

Detecção de Objetos

Método de passagem única - YOLO

Gabriel Simões

Agenda

- Problemas relacionados
- Visão do problema
- Uma alternativa de solução
- Métricas de avaliação
- YOLO
 - Detalhamento do YOLO

Problemas Relacionados

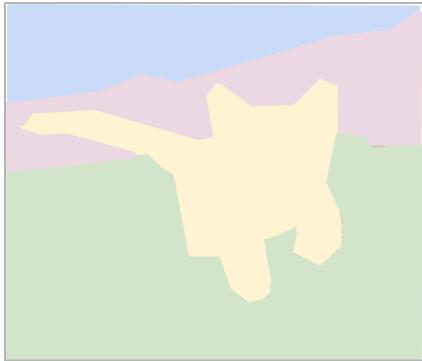


Segmentação Semântica



Classificação + Localização

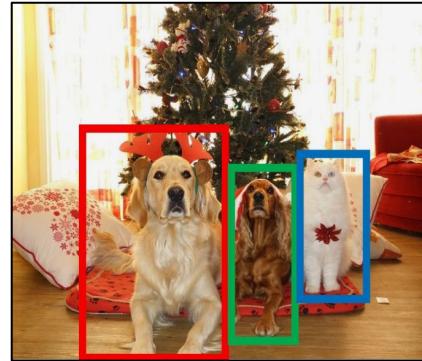
Problemas Relacionados



Segmentação Semântica



Classificação + Localização

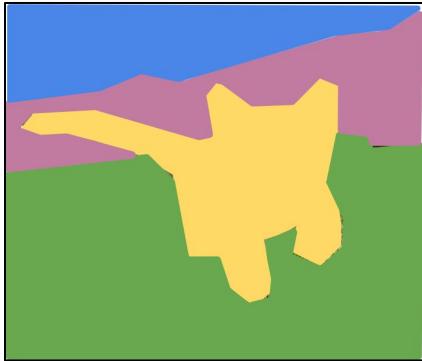


Detecção de Objetos

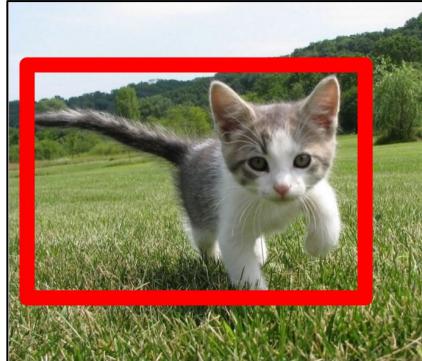


Segmentação de Instâncias

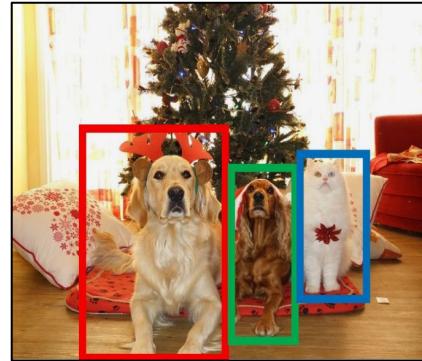
Problemas Relacionados



Segmentação Semântica



Classificação + Localização



Detecção de Objetos



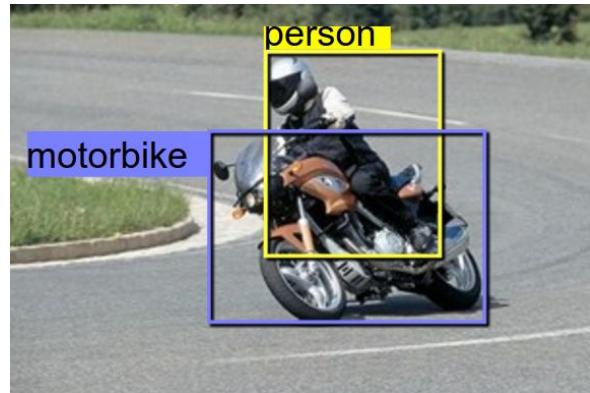
Segmentação de Instâncias

Visão do Problema

Dadas as Classes: {car, dog, motorbike, person}



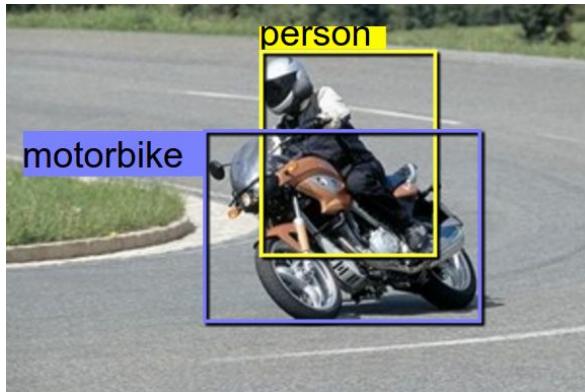
Entrada



Saída

Visão do Problema

Dadas as Classes: {car, dog, motorbike, person}



3 problemas:

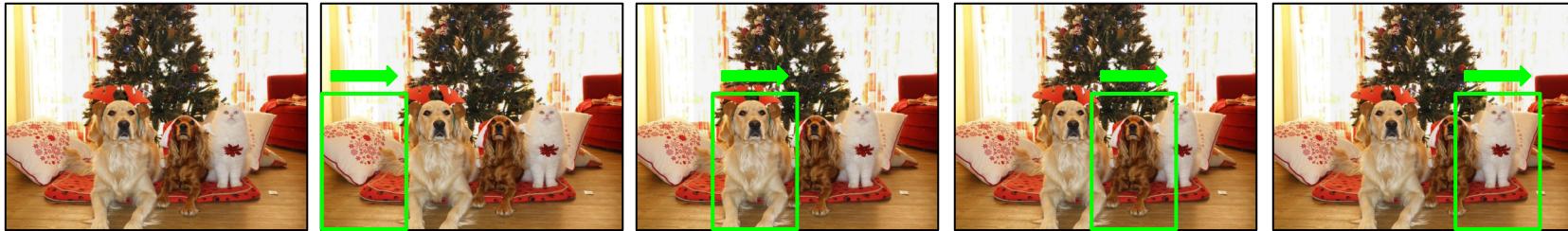
- Posição (x,y)
- Dimensão (w,h)
- Classificação {car, dog, motorbike, person}

Janela Deslizante



Dada uma imagem de entrada...

Janela Deslizante



Cachorro: 0
Gato: 0
Fundo: 1

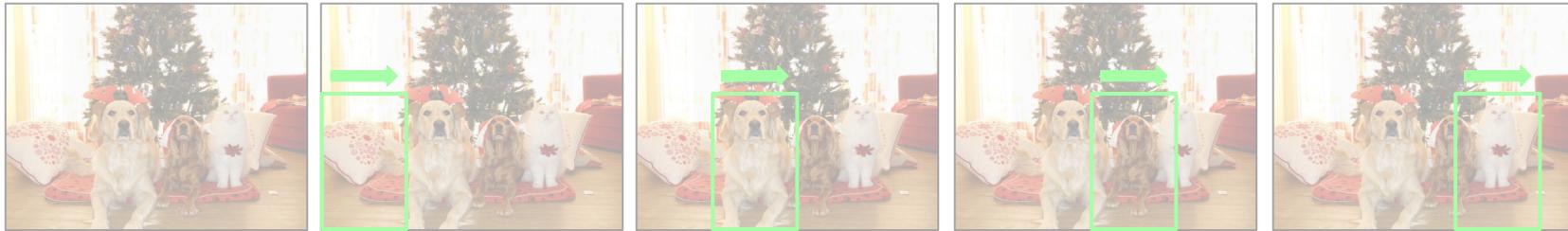
Cachorro: 1
Gato: 0
Fundo: 0

Cachorro: 1
Gato: 0
Fundo: 0

Cachorro: 0
Gato: 1
Fundo: 0

Deslizar uma janela classificando o conteúdo de um dado recorte.

Janela Deslizante



Cachorro: 0
Gato: 0
Fundo: 1

Cachorro: 1
Gato: 0
Fundo: 0

Cachorro: 1
Gato: 0
Fundo: 0

Cachorro: 0
Gato: 1
Fundo: 0

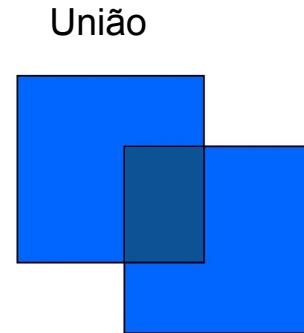
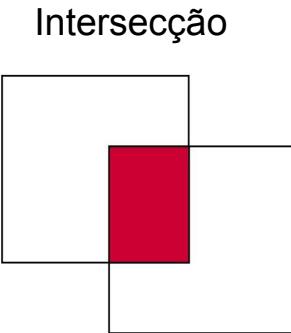
Mas qual o tamanho ideal para esta janela? E quanto ela deve deslizar? E o custo?

Métricas de Avaliação

Que medidas são utilizadas para avaliar o desempenho dos métodos e seus modelos?

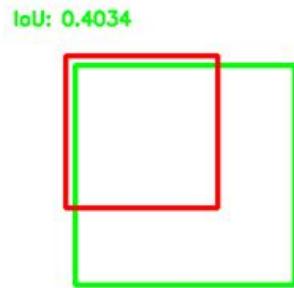
IoU

(Intersection over Union)

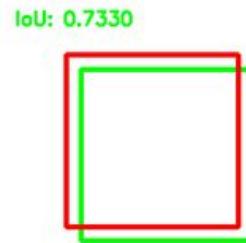


$$\text{IoU}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

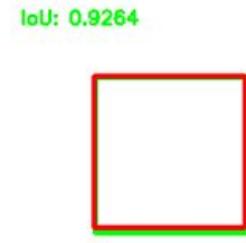
IoU (*Intersection over Union*)



Ruim



Bom



Excelente

IoU (Intersection over Union)



IoU

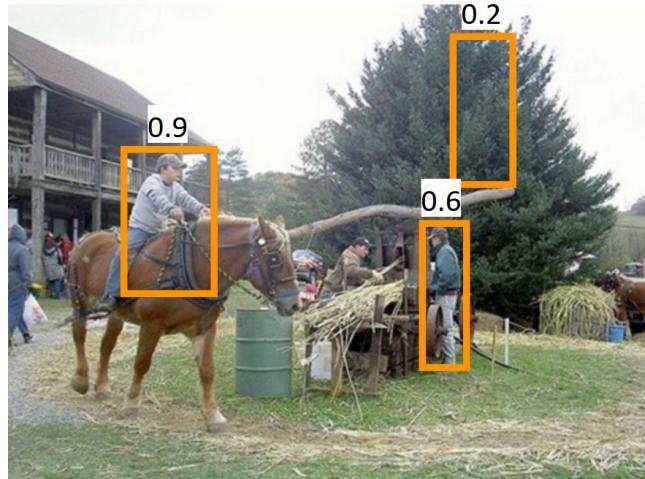
(Intersection over Union)



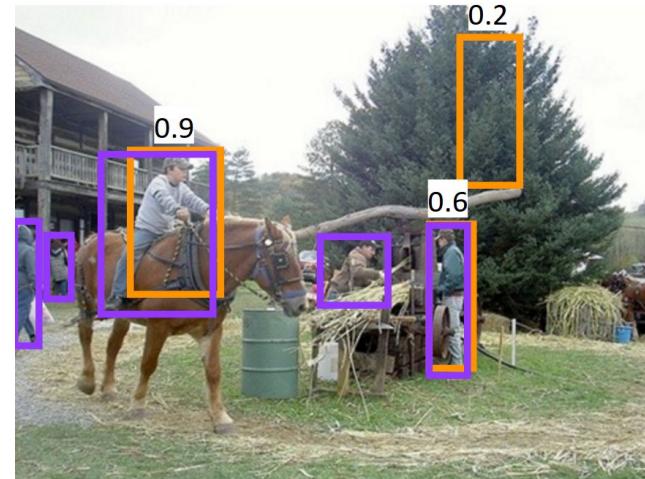
A maioria dos métodos de localização e detecção de objetos assume como corretas as predições com $\text{IoU} > 0.5$.

MAP (mean average precision)

Dada a classe person...



Predições



Ground Truth/Predições

MAP

(mean average precision)



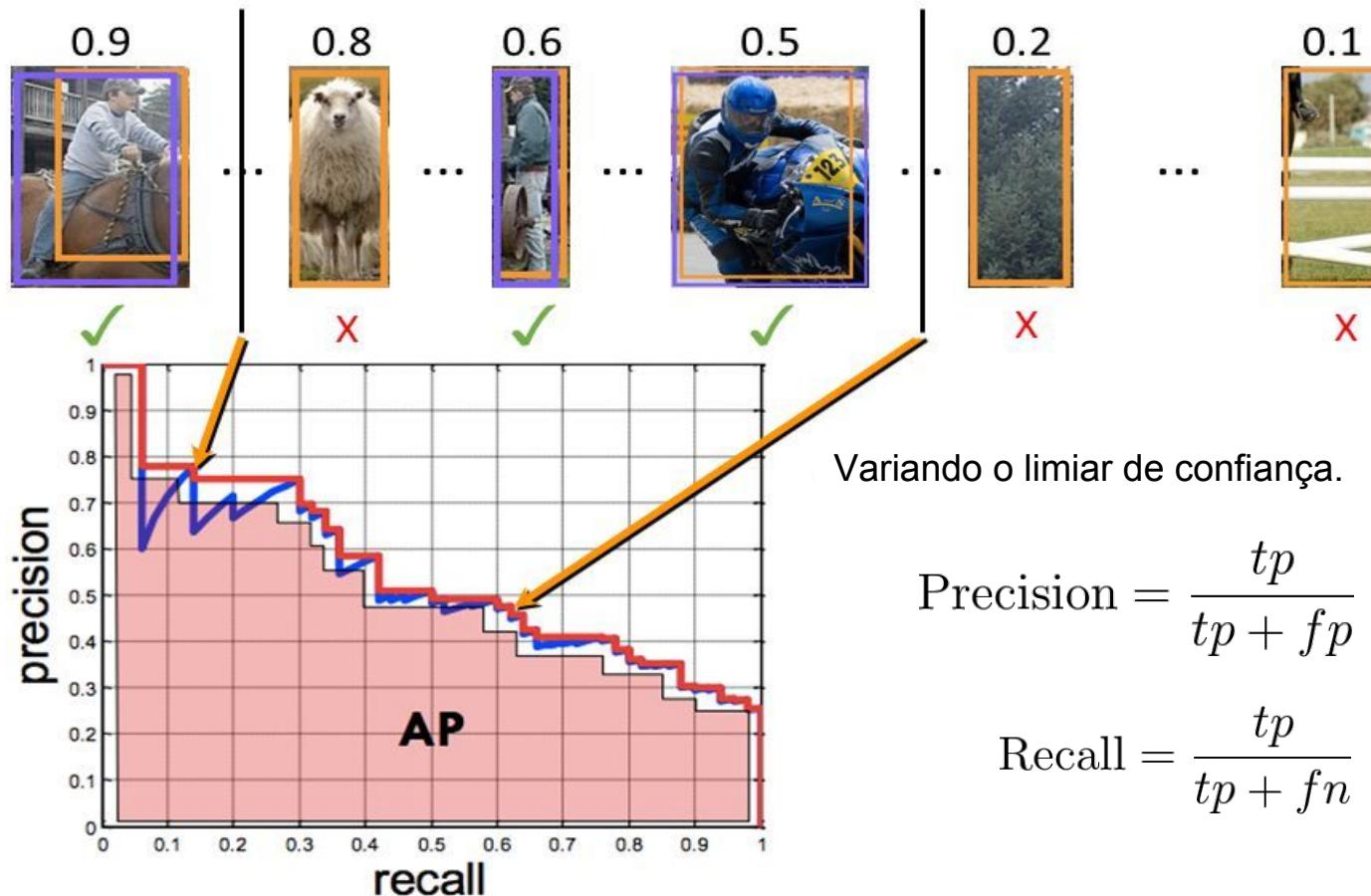
Verdadeiro
Positivo

$\text{IoU} \geq 0.5$

Falso
Positivo

$\text{IoU} < 0.5$ ou $= 0.0$
Duplicidades

MAP



YOLO - You Only Look Once

Cuidado!

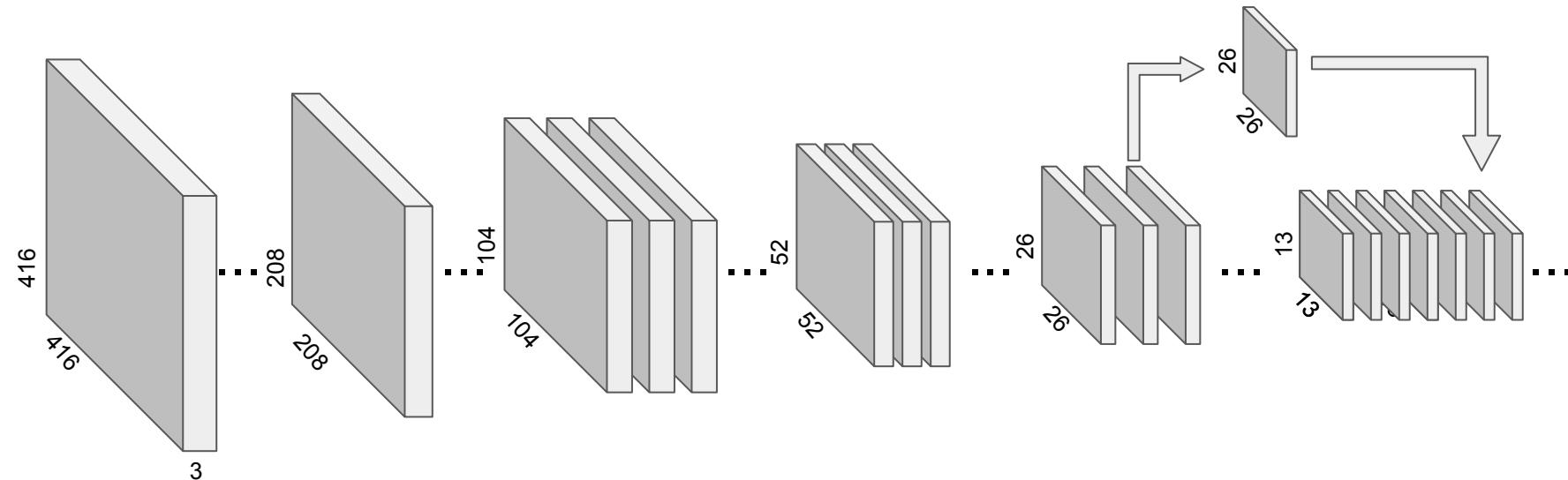
YOLO não é uma rede! É um método para detecção de objetos de passada única que utiliza uma rede convolucional como extrator de *features*.

YOLO - You Only Look Once

YOLO gera seus boxes partindo de **dimensões pré-definidas** a de objetos anotados no conjunto de treinamento. Estas dimensões pré-definidas são conhecidas como **âncoras** e estão diretamente relacionadas com a quantidade de **detectores** utilizados.

Extrator de *Features* (Darknet19)

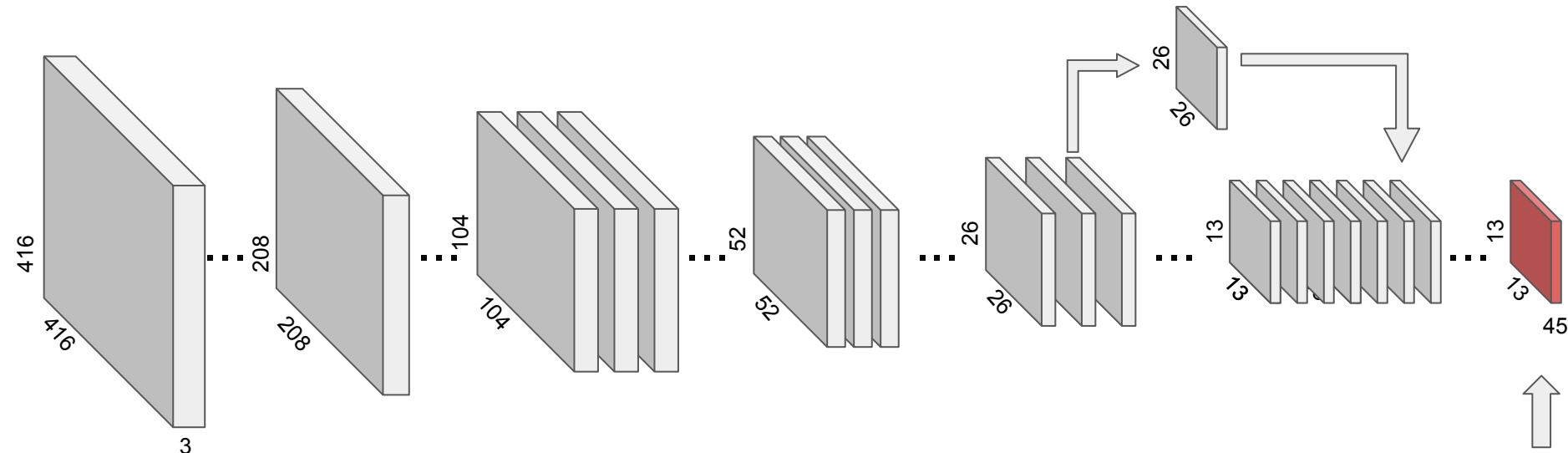
$$IN = 416 \times 416 \times 3$$



Sequência de convoluções (3×3) e (1×1) com $stride = 1$ e compensação de dimensão (*padding*), intercaladas por *maxpooling* para redução de dimensionalidade.

Extrator de *Features* (Darknet19)

$$IN = 416 \times 416 \times 3$$



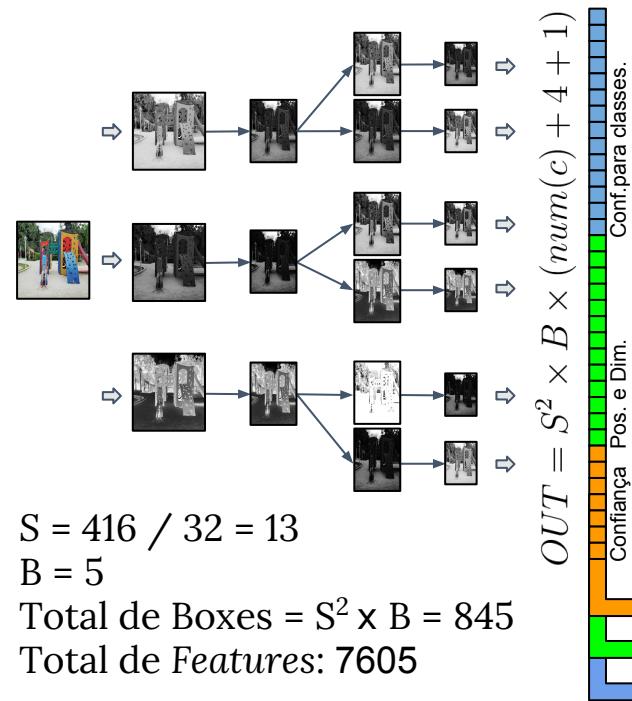
Sequência de convoluções (3×3) e (1×1) com *stride* = 1 e compensação de dimensão (*padding*), intercaladas por *maxpooling* para redução de dimensionalidade.

Assumindo um problema de 4 classes.

$$OUT = S^2 \times B \times (\text{num}(c) + 4 + 1)$$

Saída da Rede e Montagem dos Boxes

Dadas as Classes: {car, dog, motorbike, person}



Saída da Rede e Montagem dos Boxes

Dadas as Classes: {car, dog, motorbike, person}

The diagram illustrates a multi-scale feature extraction process. It shows three input images at different scales being processed by a backbone network. Each scale produces a set of feature maps. These feature maps are then processed by a head network to generate bounding boxes and class probabilities. The final output is a tensor of shape $S^2 \times B \times (num(c) + 4 + 1)$, where $S = 224 / 32 = 7$, $B = 5$, and $num(c)$ is the number of classes. The output is represented as a grid of colored boxes, with a legend indicating colors for confidence, position, and dimension.

$S = 224 / 32 = 7$

$B = 5$

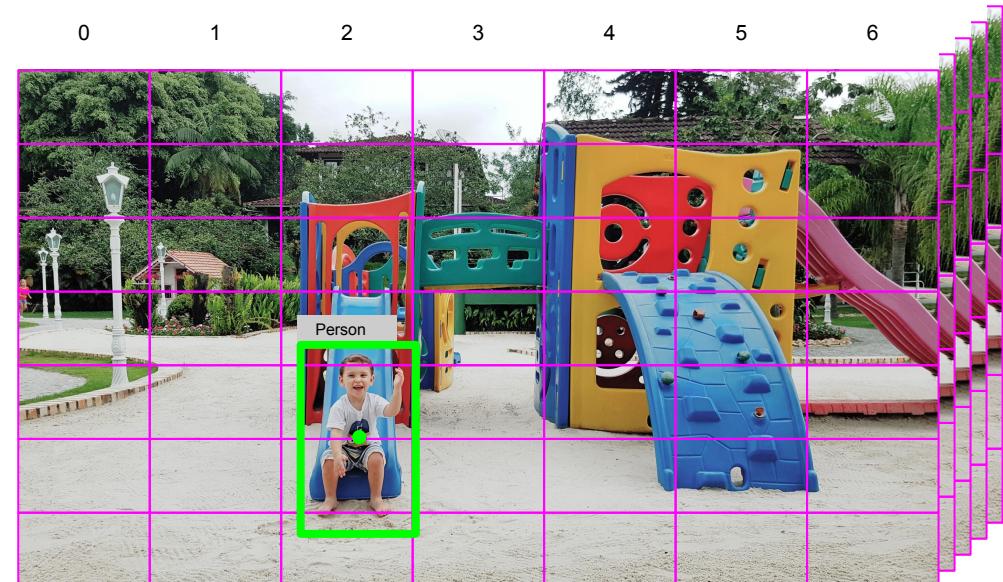
Total de Boxes = $S^2 \times B = 245$

Total de Features: 2205

$OUT = S^2 \times B \times (num(c) + 4 + 1)$

Conf. para classes

Confiância Pos. e Dim.



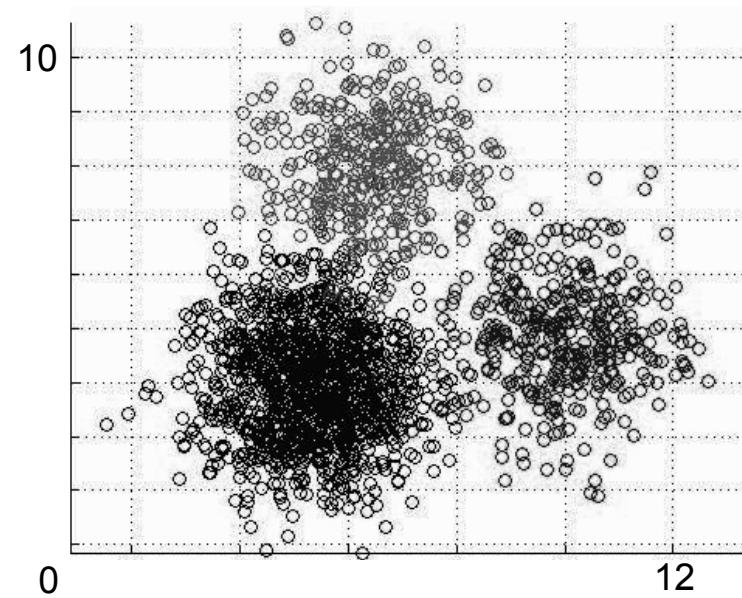
Confiança 0.78 (Sigmoide)

Pos. (2,4), detec. 0, centro (0.6, 0.95), Dim. 0.9x2.5 (Linear)

Scores {0.02, 0.12, 0.11, 0.75} (Softmax)

Âncoras

Calculado sobre todos os boxes do conjunto de treinamento.

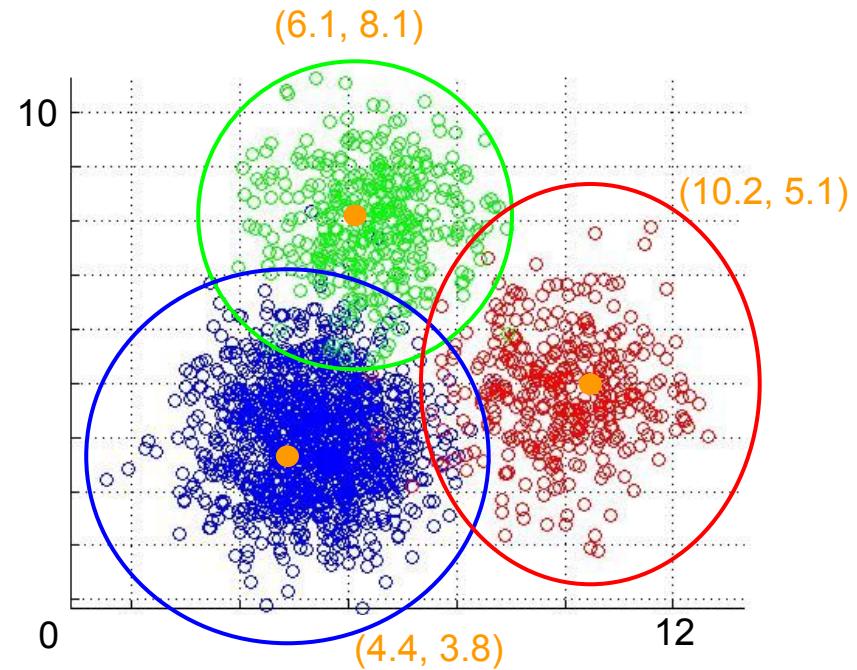


Âncoras

Calculado sobre todos os boxes do conjunto de treinamento.

O conjunto é clusterizado de acordo com o número de detectores que será aplicado no método. Por padrão, 5 clusters.

Os centróides de cada cluster são utilizados como âncoras para cada detector.



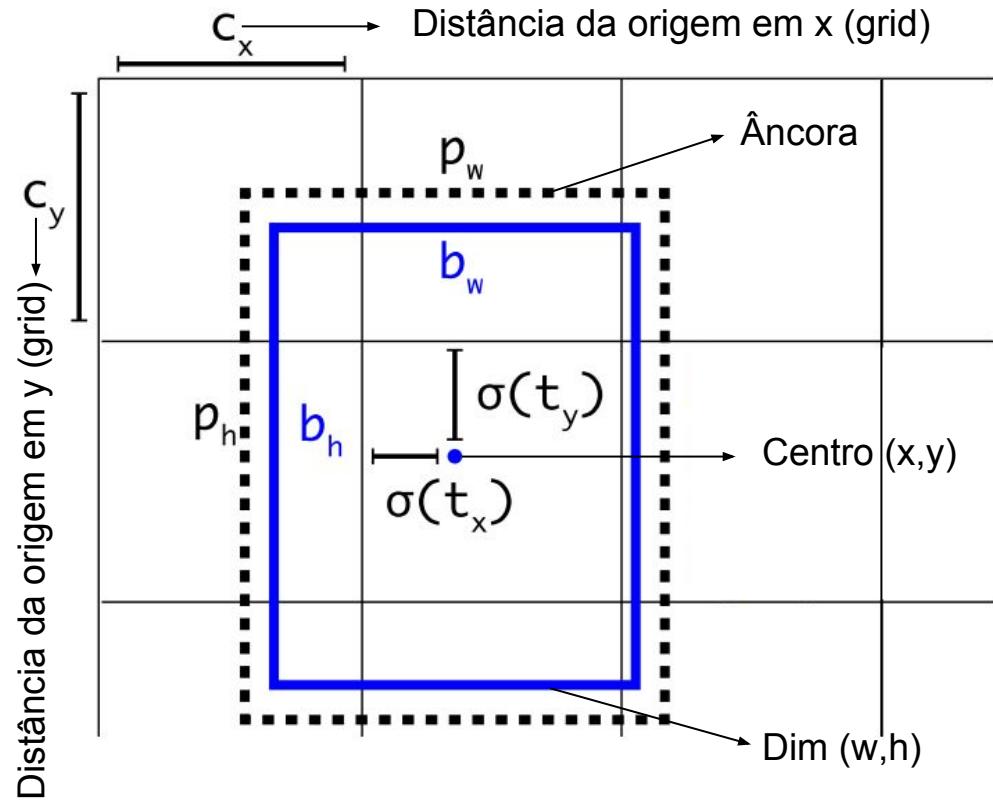
Montagem de um Box

$$b_x = \sigma(t_x) + c_x$$

$$b_y = \sigma(t_y) + c_y$$

$$b_w = p_w e^{t_w}$$

$$b_h = p_h e^{t_h}$$



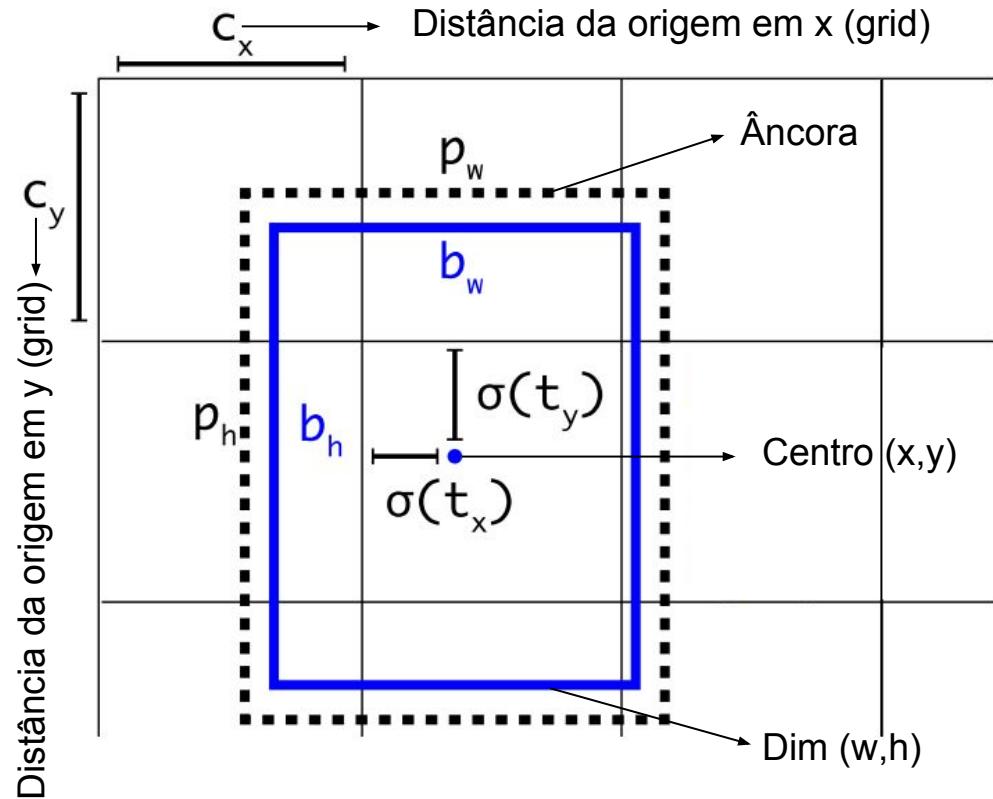
Montagem de um Box

$$b_x = \sigma(t_x) + c_x$$

$$b_y = \sigma(t_y) + c_y$$

$$b_w = p_w e^{t_w}$$

$$b_h = p_h e^{t_h}$$



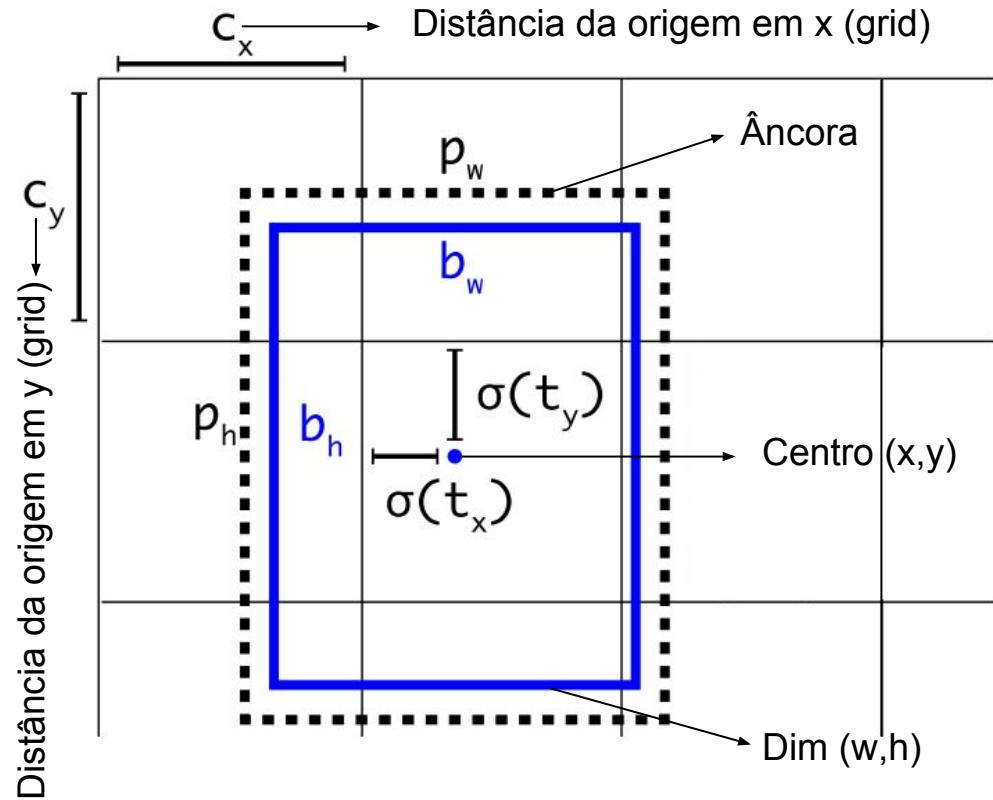
Montagem de um Box

$$b_x = \sigma(t_x) + c_x$$

$$b_y = \sigma(t_y) + c_y$$

$$b_w = p_w e^{\sigma(t_w)}$$

$$b_h = p_h e^{\sigma(t_h)}$$



Treinamento

O treinamento da rede parte do modelo **pré-treinado** com imagenet.

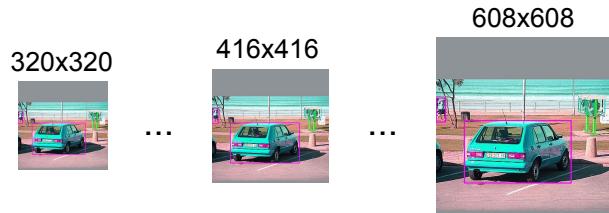
O método de treinamento segue o clássico *backpropagation* utilizando os otimizadores **SGD** (na implementação darknet) e **ADAM** em outras implementações observadas.

A quantidade de épocas de treinamento varia de acordo com o dataset. Para o PASCAL VOC, foram 135 épocas variando a taxa de aprendizado entre 10^{-2} e 10^{-4} .

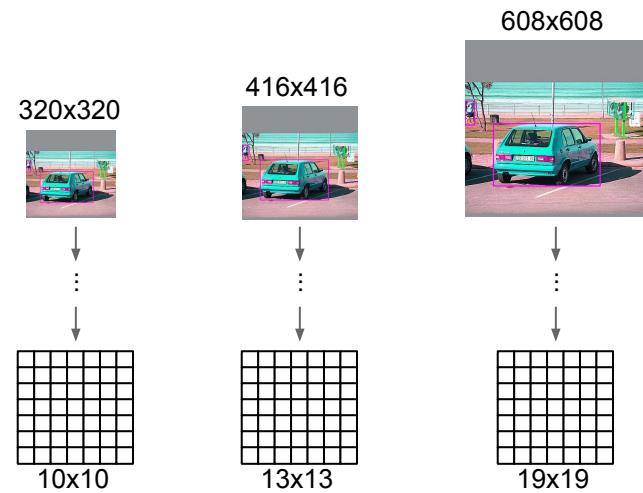
As dimensões da entrada **variam** a cada 10 batches.

Aplica um intenso **aumento de dados!**

Treinamento Multiescala



A imagem de entrada é escalada a cada 10 batches. As possíveis dimensões partem de 320x320 até 608x608, em múltiplos de 32.



Lembre que a dimensão de entrada influencia no grid de saída!

Predições Multiescala

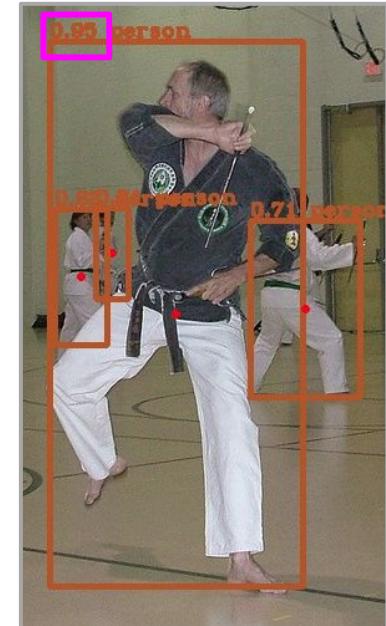
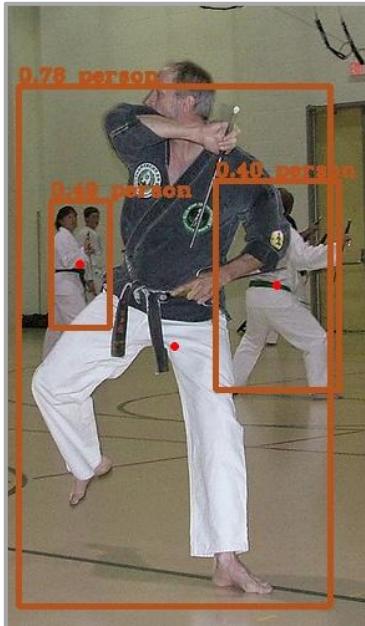
320x320



416x416



608x608



Aumento de Dados (*Data Augmentation*)

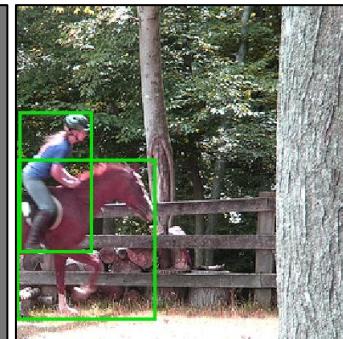
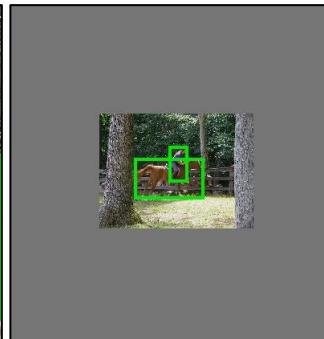
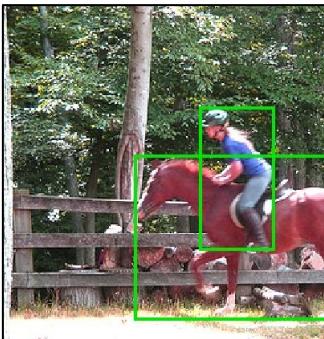
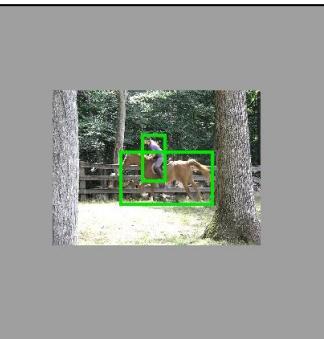
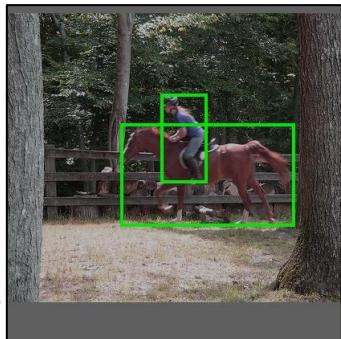
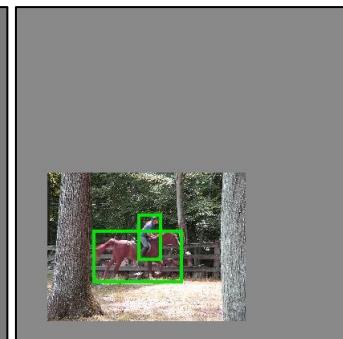
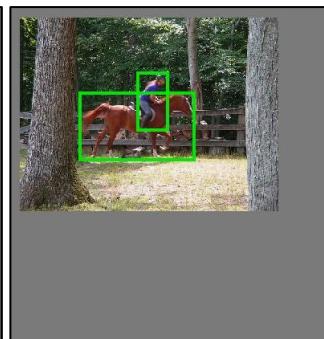
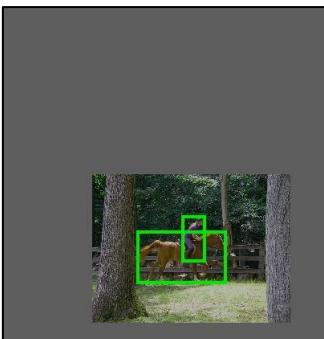
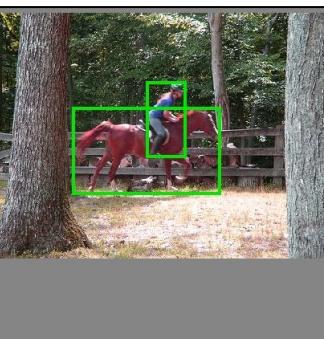
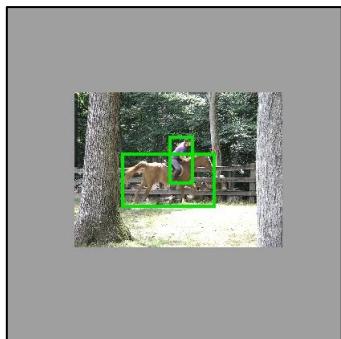


Aumento de Dados (*Data Augmentation*)

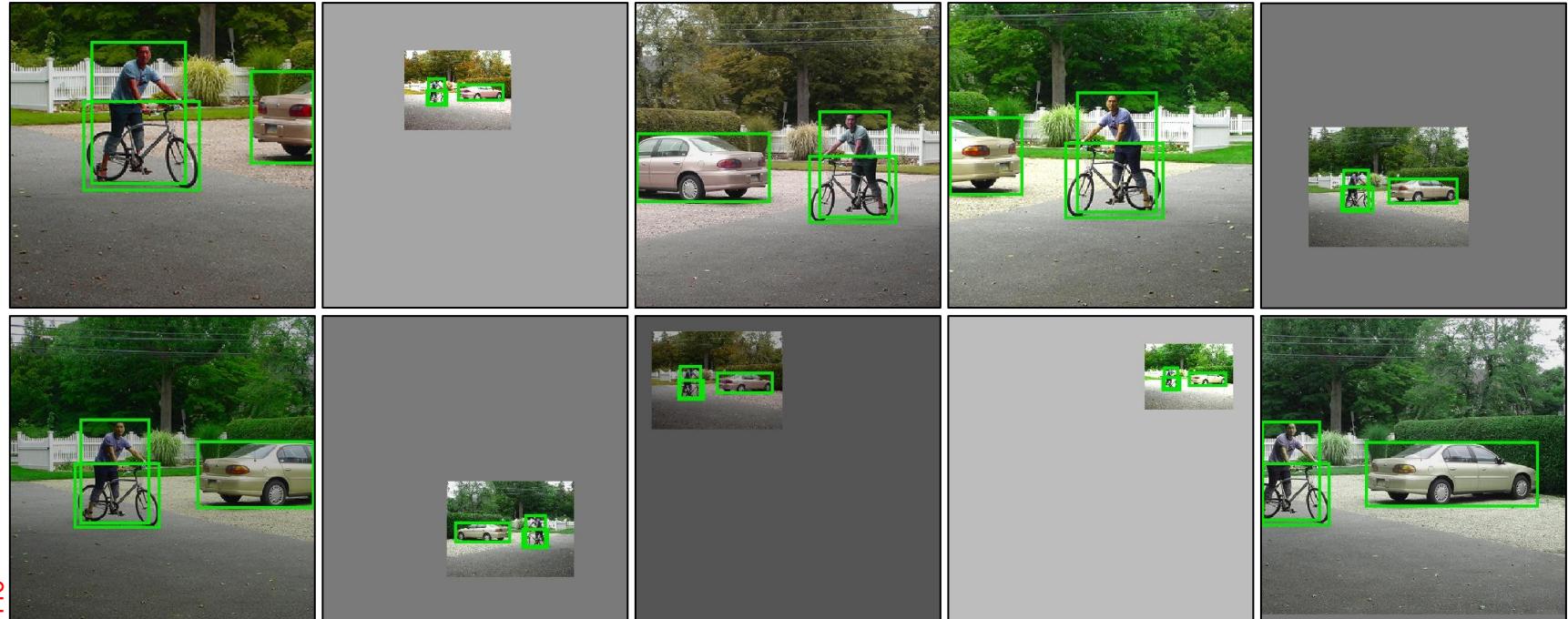


Área 78% menor que as outras.

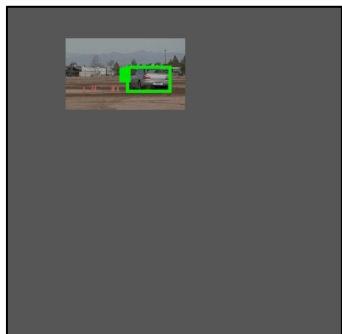
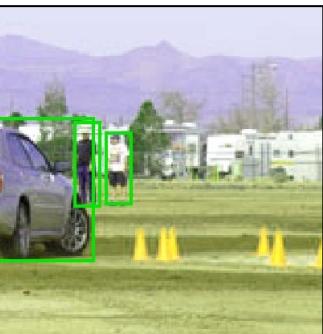
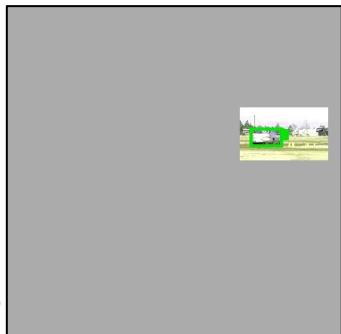
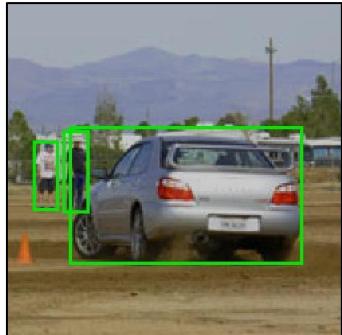
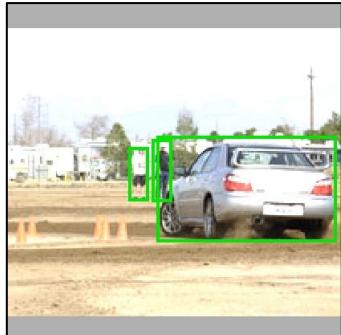
Aumento de Dados (*Data Augmentation*)



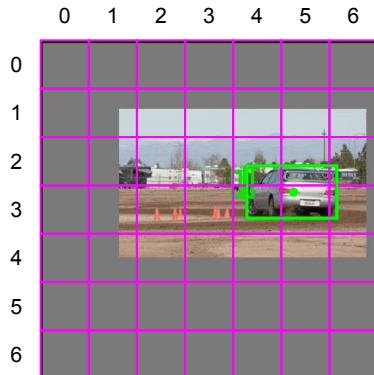
Aumento de Dados (*Data Augmentation*)



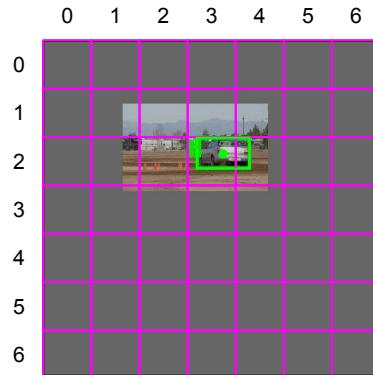
Aumento de Dados (*Data Augmentation*)



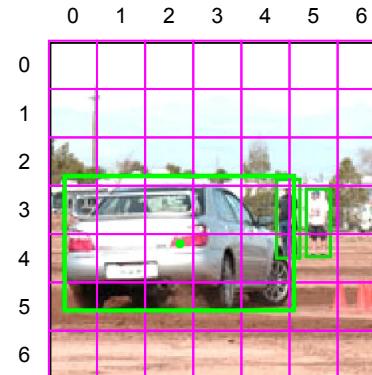
Aumento de Dados (*Data Augmentation*)



Pos. (5.3, 3.15)
Dim. 1.85x1.2



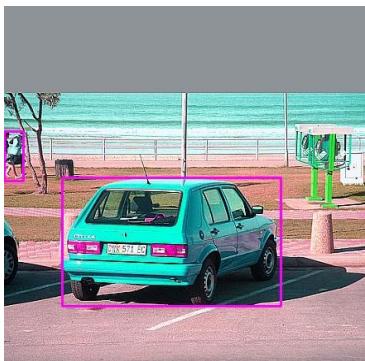
Pos. (3.8, 2.4)
Dim. 1.15x0.75



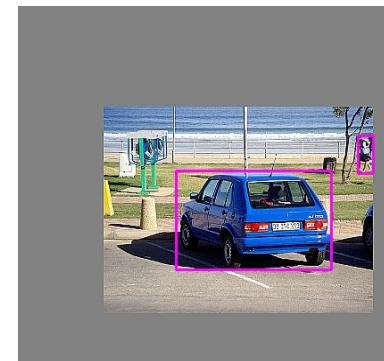
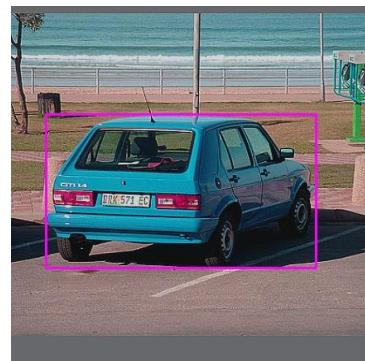
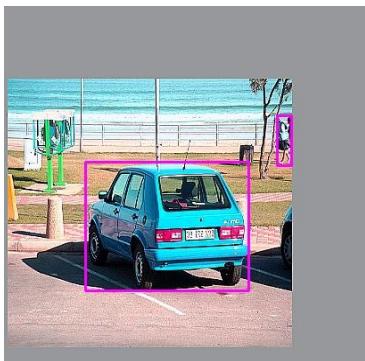
Pos. (2.7, 4.15)
Dim. 4.5x2.8

Aumento de Dados (*Data Augmentation*)

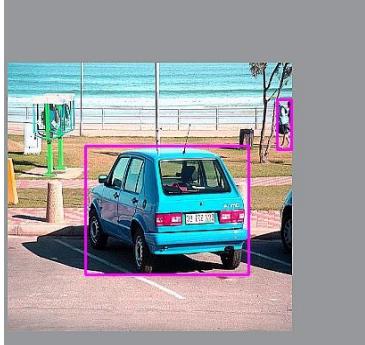
Escala: 0.25 to 2.00



Jitter: 0.2



Brilho, contraste e
saturação



Função de Custo

YOLO otimiza uma função de custo de múltiplas partes que aplica regras distintas para penalizar **confiança**, **localização**, **dimensão** e **probabilidades por classe**, tanto para situações onde exista ou não um objeto.

Função de Custo

$$\begin{aligned} & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\ & + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\ & + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\ & + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\ & + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2 \end{aligned}$$

Função de Custo

$$\begin{aligned}
& \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
& + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
& + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\
& + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\
& + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
\end{aligned}$$

x	coordenada no eixo X
y	coordenada no eixo y
w	largura
h	altura
C	confiança = $c_{\text{pred}} \times IoU_{\text{pred}}^{\text{truth}}$
$p_i(c)$	probabilidade dada a classe c na célula i
S	dimensão do grid de saída
B	quantidade de detectores
$\mathbb{1}_{ij}^{\text{obj}}$	1, caso o detector da posição i, j seja um objeto, caso contrário 0.
$\mathbb{1}_{ij}^{\text{noobj}}$	0, caso o detector da posição i, j seja um objeto, caso contrário 1.
$\mathbb{1}_i^{\text{obj}}$	1, caso exista um objeto na célula i , caso contrário 0.
λ_{coord}	penaliza o custo por coordenadas e dimensões dos boxes existente. (=5)
λ_{noobj}	suaviza o custo para a confiança de detectores que não contém objetos. (=0.5)

Função de Custo

$$\begin{aligned}
& \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
& + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \right] \\
& + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} (C_i - \hat{C}_i)^2 \\
& + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} (C_i - \hat{C}_i)^2 \\
& + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
\end{aligned}$$

x	coordenada no eixo X
y	coordenada no eixo y
w	largura
h	altura
C	confiança = $c_{\text{pred}} \times IoU_{\text{pred}}^{\text{truth}}$
$p_i(c)$	probabilidade dada a classe c na célula i
S	dimensão do grid de saída
B	quantidade de detectores
$\mathbb{1}_{ij}^{\text{obj}}$	1, caso o detector da posição i, j seja um objeto, caso contrário 0.
$\mathbb{1}_{ij}^{\text{noobj}}$	0, caso o detector da posição i, j seja um objeto, caso contrário 1.
$\mathbb{1}_i^{\text{obj}}$	1, caso exista um objeto na célula i , caso contrário 0.
λ_{coord}	penaliza o custo por coordenadas e dimensões dos boxes existente. (=5)
λ_{noobj}	suaviza o custo para a confiança de detectores que não contém objetos. (=0.5)

Função de Custo

$$\begin{aligned}
& \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
& + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
& + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\
& + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\
& + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
\end{aligned}$$

x	coordenada no eixo X
y	coordenada no eixo y
w	largura
h	altura
C	confiança = $c_{\text{pred}} \times IoU_{\text{pred}}^{\text{truth}}$
$p_i(c)$	probabilidade dada a classe c na célula i
S	dimensão do grid de saída
B	quantidade de detectores
$\mathbb{1}_{ij}^{\text{obj}}$	1, caso o detector da posição i, j seja um objeto, caso contrário 0.
$\mathbb{1}_{ij}^{\text{noobj}}$	0, caso o detector da posição i, j seja um objeto, caso contrário 1.
$\mathbb{1}_i^{\text{obj}}$	1, caso exista um objeto na célula i , caso contrário 0.
λ_{coord}	penaliza o custo por coordenadas e dimensões dos boxes existente. (=5)
λ_{noobj}	suaviza o custo para a confiança de detectores que não contém objetos. (=0.5)

Função de Custo

$$\begin{aligned}
 & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
 & + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
 & + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\
 & + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\
 & + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
 \end{aligned}$$

x	coordenada no eixo X
y	coordenada no eixo y
w	largura
h	altura
C	confiança = $c_{\text{pred}} \times IoU_{\text{pred}}^{\text{truth}}$
$p_i(c)$	probabilidade dada a classe c na célula i
S	dimensão do grid de saída
B	quantidade de detectores
$\mathbb{1}_{ij}^{\text{obj}}$	1, caso o detector da posição i, j seja um objeto, caso contrário 0.
$\mathbb{1}_{ij}^{\text{noobj}}$	0, caso o detector da posição i, j seja um objeto, caso contrário 1.
$\mathbb{1}_i^{\text{obj}}$	1, caso exista um objeto na célula i , caso contrário 0.
λ_{coord}	penaliza o custo por coordenadas e dimensões dos boxes existente. (=5)
λ_{noobj}	suaviza o custo para a confiança de detectores que não contém objetos. (=0.5)

Função de Custo

$$\begin{aligned}
 & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
 & + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
 & + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \boxed{\left(C_i - \hat{C}_i \right)^2} \\
 & + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\
 & + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
 \end{aligned}$$

x	coordenada no eixo X
y	coordenada no eixo y
w	largura
h	altura
C	confiança = $c_{\text{pred}} \times IoU_{\text{pred}}^{\text{truth}}$
$p_i(c)$	probabilidade dada a classe c na célula i
S	dimensão do grid de saída
B	quantidade de detectores
$\mathbb{1}_{ij}^{\text{obj}}$	1, caso o detector da posição i, j seja um objeto, caso contrário 0.
$\mathbb{1}_{ij}^{\text{noobj}}$	0, caso o detector da posição i, j seja um objeto, caso contrário 1.
$\mathbb{1}_i^{\text{obj}}$	1, caso exista um objeto na célula i , caso contrário 0.
λ_{coord}	penaliza o custo por coordenadas e dimensões dos boxes existente. (=5)
λ_{noobj}	suaviza o custo para a confiança de detectores que não contém objetos. (=0.5)

Função de Custo

$$\begin{aligned}
& \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
& + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
& + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\
& + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \boxed{\left(C_i - \hat{C}_i \right)^2} \\
& + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
\end{aligned}$$

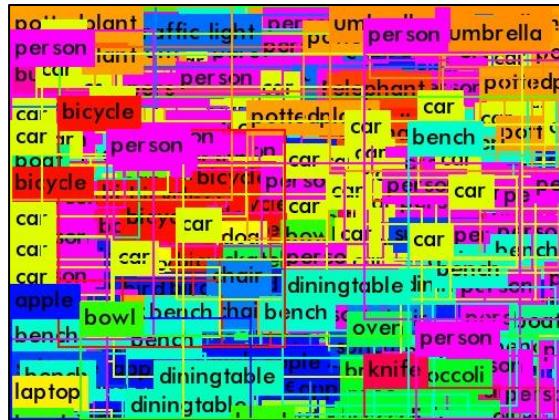
x	coordenada no eixo X
y	coordenada no eixo y
w	largura
h	altura
C	confiança = $c_{\text{pred}} \times IoU_{\text{pred}}^{\text{truth}}$
$p_i(c)$	probabilidade dada a classe c na célula i
S	dimensão do grid de saída
B	quantidade de detectores
$\mathbb{1}_{ij}^{\text{obj}}$	1, caso o detector da posição i, j seja um objeto, caso contrário 0.
$\mathbb{1}_{ij}^{\text{noobj}}$	0, caso o detector da posição i, j seja um objeto, caso contrário 1.
$\mathbb{1}_i^{\text{obj}}$	1, caso exista um objeto na célula i , caso contrário 0.
λ_{coord}	penaliza o custo por coordenadas e dimensões dos boxes existente. (=5)
λ_{noobj}	suaviza o custo para a confiança de detectores que não contém objetos. (=0.5)

Função de Custo

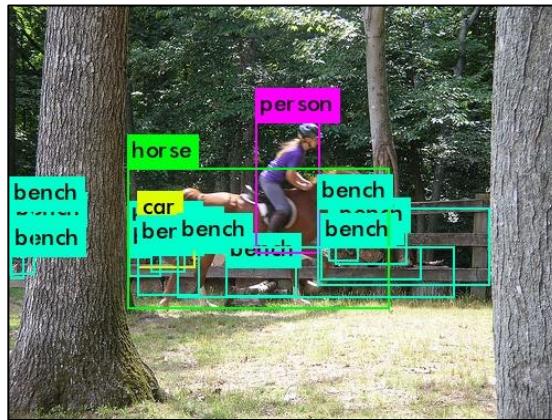
$$\begin{aligned}
& \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
& + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
& + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\
& + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\
& + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
\end{aligned}$$

x	coordenada no eixo X
y	coordenada no eixo y
w	largura
h	altura
C	confiança = $c_{\text{pred}} \times IoU_{\text{pred}}^{\text{truth}}$
$p_i(c)$	probabilidade dada a classe c na célula i
S	dimensão do grid de saída
B	quantidade de detectores
$\mathbb{1}_{ij}^{\text{obj}}$	1, caso o detector da posição i, j seja um objeto, caso contrário 0.
$\mathbb{1}_{ij}^{\text{noobj}}$	0, caso o detector da posição i, j seja um objeto, caso contrário 1.
$\mathbb{1}_i^{\text{obj}}$	1, caso exista um objeto na célula i , caso contrário 0.
λ_{coord}	penaliza o custo por coordenadas e dimensões dos boxes existente. (=5)
λ_{noobj}	suaviza o custo para a confiança de detectores que não contém objetos. (=0.5)

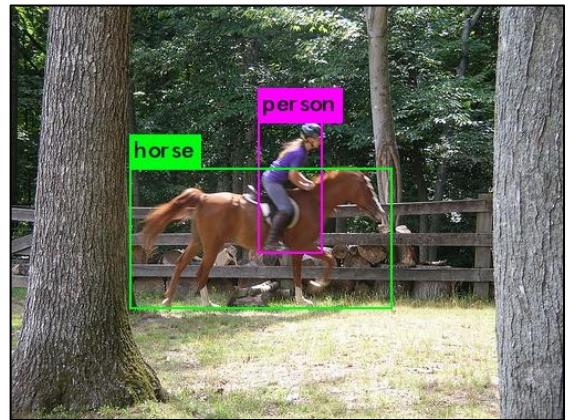
Observação do Limiar de Confiança



Limiar = 0

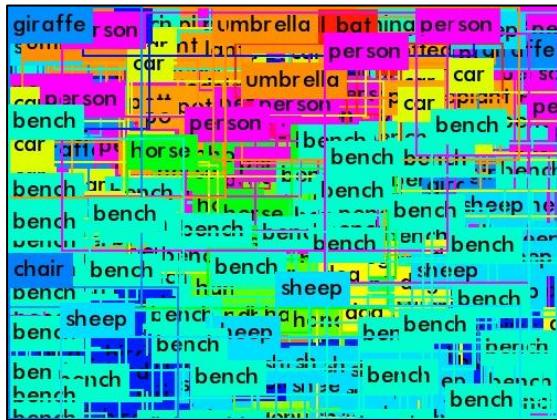


Limiar = 0.1

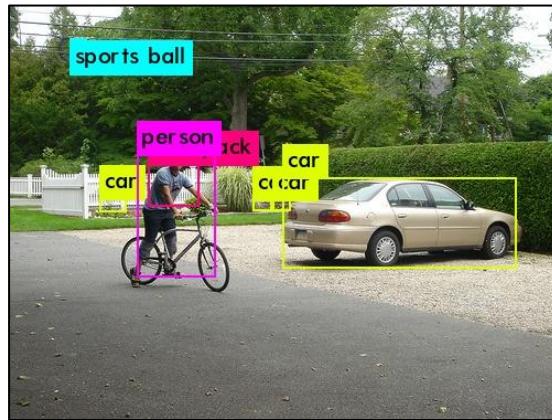


Limiar = 0.25

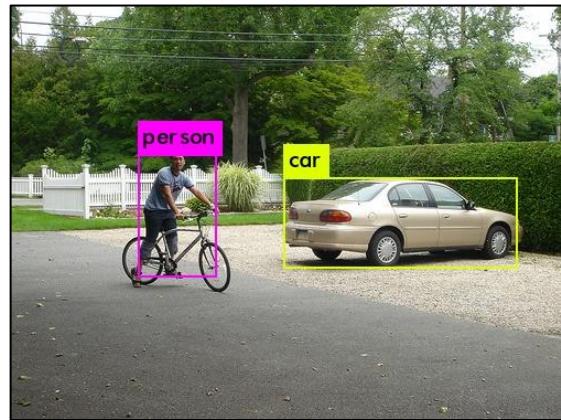
Observação do Limiar de Confiança



Limiar = 0



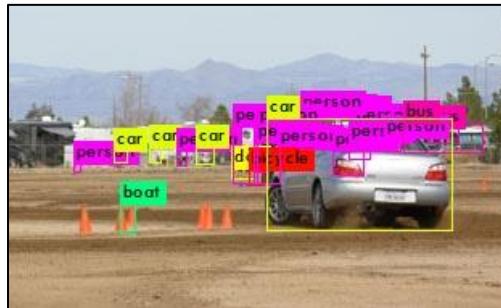
Limiar = 0.1



Limiar = 0.25

Observação do Limiar de Confiança

Límiar = 0



Límiar = 0.1

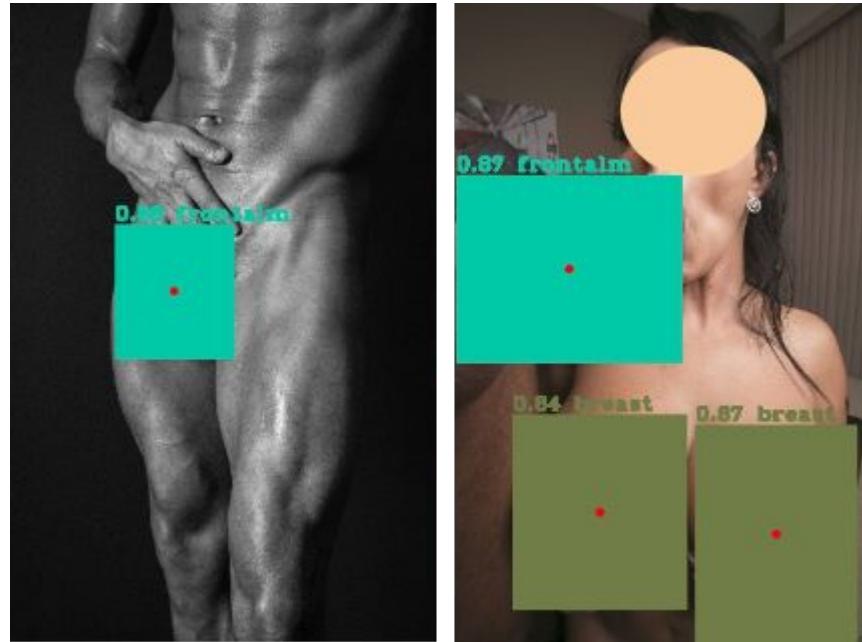


Límiar = 0.25

Exemplos (PASCAL VOC + Raccoons)



Exemplos (XDataset)



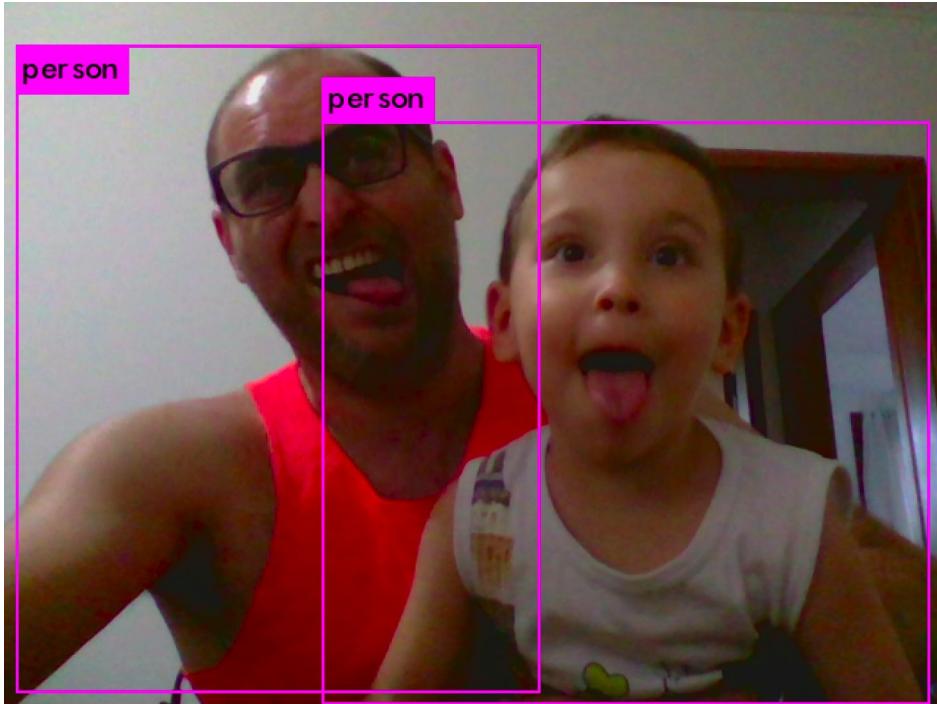
Implementações

- **YOLO sobre Darknet**
 - <https://github.com/pjreddie/darknet>
- **YAD2K: Yet Another Darknet 2 Keras**
 - <https://github.com/allanzelener/YAD2K>
- **Implementation of YOLO version 2 and 3 in Keras**
 - <https://github.com/experiencor/keras-yolo2>
 - <https://github.com/experiencor/keras-yolo3>

Artigos

- **You Only Look Once: Unified, Real-Time Object Detection**
 - <https://arxiv.org/abs/1506.02640>
- **YOLO9000: Better, Faster, Stronger**
 - <https://arxiv.org/abs/1612.08242>
- **YOLOv3: An Incremental Improvement**
 - <https://pjreddie.com/media/files/papers/YOLOv3.pdf>

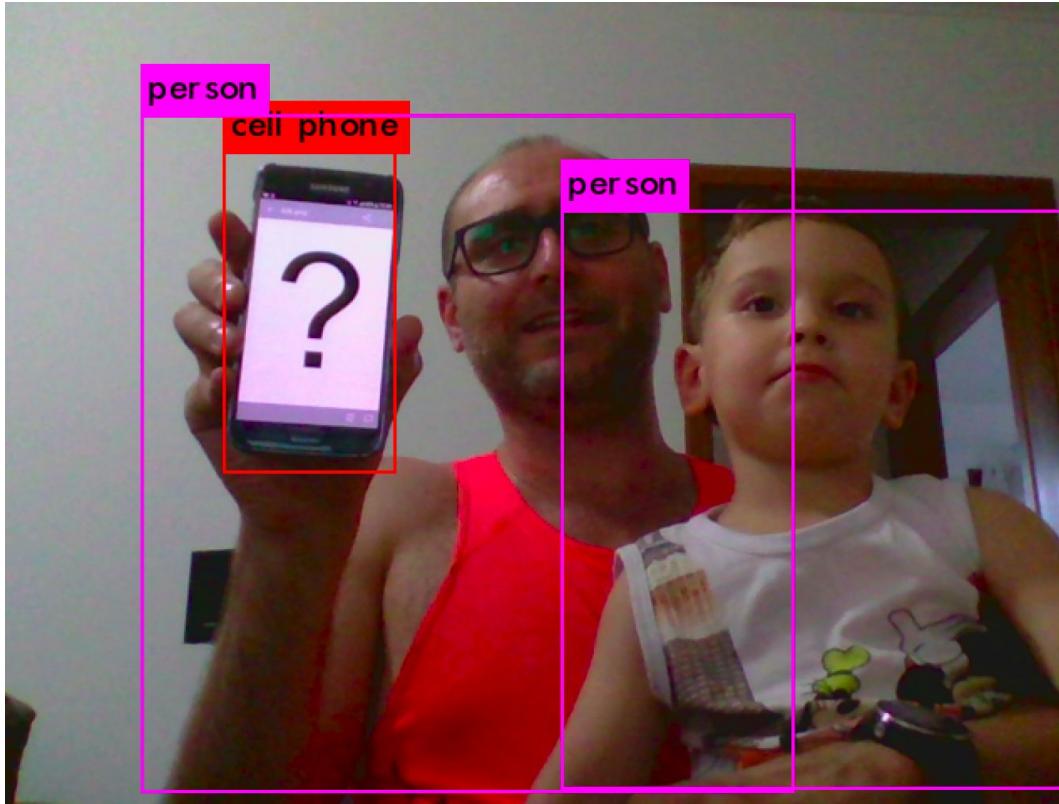
Exemplo de Execução Local



Ambiente:

- Core i7
- 16Gb RAM
- GPU GeForce 940MX 4Gb
- CUDA 8 + CUDNN 7
- ~3.5 Frames/s
- Darknet

Perguntas...



...obrigado!