

# **DEEP LEARNING BRASIL**

## **SUMMER SCHOOL**

# Region Based CNNs

**Francisco Calaça**

**AI Engineer at Deep Learning Brasil**

Patrocínio:



Saudemobi



[www.deeplearningbrasil.com.br](http://www.deeplearningbrasil.com.br)



Apoio:



# Region Based CNNs

**Francisco Calaça**

**AI Engineer at Deep Learning Brasil**

# Agenda

- O problema
- Estado da arte
- R-CNN
- Fast R-CNN
- Faster R-CNN
- Conclusão

airplane : 0.997

person : 0.867



- R-CNN - 2013,
- Fast RCNN - 2015,
- Faster R-CNN – 2015
- Pesquisadores
  - Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik
  - UC Berkeley

# Divisão dos problemas

- Reconhecimento de imagens (visual recognition)
- Classificação de imagens (image classification)

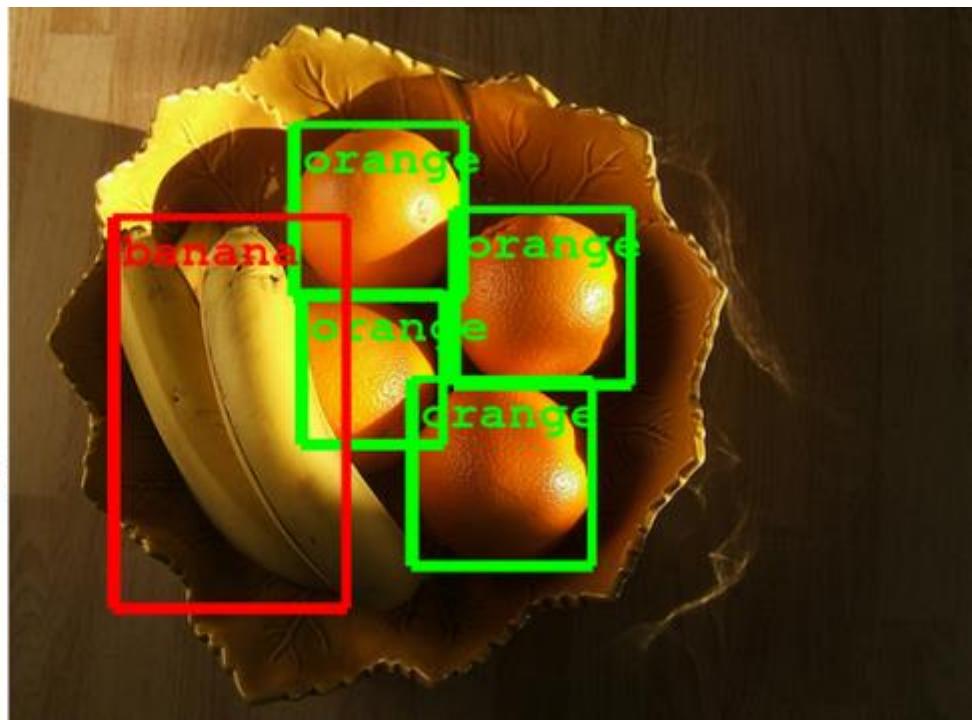
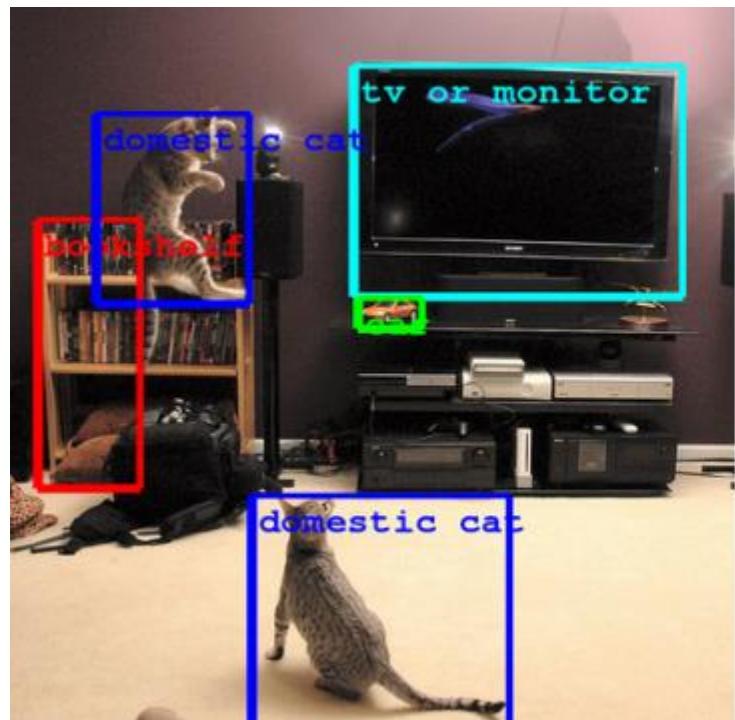
# Visual recognition



# Image classification



# O que queremos?



# Estado da arte – visual recognition

- Na última década, vários trabalhos de reconhecimento visual utilizaram SIFT ou HOG.
- SIFT (2003)– Scale Invariant Feature Transform
- HOG(2005) - Histograms of Oriented Gradients

DAVID G. LOWE

Computer Science Department, University of British Columbia, Vancouver, B.C., Canada

Lowe@cs.ubc.ca

*Received January 10, 2003; Revised January 7, 2004; Accepted January 22, 2004*

**Abstract.** This paper presents a method for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of an object or scene. The features are invariant to image scale and rotation, and are shown to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. The features are highly distinctive, in the sense that a single feature can be correctly matched with high probability against a large database of features from many images. This paper also describes an approach to using these features for object recognition. The recognition proceeds by matching individual features to a database of features from known objects using a fast nearest-neighbor algorithm, followed by a Hough transform to identify clusters belonging to a single object, and finally performing verification through least-squares solution for consistent pose parameters. This approach to recognition can robustly identify objects among clutter and occlusion while achieving near real-time performance.

**Keywords:** invariant features, object recognition, scale invariance, image matching

## 1. Introduction

Image matching is a fundamental aspect of many problems in computer vision, including object or scene recognition, solving for 3D structure from multiple images, stereo correspondence, and motion tracking. This paper describes image features that have many properties that make them suitable for matching differing images of an object or scene. The features are invariant to image scaling and rotation, and partially invariant to change in illumination and 3D camera viewpoint. They are well localized in both the spatial and frequency domains, reducing the probability of disruption by occlusion, clutter, or noise. Large numbers of features can be extracted from typical images with efficient algorithms. In addition, the features are highly distinctive, which allows a single feature to be correctly matched with high probability against a large database of features, providing a basis for object and scene recognition.

The cost of extracting these features is minimized by taking a cascade filtering approach, in which the more

expensive operations are applied only at locations that pass an initial test. Following are the major stages of computation used to generate the set of image features:

1. *Scale-space extrema detection:* The first stage of computation searches over all scales and image locations. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation.
2. *Keypoint localization:* At each candidate location, a detailed model is fit to determine location and scale. Keypoints are selected based on measures of their stability.
3. *Orientation assignment:* One or more orientations are assigned to each keypoint location based on local image gradient directions. All future operations are performed on image data that has been transformed relative to the assigned orientation, scale, and location for each feature, thereby providing invariance to these transformations.

Cited by [25688](#)

# Histograms of Oriented Gradients for Human Detection

Navneet Dalal and Bill Triggs

INRIA Rhône-Alpes, 655 avenue de l'Europe, Montbonnot 38334, France  
 {Navneet.Dalal,Bill.Triggs}@inrialpes.fr, http://lear.inrialpes.fr

## Abstract

We study the question of feature sets for robust visual object recognition, adopting linear SVM based human detection as a test case. After reviewing existing edge and gradient based descriptors, we show experimentally that grids of Histograms of Oriented Gradient (HOG) descriptors significantly outperform existing feature sets for human detection. We study the influence of each stage of the computation on performance, concluding that fine-scale gradients, fine orientation binning, relatively coarse spatial binning, and high-quality local contrast normalization in overlapping descriptor blocks are all important for good results. The new approach gives near-perfect separation on the original MIT pedestrian database, so we introduce a more challenging dataset containing over 1800 annotated human images with a large range of pose variations and backgrounds.

## 1 Introduction

Detecting humans in images is a challenging task owing to their variable appearance and the wide range of poses that they can adopt. The first need is a robust feature set that allows the human form to be discriminated cleanly, even in cluttered backgrounds under difficult illumination. We study the issue of feature sets for human detection, showing that locally normalized Histogram of Oriented Gradient (HOG) descriptors provide excellent performance relative to other existing feature sets including wavelets [17, 22]. The proposed descriptors are reminiscent of edge orientation histograms [4, 5], SIFT descriptors [12] and shape contexts [1], but they are computed on a dense grid of uniformly spaced cells and they use overlapping local contrast normalizations for improved performance. We make a detailed study of the effects of various implementation choices on detector performance, taking “pedestrian detection” (the detection of mostly visible people in more or less upright poses) as a test case. For simplicity and speed, we use linear SVM as a baseline classifier throughout the study. The new detectors give essentially perfect results on the MIT pedestrian test set [18, 17], so we have created a more challenging set containing over 1800 pedestrian images with a large range of poses and backgrounds. Ongoing work suggests that our feature set performs equally well for other shape-based object classes.

We briefly discuss previous work on human detection in §2, give an overview of our method in §3, describe our data sets in §4 and give a detailed description and experimental evaluation of each stage of the process in §5–6. The main conclusions are summarized in §7.

## 2 Previous Work

There is an extensive literature on object detection, but here we mention just a few relevant papers on human detection [18, 17, 22, 16, 20]. See [6] for a survey. Papageorgiou *et al* [18] describe a pedestrian detector based on a polynomial SVM using rectified Haar-wavelets as input descriptors, with a parts (subwindow) based variant in [17]. Depoortere *et al* give an optimized version of this [2]. Gavrila & Philomin [8] take a more direct approach, extracting edge images and matching them to a set of learned exemplars using chamfer distance. This has been used in a practical real-time pedestrian detection system [7]. Viola *et al* [22] build an efficient moving person detector, using AdaBoost to train a chain of progressively more complex region rejection rules based on Haar-like wavelets and space-time differences. Ronfard *et al* [19] build an articulated body detector by incorporating SVM based limb classifiers over 1<sup>st</sup> and 2<sup>nd</sup> order Gaussian filters in a dynamic programming framework similar to those of Fleet et al [3] and Isikoff & Forsyth [9]. Mikolajczyk *et al* [16] use combinations of orientation-position histograms with binary-thresholded gradient magnitudes to build a parts based method containing detection for faces, heads, and front and side profiles of upper and lower body parts. In contrast, our detector uses a simpler architecture with a single detection window, but appears to give significantly higher performance on pedestrian images.

## 3 Overview of the Method

This section gives an overview of our feature extraction chain, which is summarized in fig. 1. Implementation details are postponed until §6. The method is based on evaluating well-normalized local histograms of image gradient orientations in a dense grid. Similar features have seen increasing use over the past decade [4, 5, 12, 15]. The basic idea is that local object appearance and shape can often be characterized rather well by the distribution of local intensity gradients or

Cited by 8908

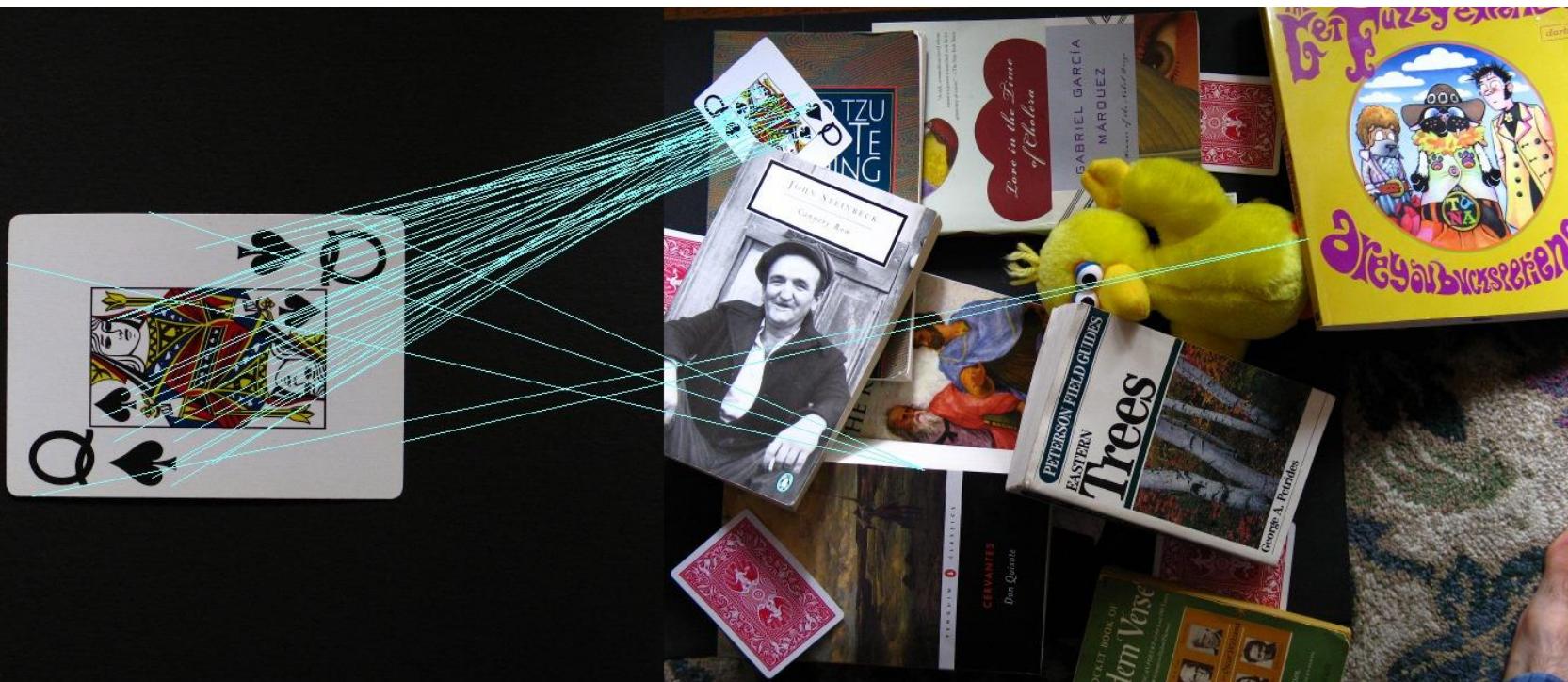
# Relevância SIFT e HOG

Title	1–20	Cited by	Year
<a href="#">Distinctive image features from scale-invariant keypoints</a> DG Lowe International journal of computer vision 60 (2), 91-110		37838	2004
Title	1–20	Cited by	Year
<a href="#">Histograms of oriented gradients for human detection</a> N Dalal, B Triggs 2005 IEEE Computer Society Conference on Computer Vision and Pattern ...		16547	2005
<a href="#">Human detection using oriented histograms of flow and appearance</a> N Dalal, B Triggs, C Schmid European conference on computer vision, 428-441		1108	2006

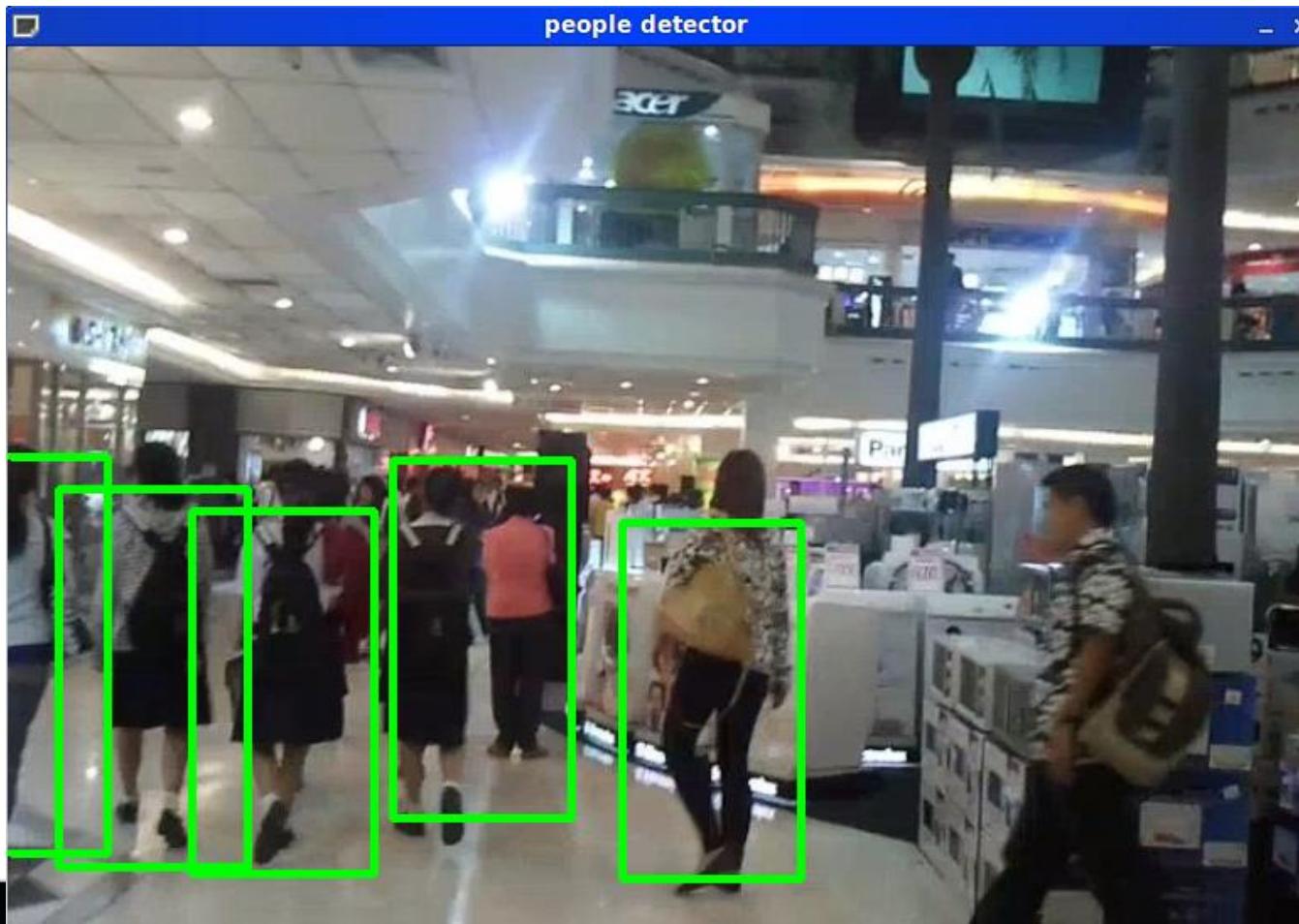
# Ferramentas

- OpenCV
  - Contém a implementação do SIFT e do HOG

# OpenCV - SIFT



# OpenCV - HOG



# Estado da Arte – Image classification

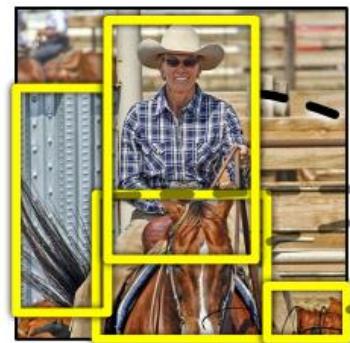
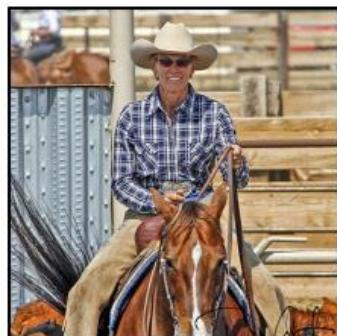
- CNN - Convolutional Neural Network
  - CNN começou a ser usada de forma pesada na década de 90
  - Em 2012 Alex Krizhevsky reacendeu o interesse em CNN depois que apresentou um significativo trabalho de classificação de imagens

# Relevância

Title	Cited by	Year
<a href="#">Imagenet classification with deep convolutional neural networks</a> A Krizhevsky, I Sutskever, GE Hinton Advances in neural information processing systems, 1097-1105	8194	2012

# A proposta do R-CNN

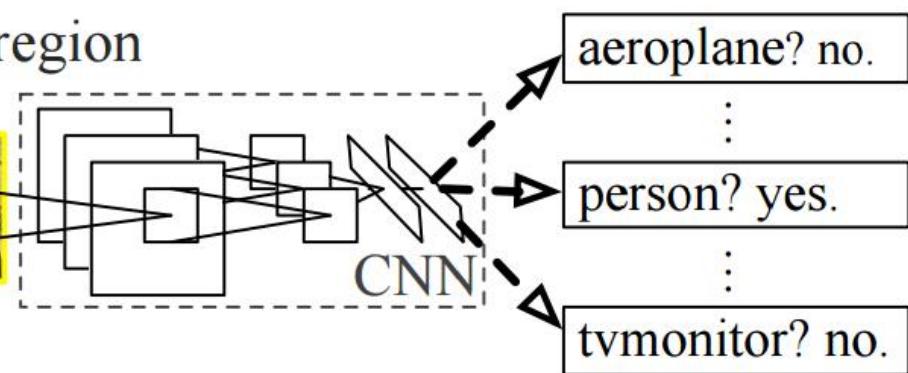
**R-CNN: *Regions with CNN features***



1. Input image

2. Extract region proposals (~2k)

warped region



3. Compute CNN features

4. Classify regions

# Object detection with R-CNN

- Três módulos
  - O primeiro gera regiões independentes de categorias. Essas são as regiões candidatas para o detector
  - O segundo é uma grande CNN que extrai as características das regiões.
  - O terceiro utiliza um classificador SVM a partir das características

# Módulo 1: descoberta das regiões

- R-CNN não define um algoritmo específico para essa tarefa. O autor identifica:
  - objectness
  - **selective search**
  - category-independent object proposals
  - constrained parametric min-cuts (CPMC)
  - multi-scale combinatorial grouping
  - Dentre outros

# Módulo 1: descoberta das regiões

- Para implementação dos resultados, o autor utilizou apenas o:
  - **selective search (2012)**
- Diversifica as estratégias: cor, textura, fechamento.

# Selective search



(a)



(b)



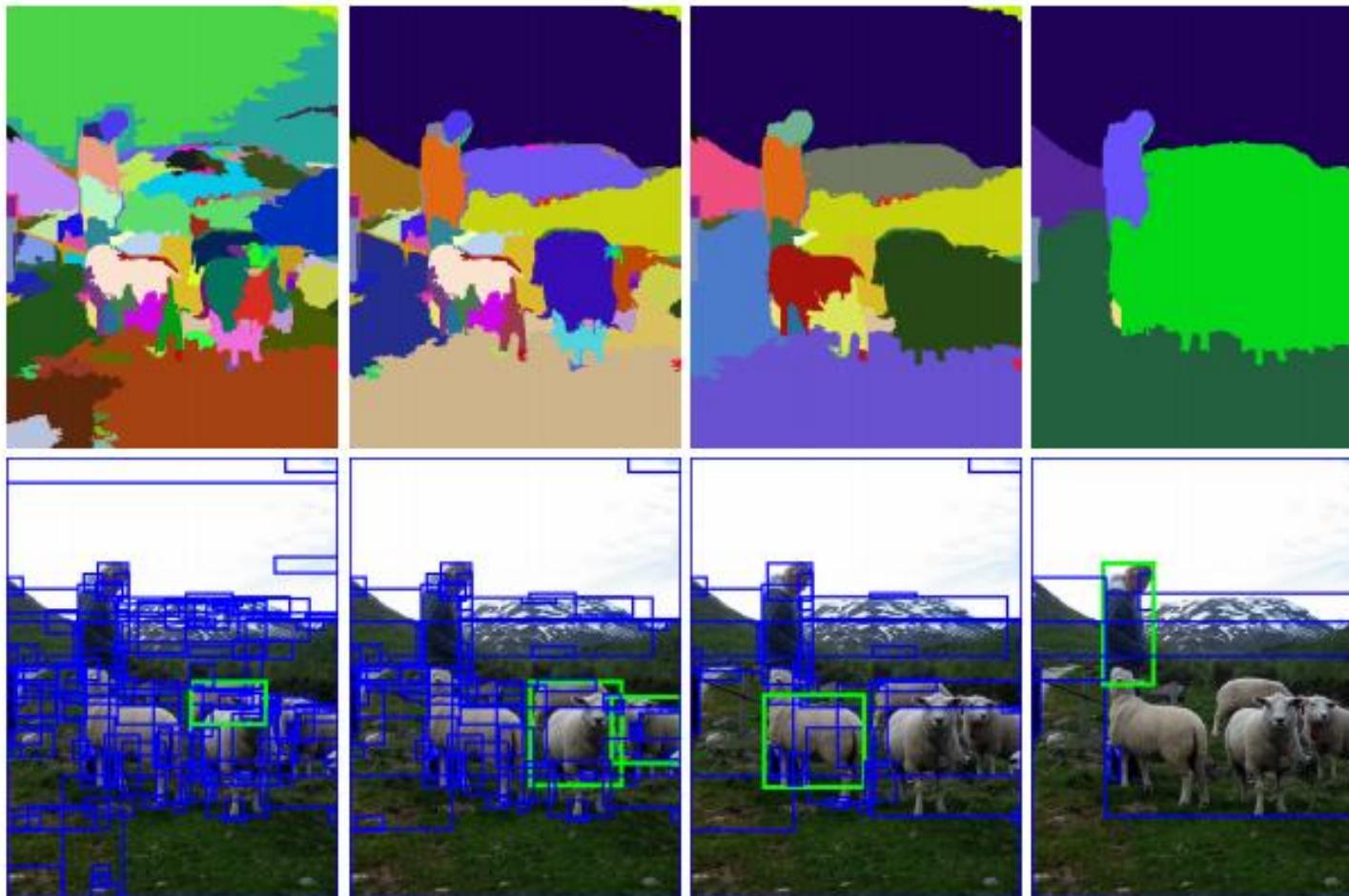
(c)



(d)

- b) Gatos podem ser distinguidos por cor
  - c) O camaleão pode ser distinguido por textura
  - d) As rodas podem ser distinguidas por ser uma figura fechada
- 
- a) Geralmente as imagens demandam uma combinação de estratégias

# Selective search



(a)

# Selective search



(b)

# Ferramentas sugestão!!!



Open Source Computer Vision

Main Page Related Pages Modules Namespaces ▾ Classes ▾ Files ▾ Examples Sphinx Documentation

cv ximgproc segmentation SelectiveSearchSegmentationStrategyFill

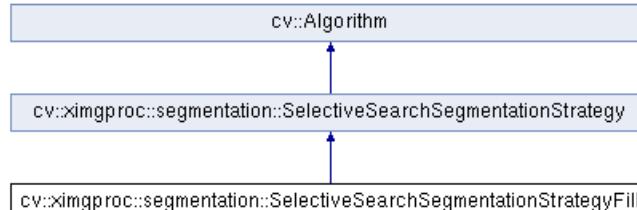
## cv::ximgproc::segmentation::SelectiveSearchSegmentationStrategyFill Class Reference

Extended Image Processing » Image segmentation

Fill-based strategy for the selective search segmentation algorithm. The class is implemented from the algorithm described in [156]. More...

```
#include "segmentation.hpp"
```

Inheritance diagram for cv::ximgproc::segmentation::SelectiveSearchSegmentationStrategyFill:



# Ferramentas sugestão!!!

[belltailjp / selective\\_search\\_py](#)

Code Issues 13 Pull requests 1 Projects 0 Wiki Pulse Graphs

99 commits 6 branches 0 releases 2 contributors MIT

Branch: master New pull request Create new file Upload files Find file Clone or download ▾

belltailjp Merge pull request #4 from paulinder/fix-segment-cmake ... Latest commit dbf9161 on 3 Aug 2015

doc	Added one more sample image	2 years ago
.gitignore	Ignore image files	2 years ago
CMakeLists.txt	specify which python-config, fixes undefined symbol	a year ago
LICENSE.txt	Added License	2 years ago
README.md	Supplementaly explained about license	2 years ago

# Módulo 2: Extração de características

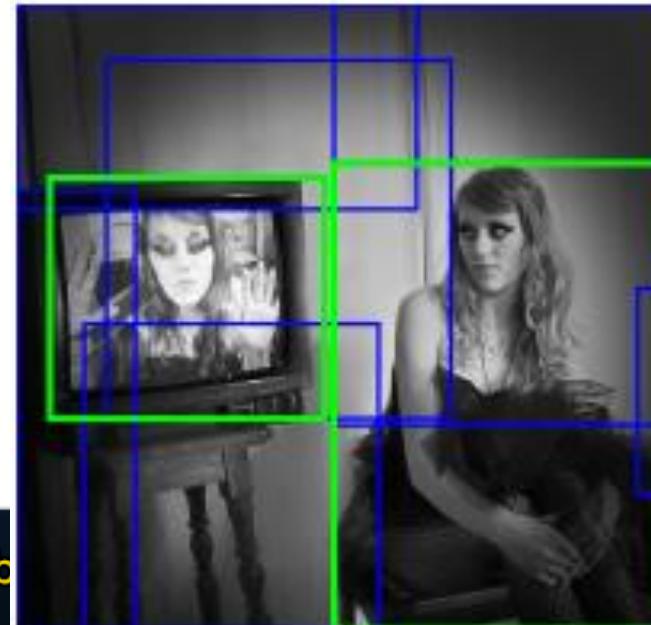
- Foi utilizada a **CNN de Alex Krizhevsky**
- Extraído um vetor de características com 4096
- Framework Caffe
- No exemplo, as imagens de entrada para a CNN tem dimensão  $227 \times 227$

# Módulo 2: Extração de características

- O **selective search** não retorna imagens com dimensões  $227 \times 227$ .
- Além do mais, a largura de uma região pode ser diferente da altura.
- O que fazer?



olearningb

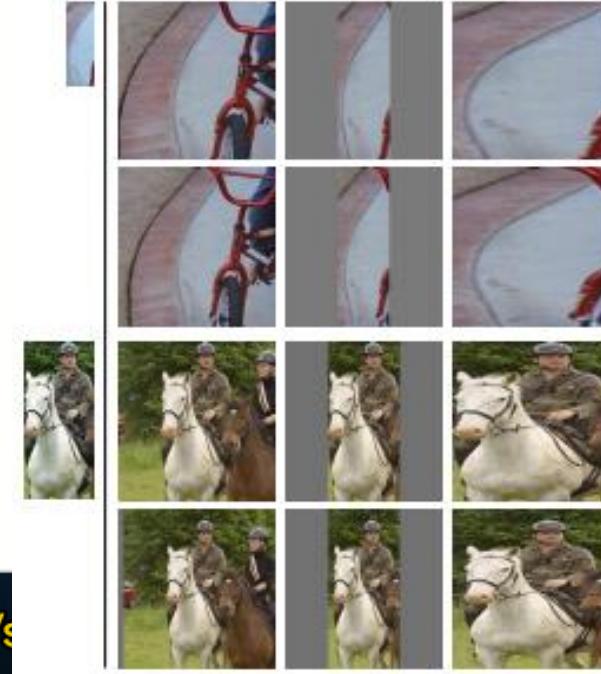


- Our dataset contains images of aeroplane



# Módulo 2: Extração de características

- O autor propõe outras estratégias, mas coloca a avaliação como trabalho futuro



# Módulo 2: Extração de características

- O autor utilizou a base **ILSVRC2012** para treinar a CNN no framework Caffe

# Ferramentas

## Caffe

Deep learning framework  
by the [BVLC](#)

Created by  
[Yangqing Jia](#)  
Lead Developer  
[Evan Shelhamer](#)

 [View On GitHub](#)

## Caffe Tutorial

Caffe is a deep learning framework and this tutorial explains its philosophy, architecture, and usage. This is a practical guide and framework introduction, so the full frontier, context, and history of deep learning cannot be covered here. While explanations will be given where possible, a background in machine learning and neural networks is helpful.

## Philosophy

In one sip, Caffe is brewed for

- Expression: models and optimizations are defined as plaintext schemas instead of code.
- Speed: for research and industry alike speed is crucial for state-of-the-art models and massive data.
- Modularity: new tasks and settings require flexibility and extension.
- Openness: scientific and applied progress call for common code, reference models, and reproducibility.
- Community: academic research, startup prototypes, and industrial applications all share strength by joint discussion and development in a BSD-2 project.

and these principles direct the project.

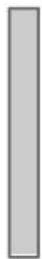
# Módulo 3: Classificação

- Utilização de um SVM
- Treinamento das classes à partir das características geradas pela CNN
- Todas as classificações são pontuadas e aplicadas de supressão dos não máximos.
  - Rejeita classificações sobrepostas

# Módulo 3: Classificação

- O autor utilizou a base ILSVRC2012 para treinar a CNN.

# Módulo 3: Classificação CNN + SVM



Positive samples for catSVM

Negative samples for catSVM

# Módulo 3: Classificação CNN + SVM



Negative samples for dogSVM  
[www.deeplearningbrasil.com.br/summerschool2018](http://www.deeplearningbrasil.com.br/summerschool2018)

Positive samples for dogSVM

# Bounding-box regression

- Reduz erros na detecção de objetos
- Treina um modelo de regressão linear para predizer janelas de detecção
- Ajusta as janelas de detecção para o tamanho ideal da imagem da região

# Bounding-box regression

Training image regions



Cached region features



Regression targets  
(dx, dy, dw, dh)  
Normalized coordinates

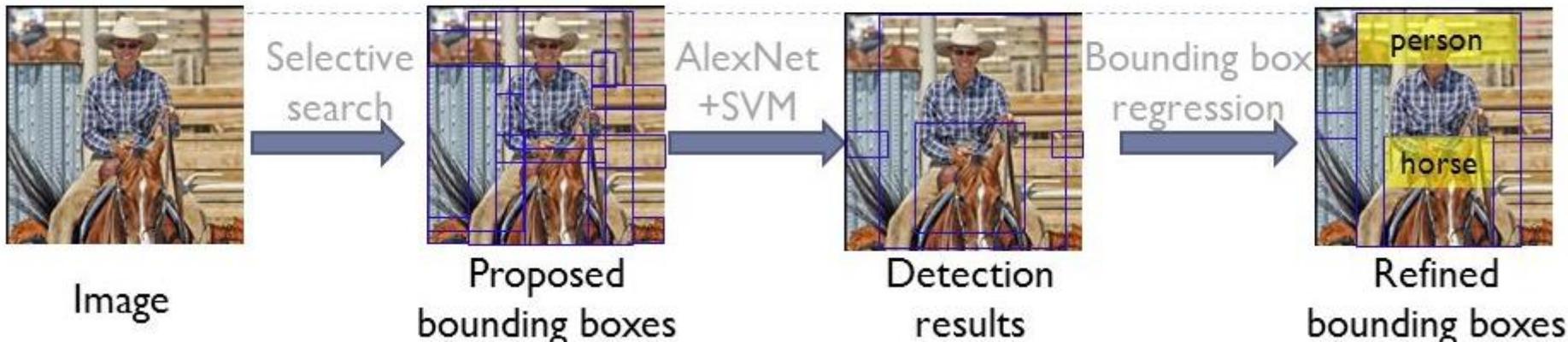
(0, 0, 0, 0)  
Proposal is good

(.25, 0, 0, 0)  
Proposal too far to left

(0, 0, -0.125, 0)  
Proposal too wide

# Bounding-box regression

## RCNN



# Análise de execução

- Alta performance
- A CNN gera poucas características:
  - “*4096-dimensional features*”
- Outros times (*UvA*) lidaram com vetores de ordem  $4.000 \times 360.000$

# Testes de Ross Girshick (autor)

- 13s/image on a GPU NVIDIA Titan Black
- 53s/image on a CPU
- 1.5GB de memória
- Abordagens como as do UvA demandariam cerca de 134GB de memória

# Resultados

VOC 2010 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
DPM v5 [20] <sup>†</sup>	49.2	53.8	13.1	15.3	35.5	53.4	49.7	27.0	17.2	28.8	14.7	17.8	46.4	51.2	47.7	10.8	34.2	20.7	43.8	38.3	33.4
UVA [39]	56.2	42.4	15.3	12.6	21.8	49.3	36.8	46.1	12.9	32.1	30.0	36.5	43.5	52.9	32.9	15.3	41.1	31.8	47.0	44.8	35.1
Regionlets [41]	65.0	48.9	25.9	24.6	24.5	56.1	54.5	51.2	17.0	28.9	30.2	35.8	40.2	55.7	43.5	14.3	43.9	32.6	54.0	45.9	39.7
SegDPM [18] <sup>†</sup>	61.4	53.4	25.6	25.2	35.5	51.7	50.6	50.8	19.3	33.8	26.8	40.4	48.3	54.4	47.1	14.8	38.7	35.0	52.8	43.1	40.4
R-CNN	67.1	64.1	46.7	32.0	30.5	56.4	57.2	65.9	27.0	47.3	40.9	66.6	57.8	65.9	53.6	26.7	56.5	38.1	52.8	50.2	50.2
R-CNN BB	<b>71.8</b>	<b>65.8</b>	<b>53.0</b>	<b>36.8</b>	<b>35.9</b>	<b>59.7</b>	<b>60.0</b>	<b>69.9</b>	<b>27.9</b>	<b>50.6</b>	<b>41.4</b>	<b>70.0</b>	<b>62.0</b>	<b>69.0</b>	<b>58.1</b>	<b>29.5</b>	<b>59.4</b>	<b>39.3</b>	<b>61.2</b>	<b>52.4</b>	<b>53.7</b>

**Table 1: Detection average precision (%) on VOC 2010 test.** R-CNN is most directly comparable to UVA and Regionlets since all methods use selective search region proposals. Bounding-box regression (BB) is described in Section C. At publication time, SegDPM was the top-performer on the PASCAL VOC leaderboard. <sup>†</sup>DPM and SegDPM use context rescoring not used by the other methods.

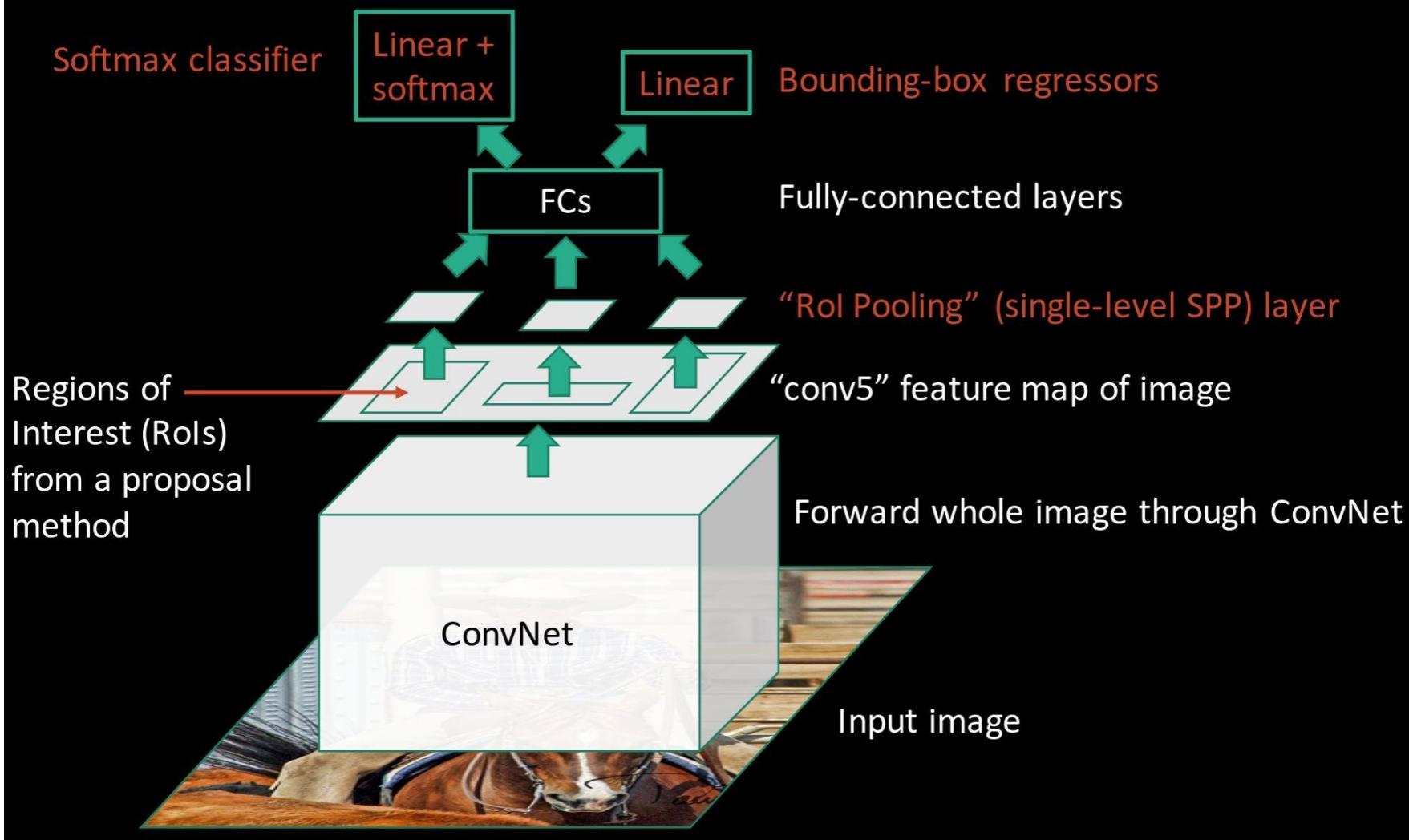
# Problemas CNN

- Lento! Necessita executar uma CNN para cada região
- SVM e Regressor são executados posteriormente. Características não são atualizadas nas respostas do SVM e Regressor

# FAST R-CNN

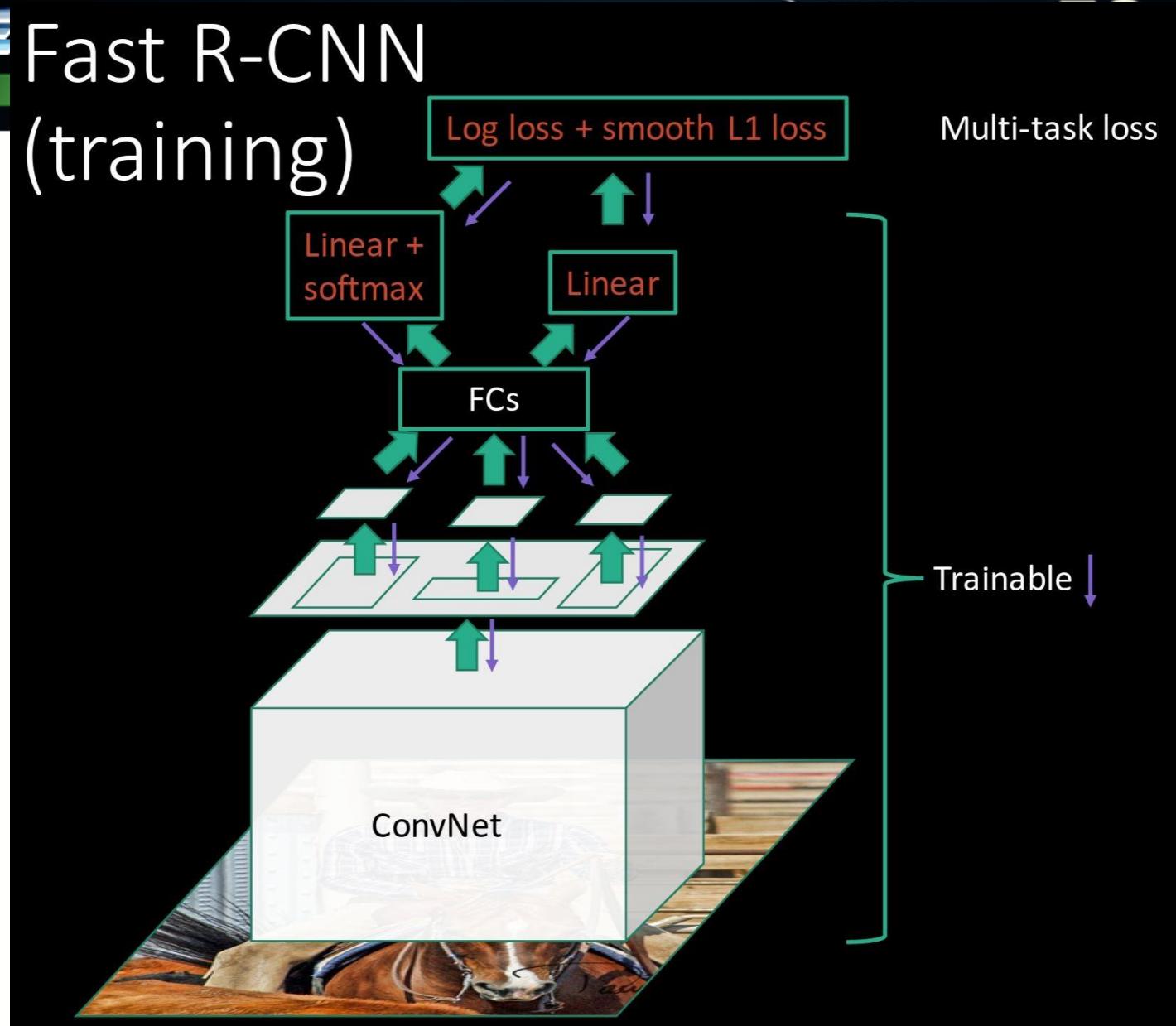
- Lento! Necessita executar uma CNN para cada região
  - Solução: Compartilhar o processamento da camada convolucional com todas as regiões.
- SVM e Regressor são executados posteriormente. Características não são atualizadas nas respostas do SVM e Regressor
  - No treinamento, o resultado é reutilizado na proposição de regiões para uma nova execução (backpropagation).

# Fast R-CNN (test time)



Girschick, "Fast R-CNN", ICCV 2015

# Fast R-CNN (training)



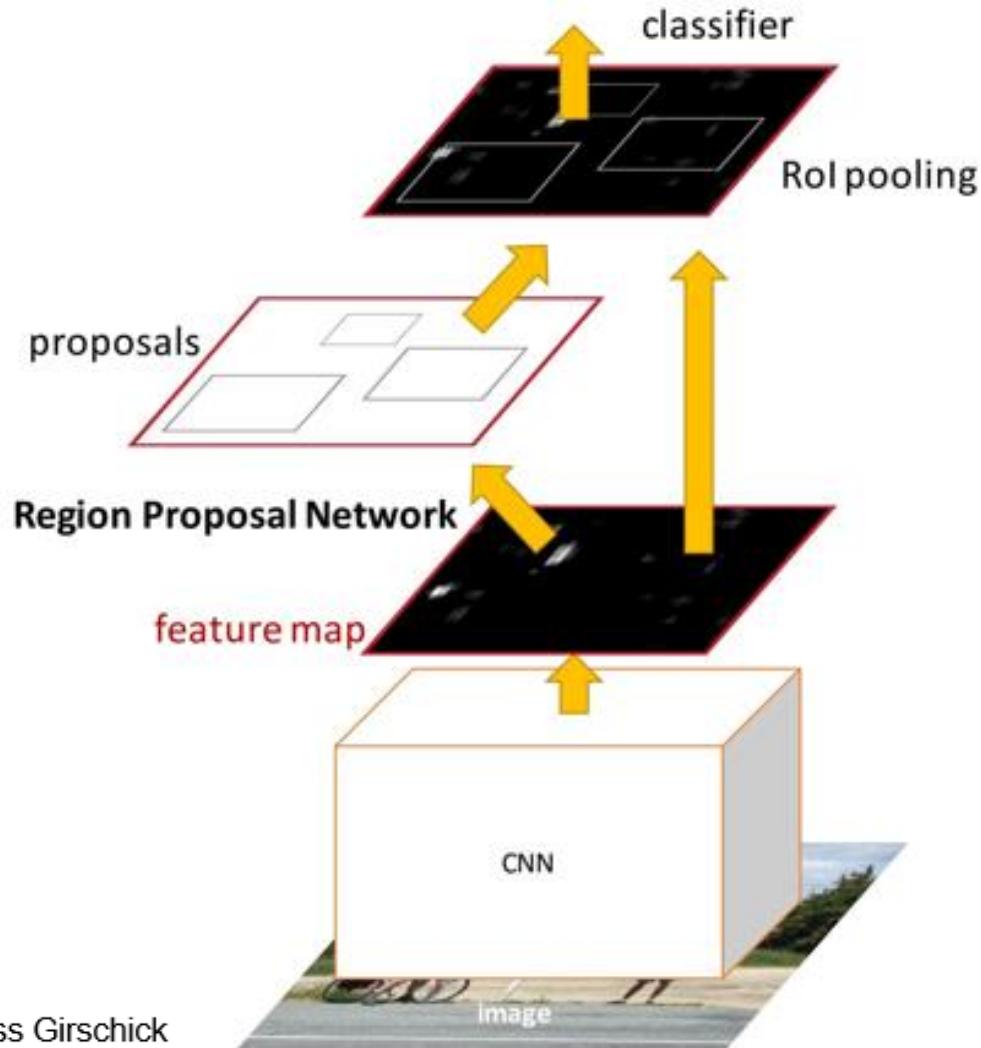
Girschick, "Fast R-CNN", ICCV 2015

# Faster R-CNN

- Deixa de usar o Selective Search para pré-selecionar regiões.
- Selective Search foi considerado lento: 2s por imagem (CPU)
- No lugar, Inserida a camada Region Proposal Network (RPN) após a última camada de convolução

# Faster R-CNN

- RPN treinada produz as propostas de regiões. Não é mais necessária camada de proposição de regiões externa.
- Após proposição de regiões utiliza regressor BBOX como o Fast R-CNN

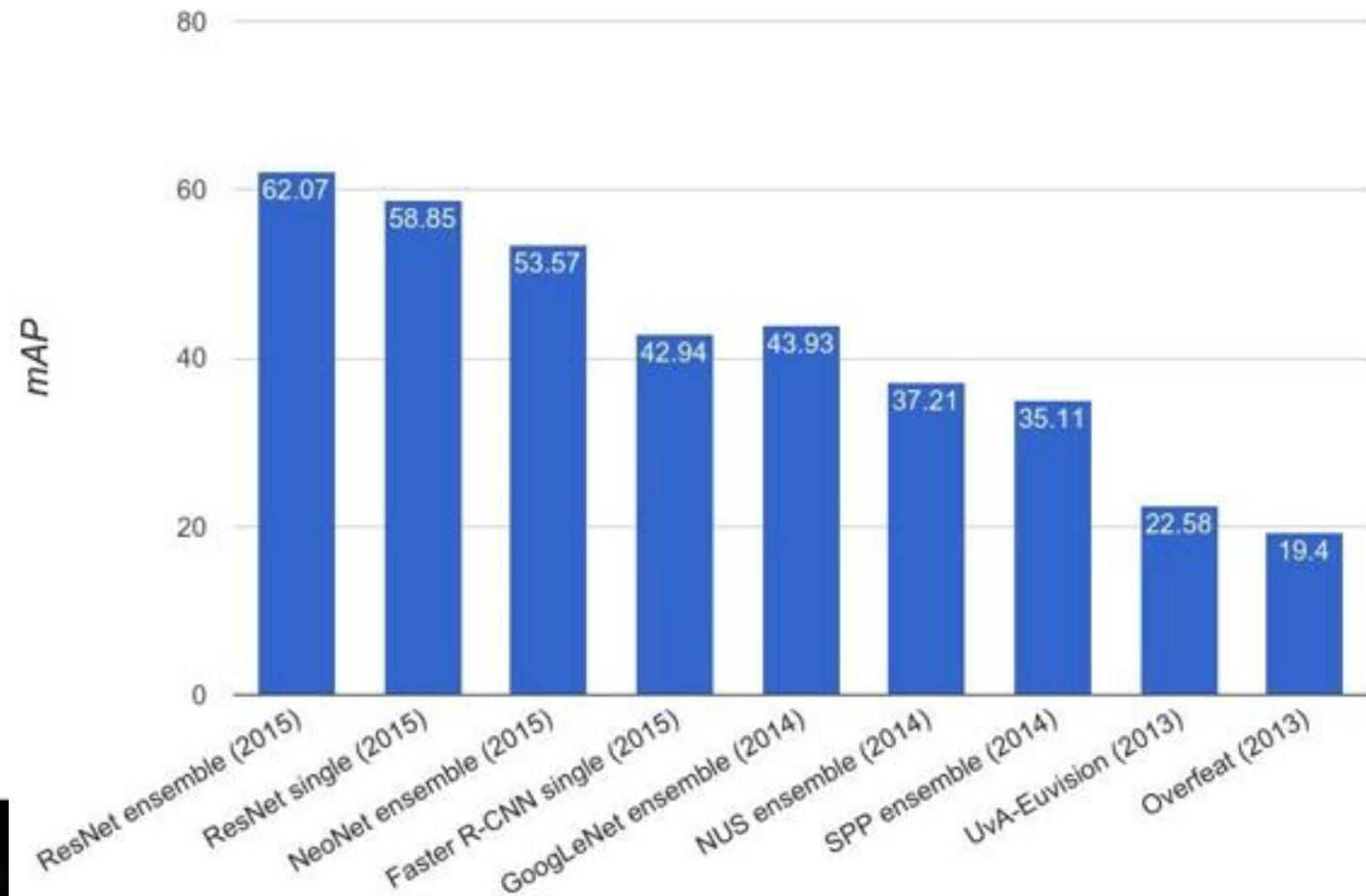


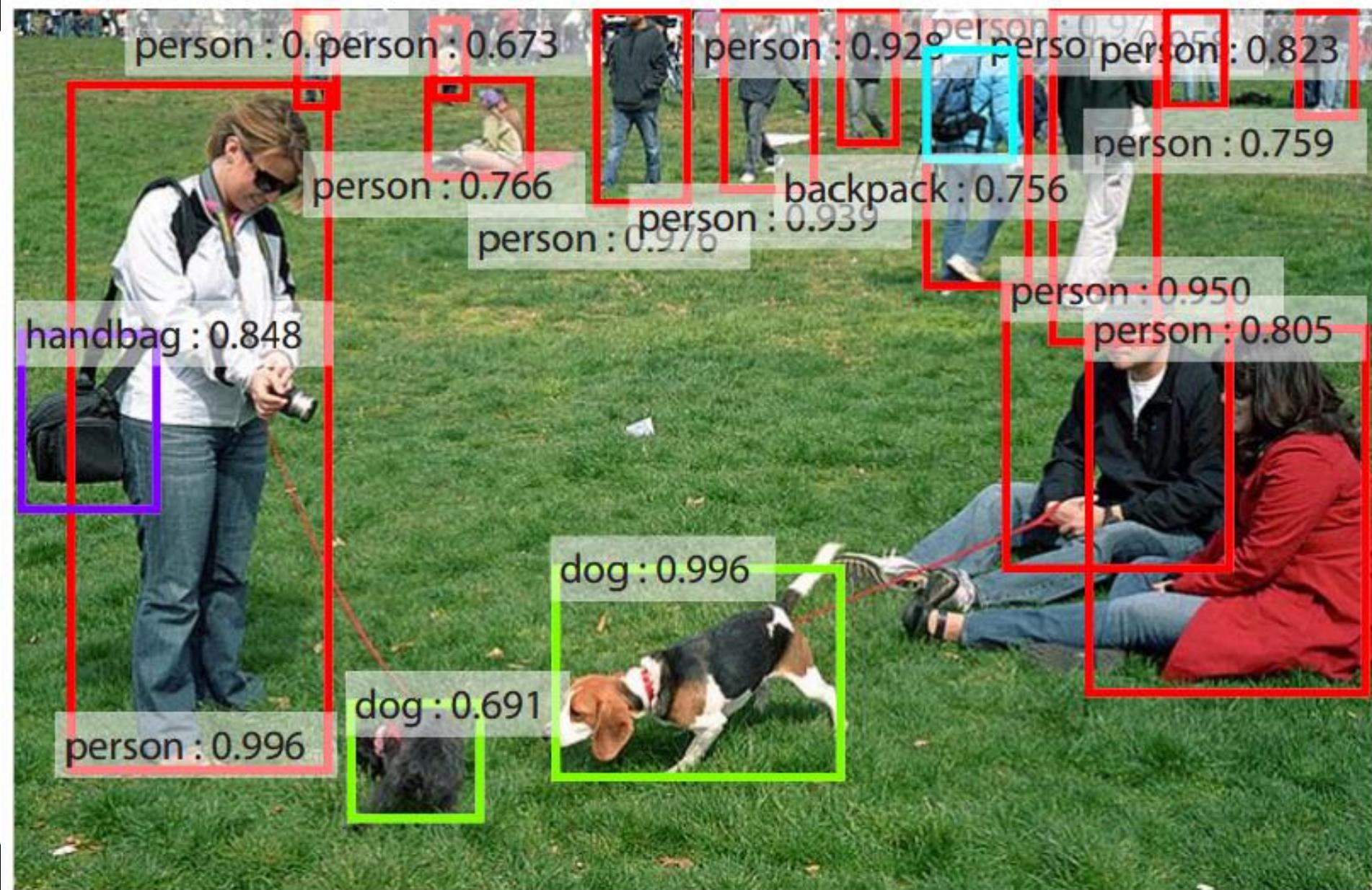
# Resultados

	R-CNN	Fast R-CNN	Faster R-CNN
Test time per image (with proposals)	50 seconds	2 seconds	<b>0.2 seconds</b>
(Speedup)	1x	25x	<b>250x</b>
mAP (VOC 2007)	66.0	<b>66.9</b>	<b>66.9</b>

# ImageNet Detection 2013 - 2015

## ImageNet Detection (mAP)





airplane : 0.997

person : 0.867



clock: 0.986

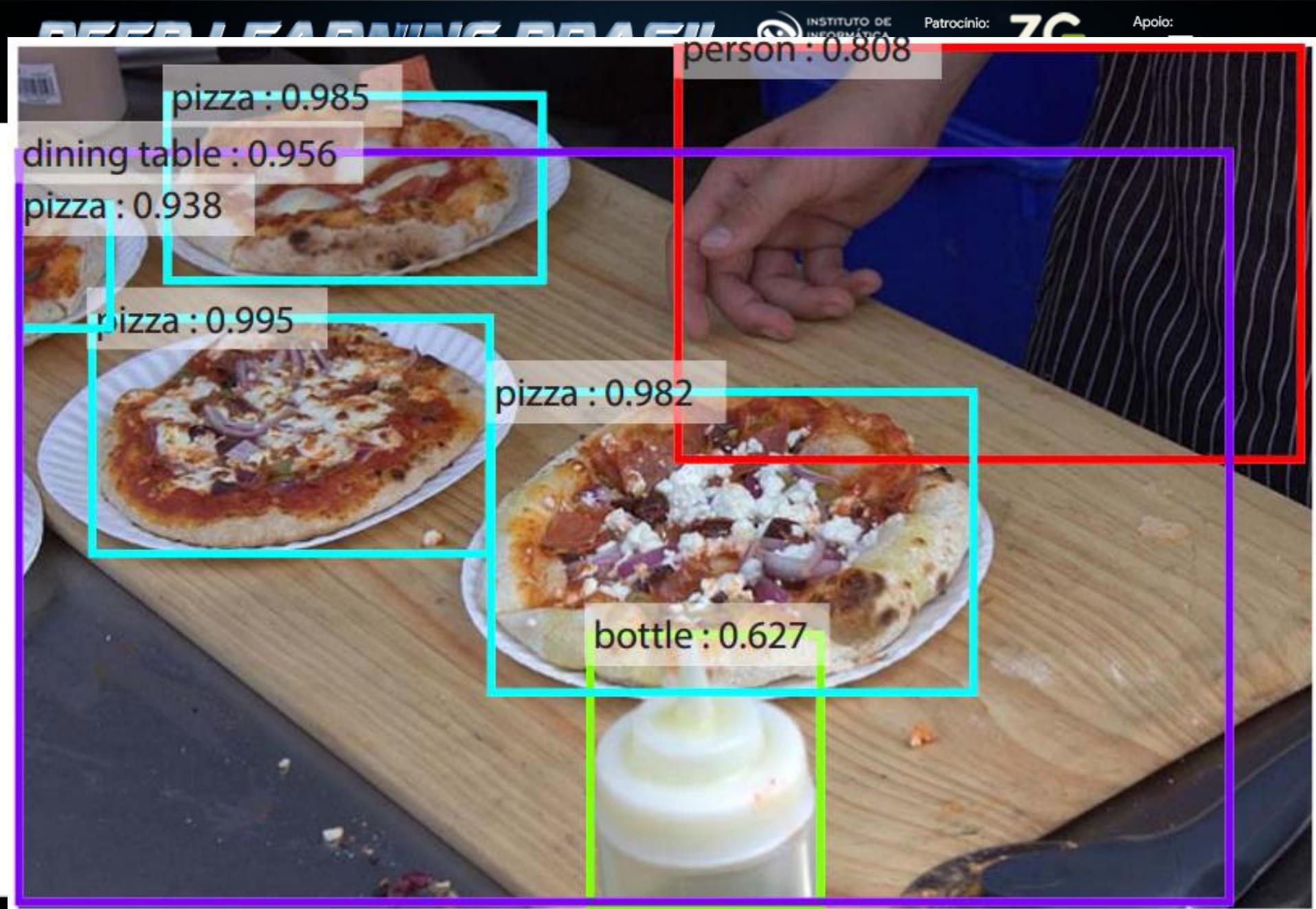


clock : 0.981

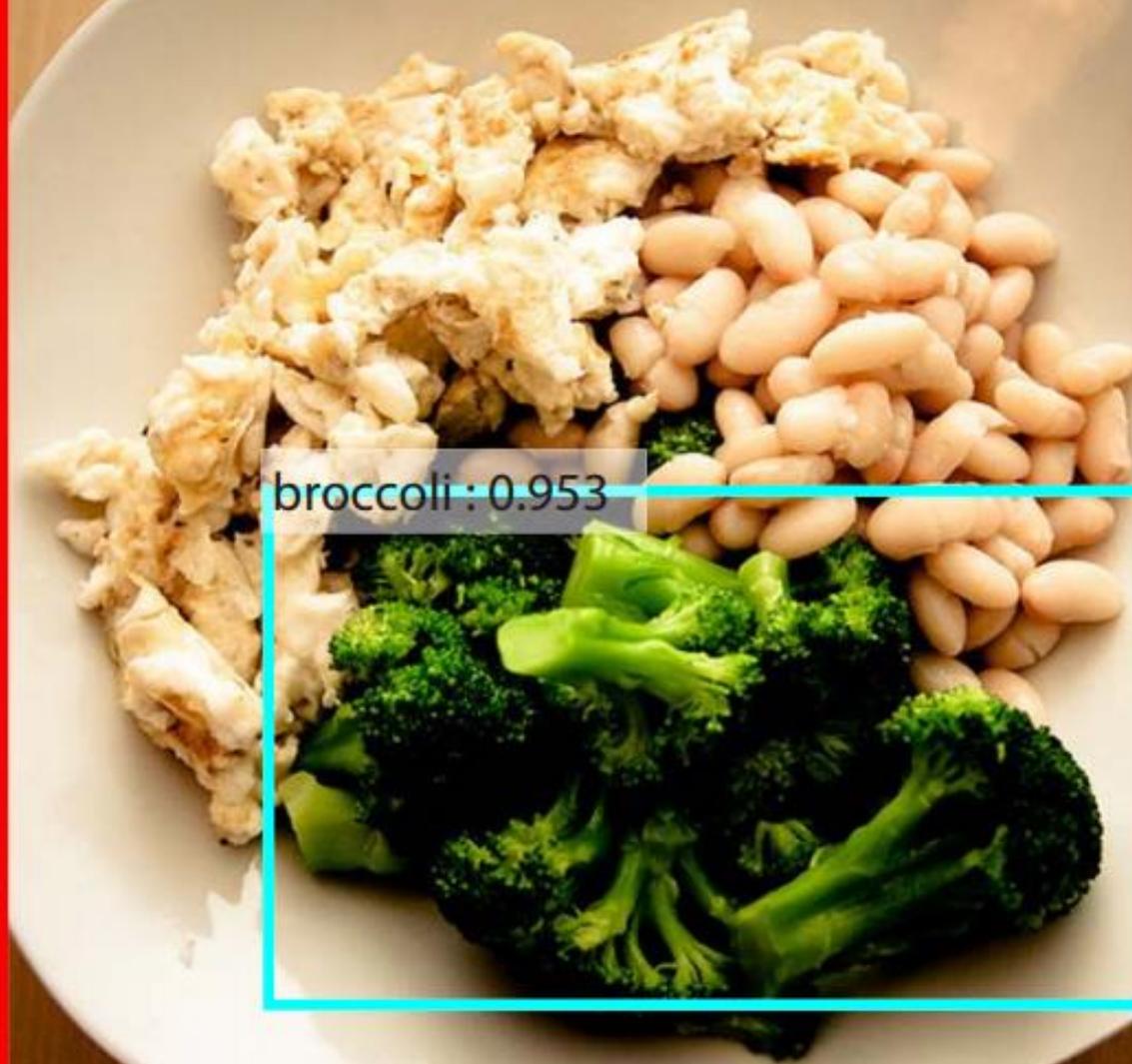


person : 0.800





bowl : 0.759



# Conclusão

- RPN diminuiu significativamente o tempo de detecção e proposição de regiões implicando em melhorias na performance.
- Tempo real
- Sem perda da capacidade de reconhecimento

# Obrigado