 2022. Disponível em: <<https://www.v7labs.com/blog/mean-average-precision>>. Acesso em: 18 out. 2022

REDMON, J. YOLO: Real-time object detection. Disponível em: <<https://pjreddie.com/darknet/yolo/>>. Acesso em: 18 out. 2022.

HUI, J. Understanding Feature Pyramid Networks for object detection (FPN). Disponível em: <<https://jonathan-hui.medium.com/understanding-feature-pyramid-networks-for-object-detection-fpn-45b227b9106c>>. Acesso em: 18 oct. 2022.

How RetinaNet works? Disponível em: <<https://developers.arcgis.com/python/guide/how-retinanet-works/>>. Acesso em: 18 oct. 2022.





Obrigado!

