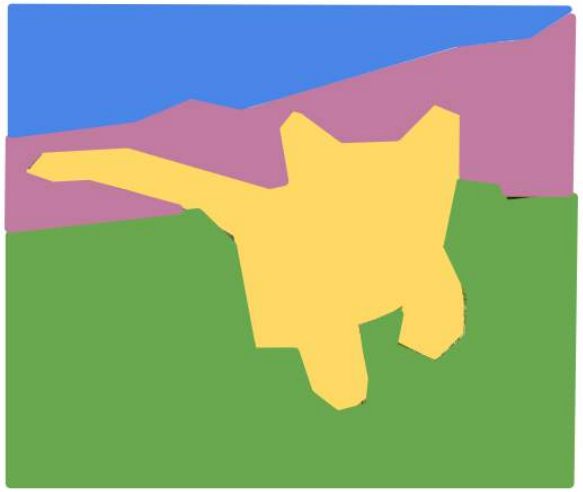
Detection and Segmentation

Core CV Tasks

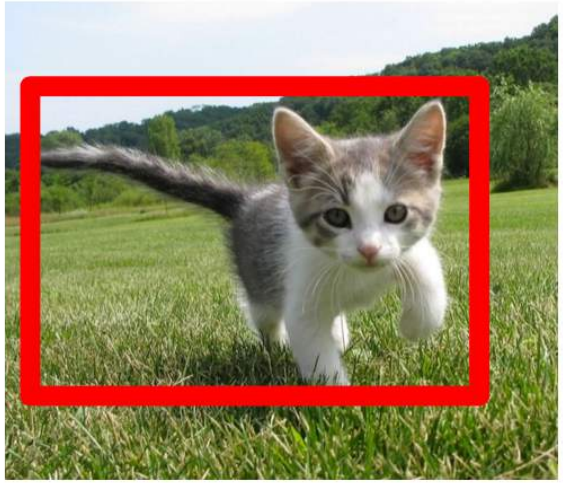
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

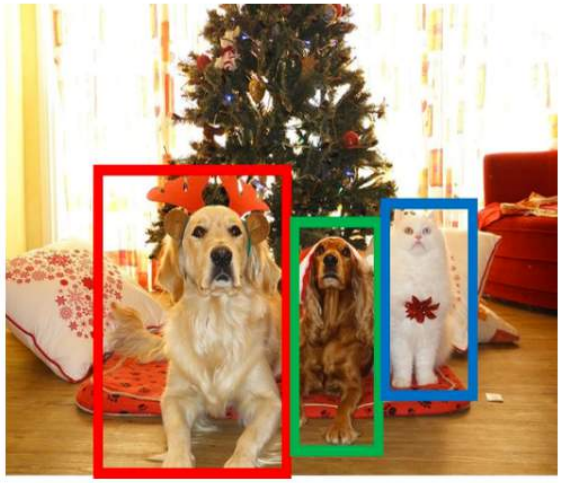
Classification + Localization



CAT

Single Object

Object Detection



DOG, DOG, CAT

Instance Segmentation

DOG, DOG, CAT

Multiple Object

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

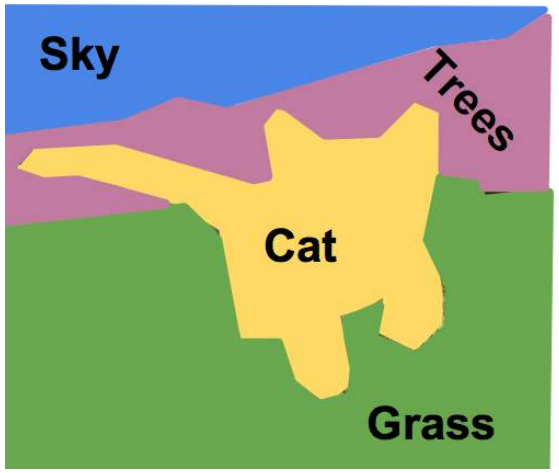
Semantic Segmentation

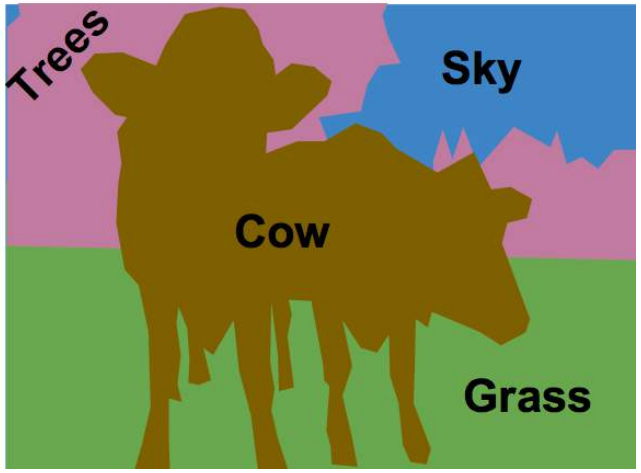
Задача:

Дать лейбл каждому пикселю на картинке.

Важная деталь:

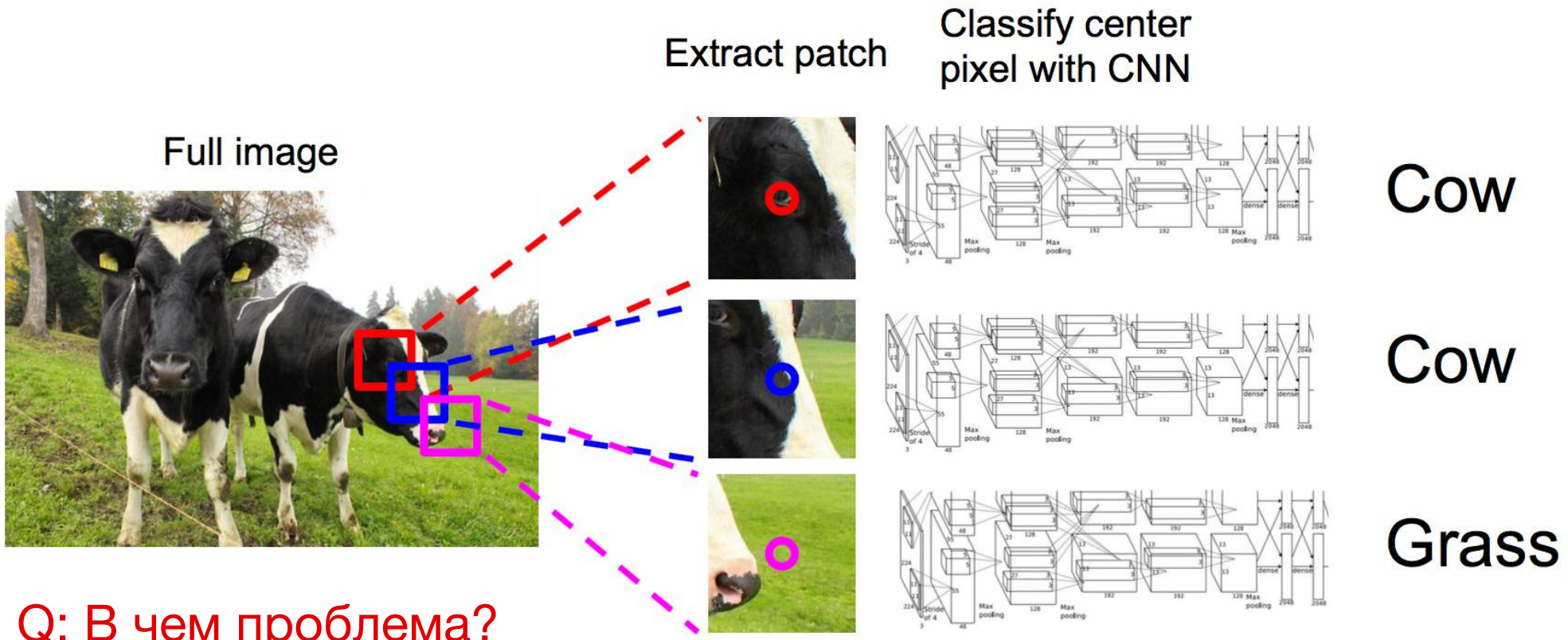
Нас волнуют только пиксели, а не конкретные объекты.





Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Semantic Segmentation: Sliding Window



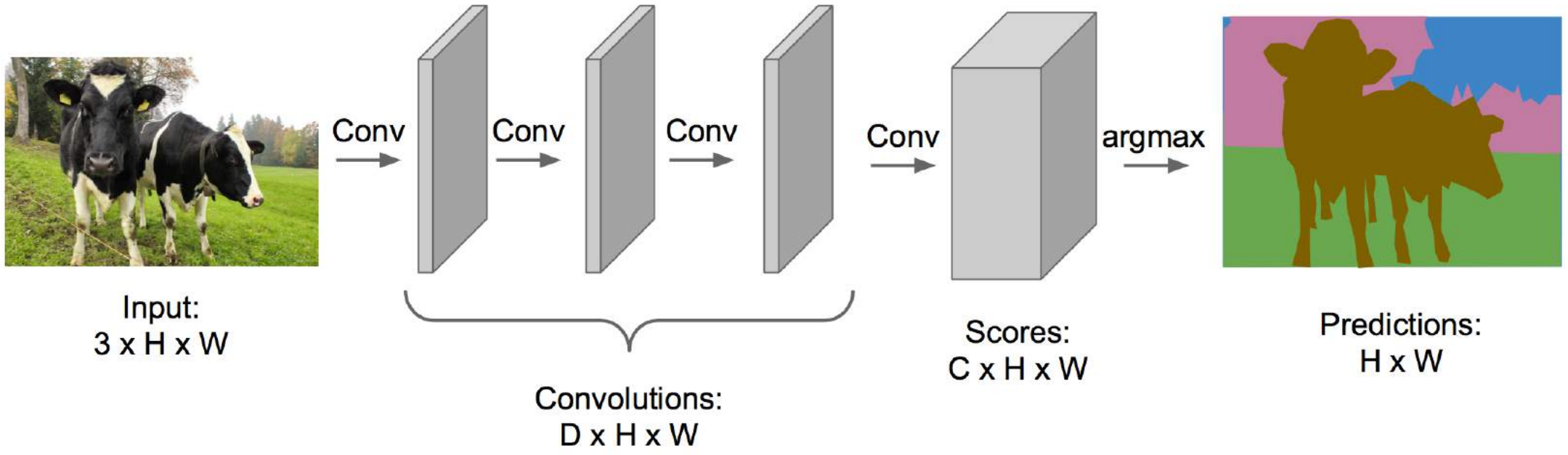
Q: В чем проблема?

Очень дорого считать.

Farabet et al, “Learning Hierarchical Features for Scene Labeling,” TPAMI 2013
Pinheiro and Collobert, “Recurrent Convolutional Neural Networks for Scene Labeling”, ICML 2014

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Semantic Segmentation: Fully Convolutional Net

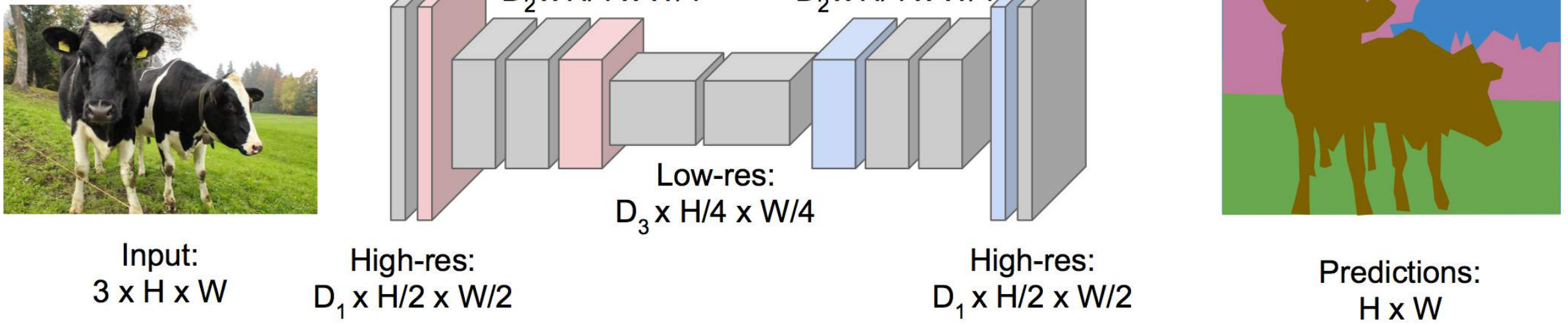


Q: В чем проблема?

Все еще дорого считать + нужно много памяти.

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Semantic Segmentation: Downsampling + Upsampling

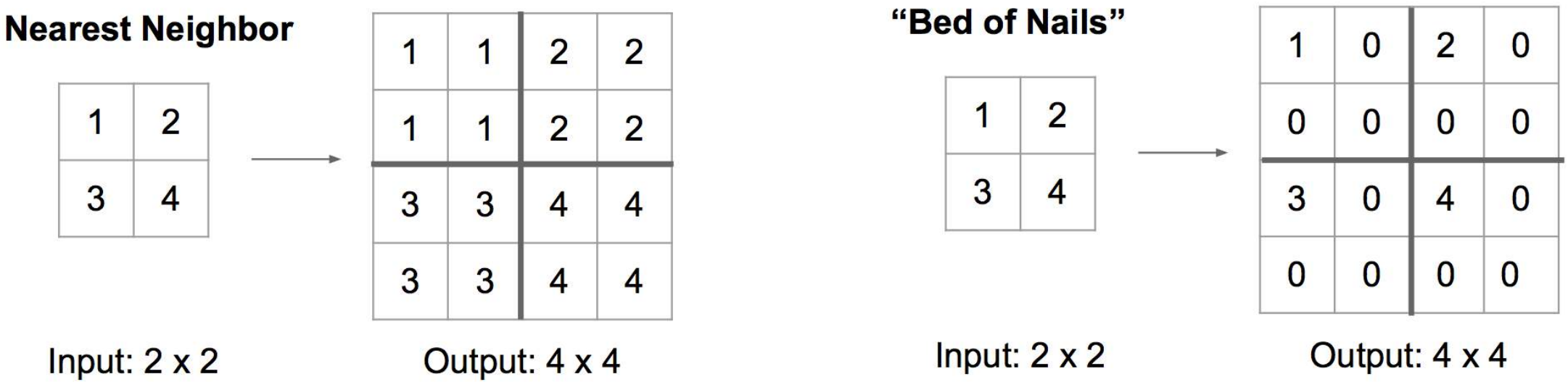


Q: Как делать upsampling?

В основном двумя способами: unpooling и transpose convolution.

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Unpooling

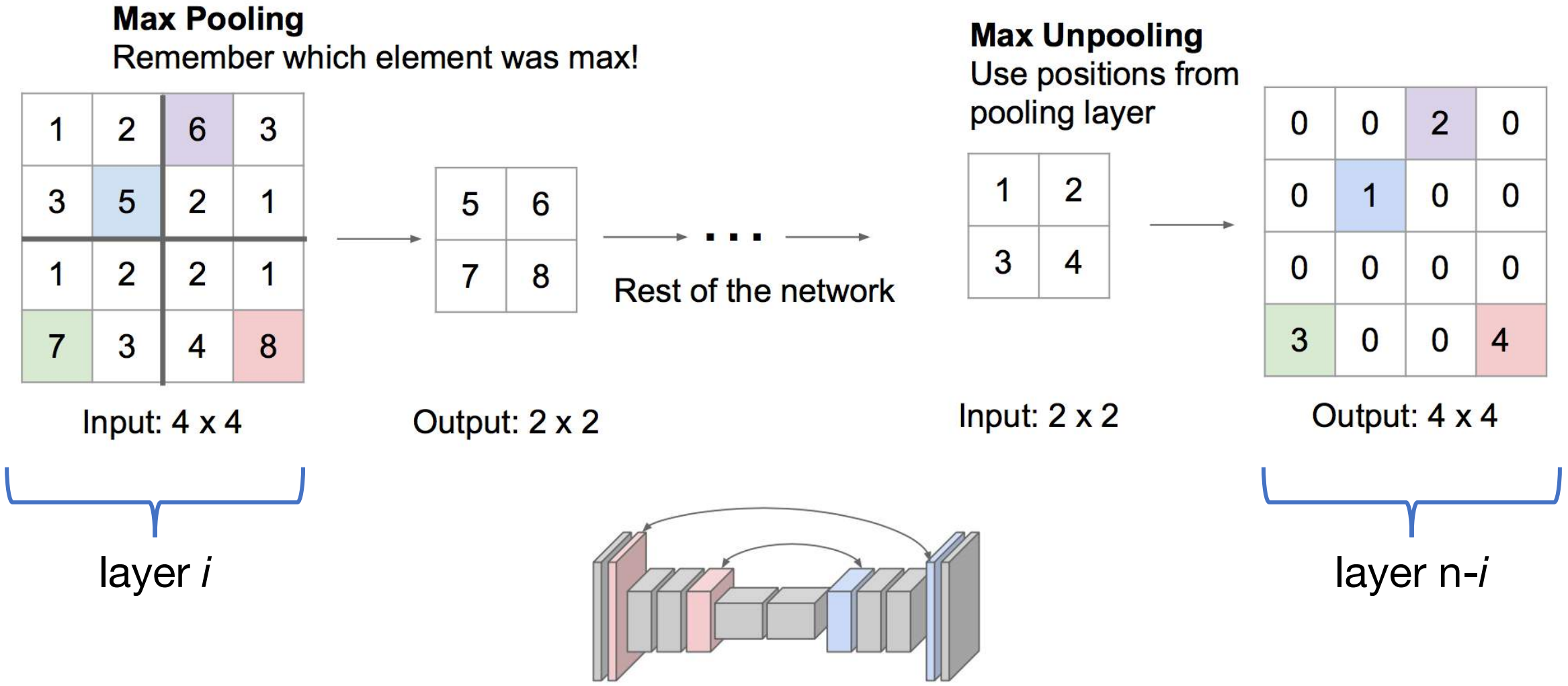


Q: Что плохо?

Это слишком просто, чтобы выучить сложные зависимости.

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Unpooling



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

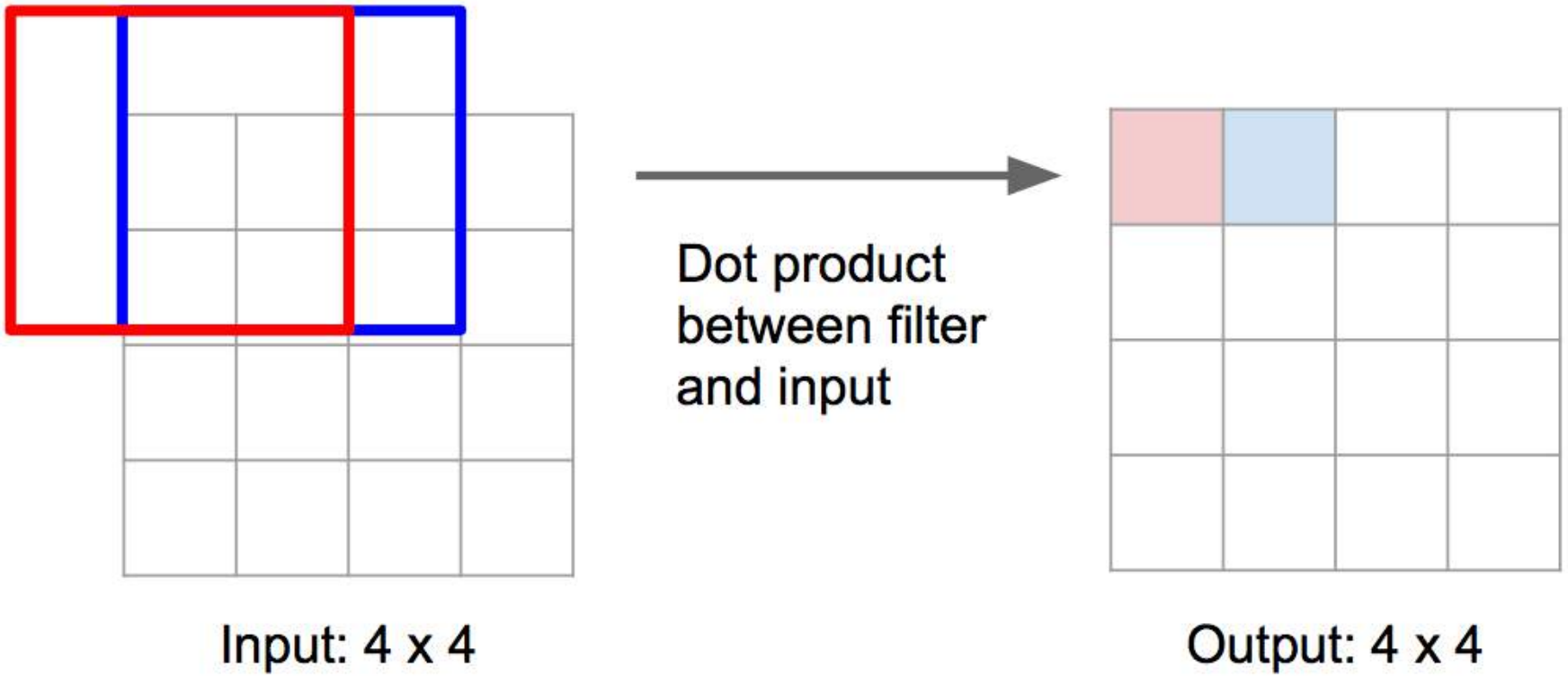
Transpose Convolution

Обычная свертка filter = 3x3, stride = 1, pad = 1

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Transpose Convolution

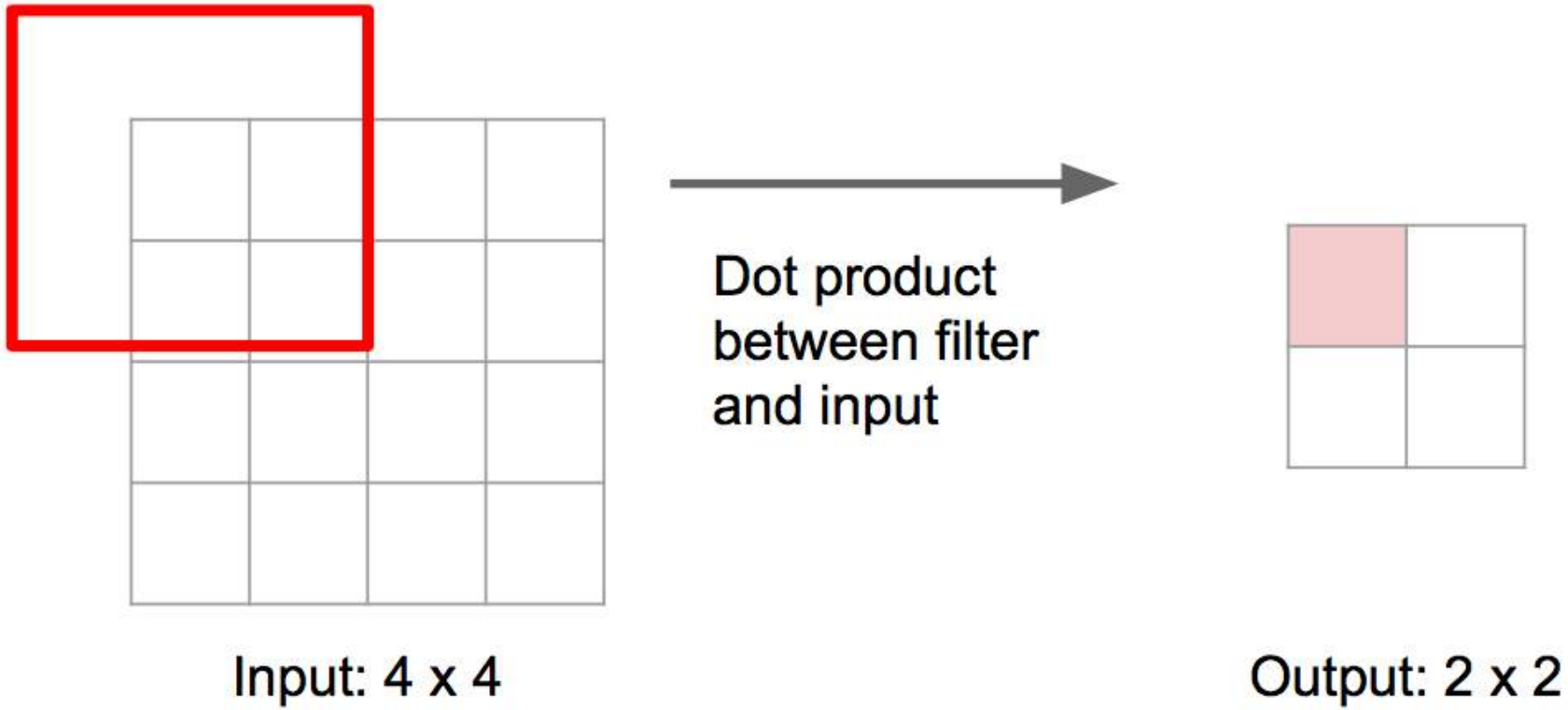
Обычная свертка filter = 3x3, stride = 1, pad = 1



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Transpose Convolution

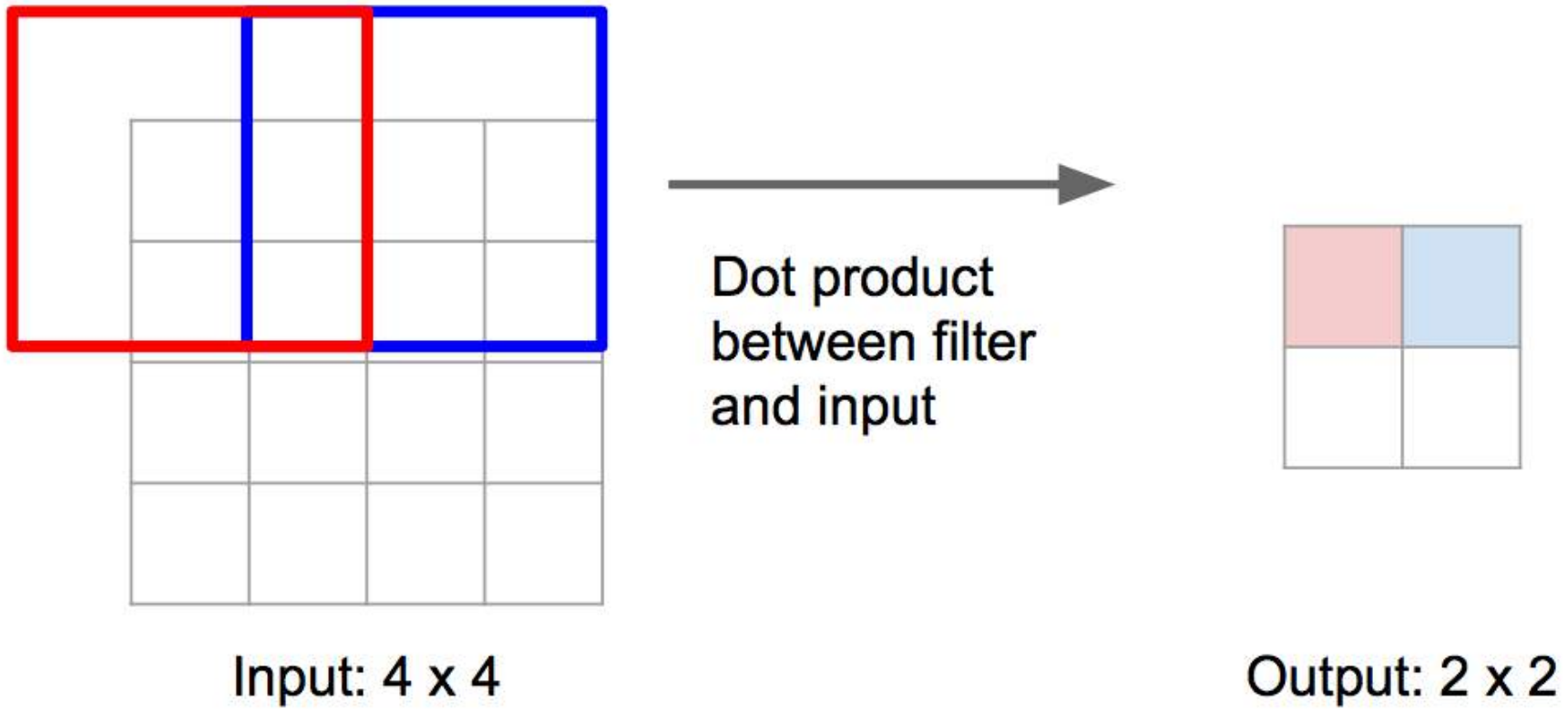
Обычная свертка filter = 3x3, stride = 2, pad = 1



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Transpose Convolution

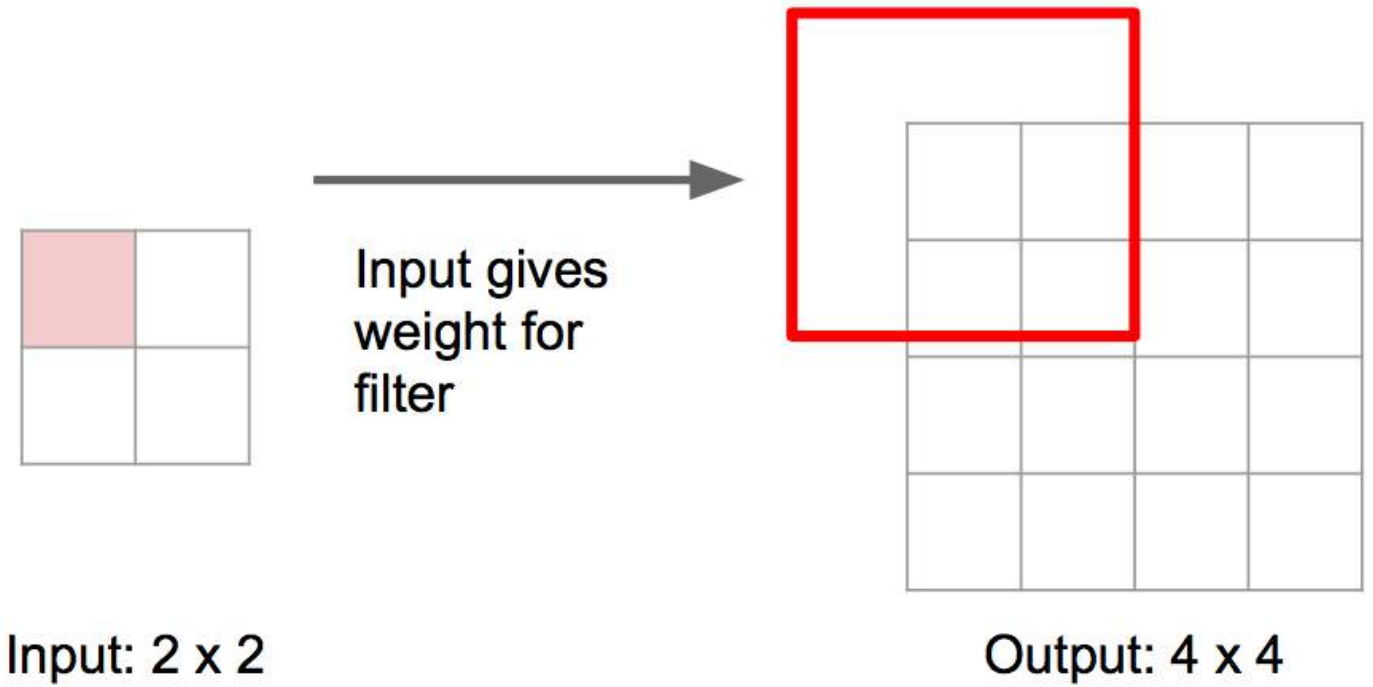
Обычная свертка filter = 3x3, stride = 2, pad = 1



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Transpose Convolution

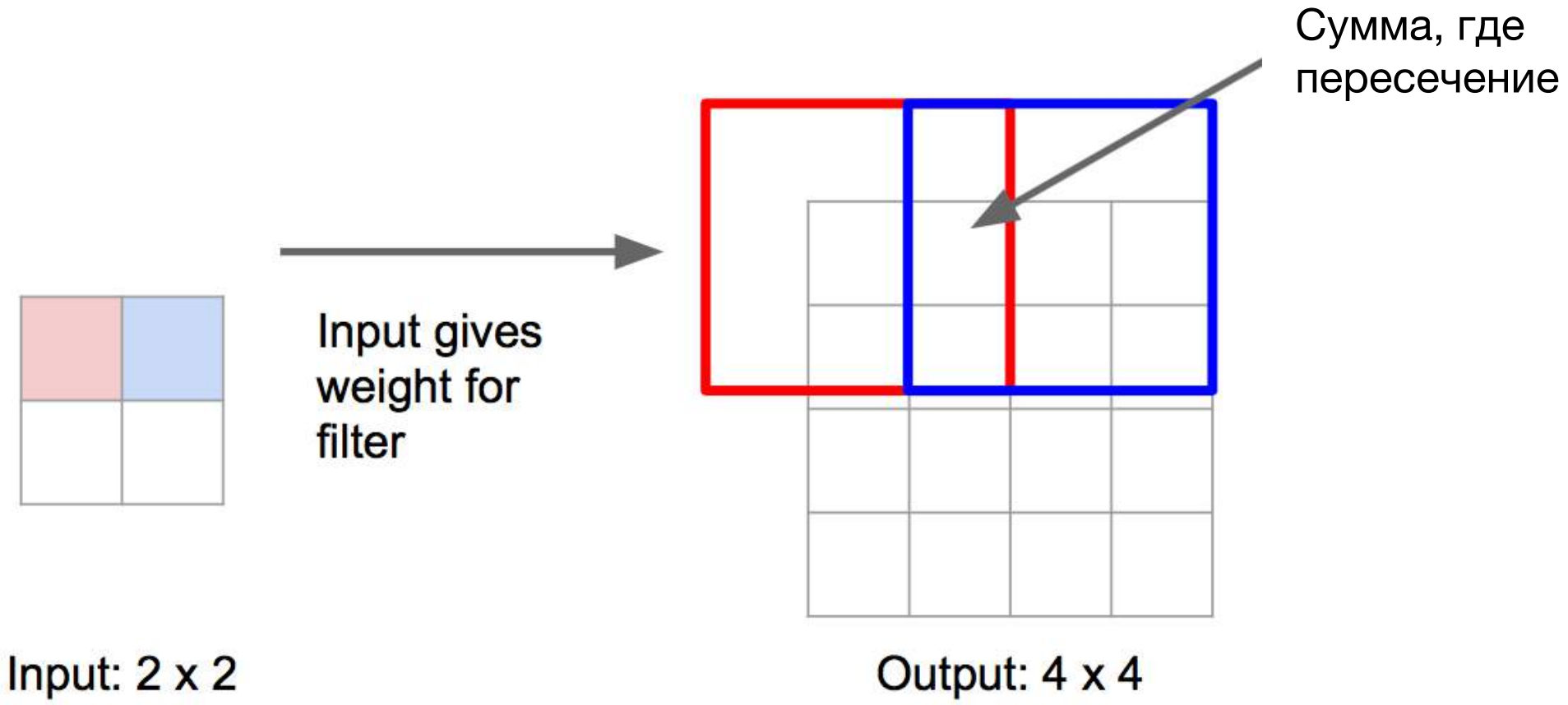
Transpose свертка filter = 3x3, stride = 2, pad = 1



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Transpose Convolution

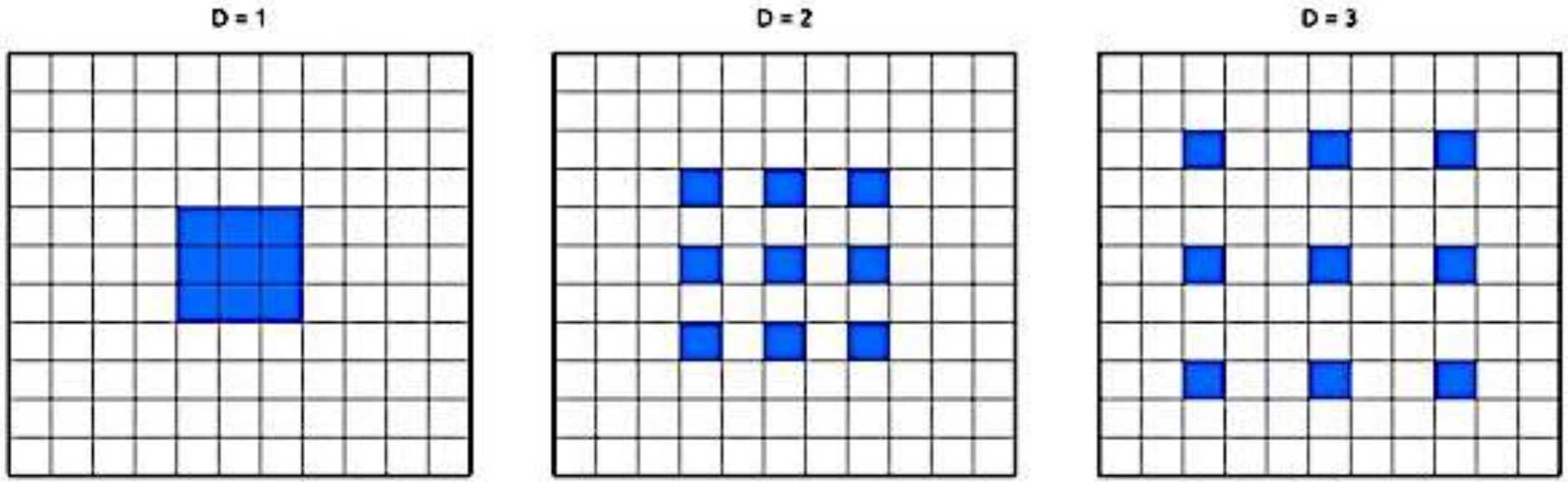
Transpose свертка filter = 3x3, stride = 2, pad = 1



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

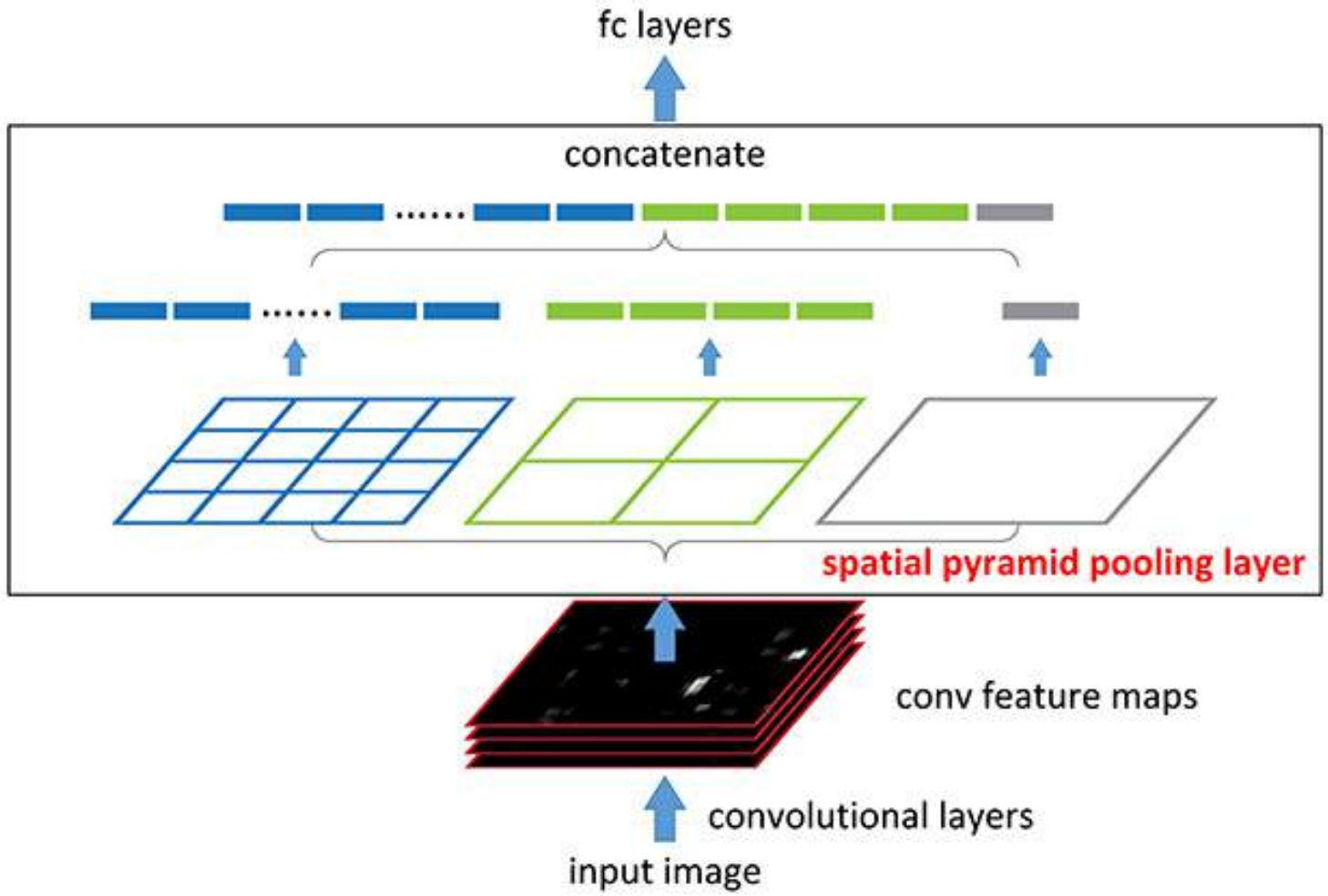
Dilated Convolution

Dilated свертка filter = 3x3



Source: https://www.quora.com/What-is-the-difference-between-dilated-convolution-and-convolution+stride

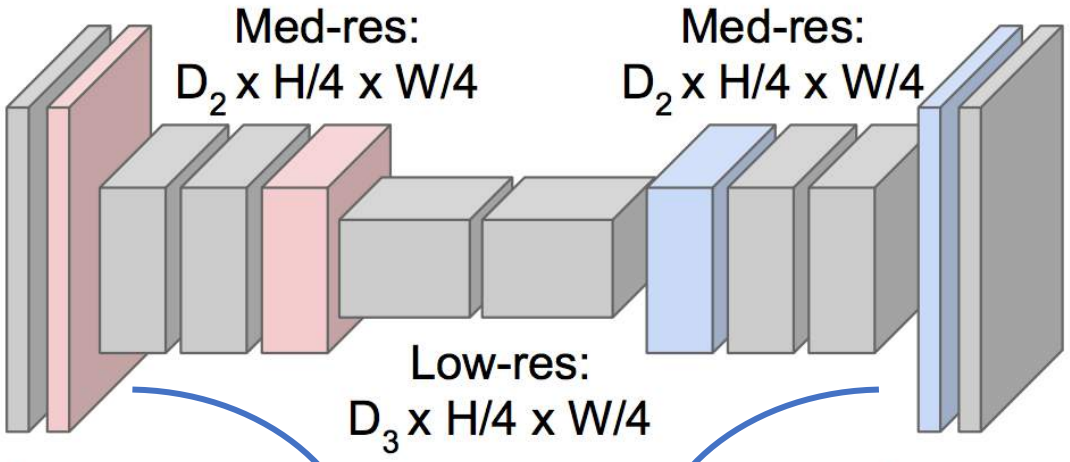
Pyramid Pooling



Source: http://kaiminghe.com/eccv14sppnet/index.html

Semantic Segmentation: General Architecture

Input:
3 x H x W



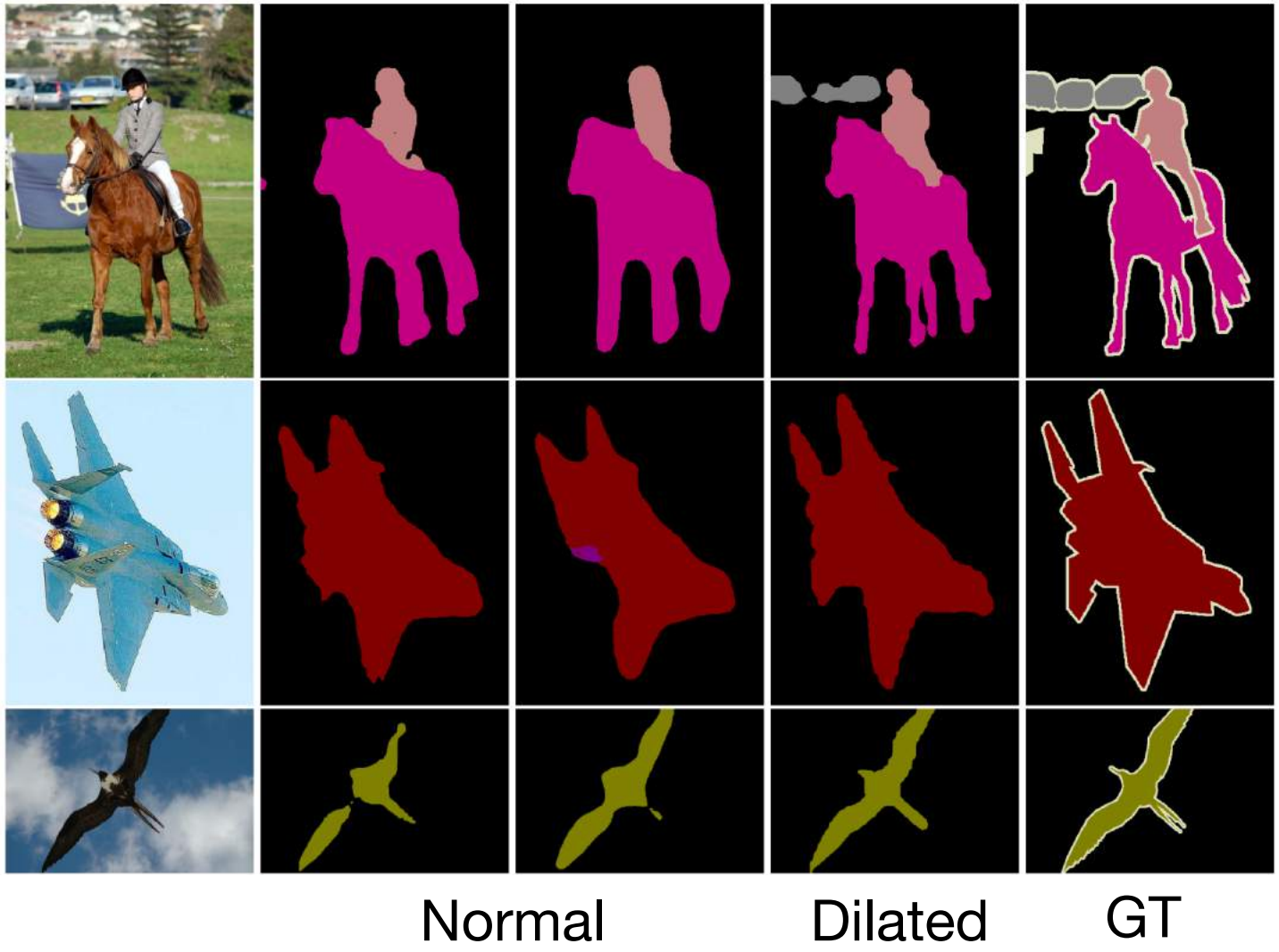
Predictions:
H x W

Pooling, strided conv's

Unpooling, strided transpose conv's

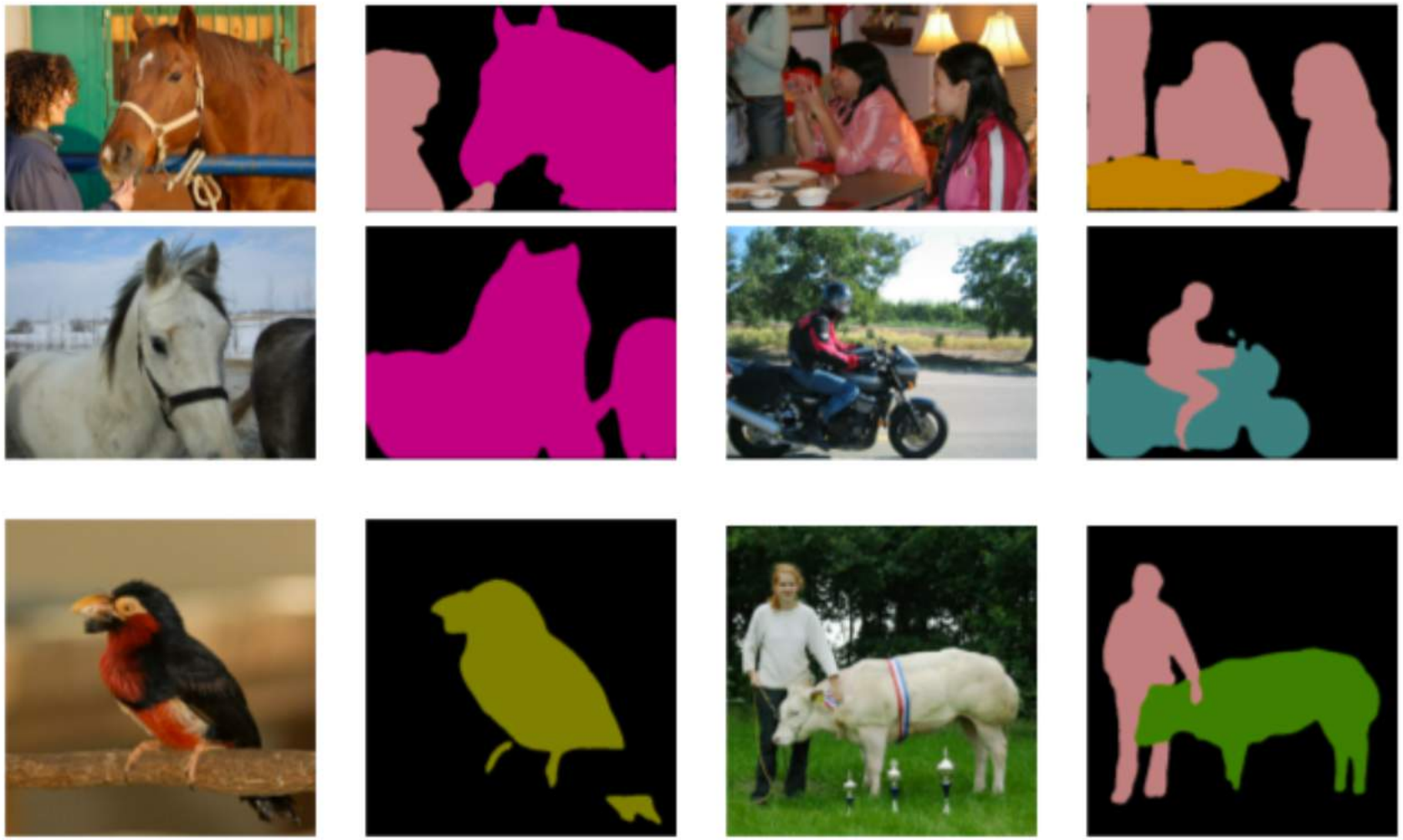
Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Semantic Segmentation: Outcome



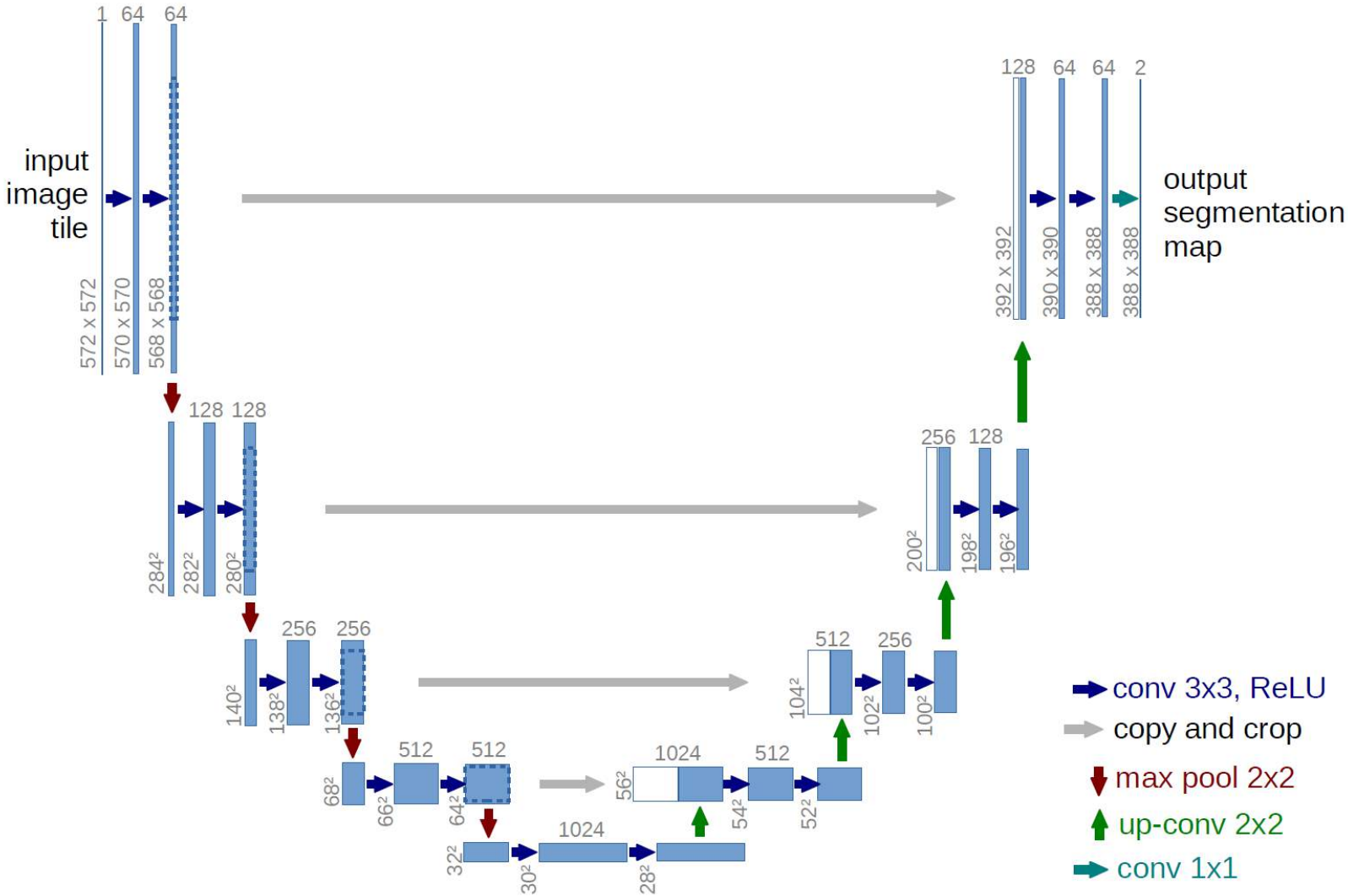
Source: http://vladlen.info/publications/multi-scale-context-aggregation-by-dilated-convolutions/dilated-convolutions/

Semantic Segmentation: Outcome



Source: https://arxiv.org/pdf/1802.02611.pdf

Semantic Segmentation: U-Net (skip-connections)



Source: https://arxiv.org/pdf/1505.04597.pdf

Semantic Segmentation: Basic Ideas

- Downsampling + Upsampling
- Transpose conv’s вместо unpooling
- Dilated свертки вместо обычных
- Skip-connections
- Pyramid pooling

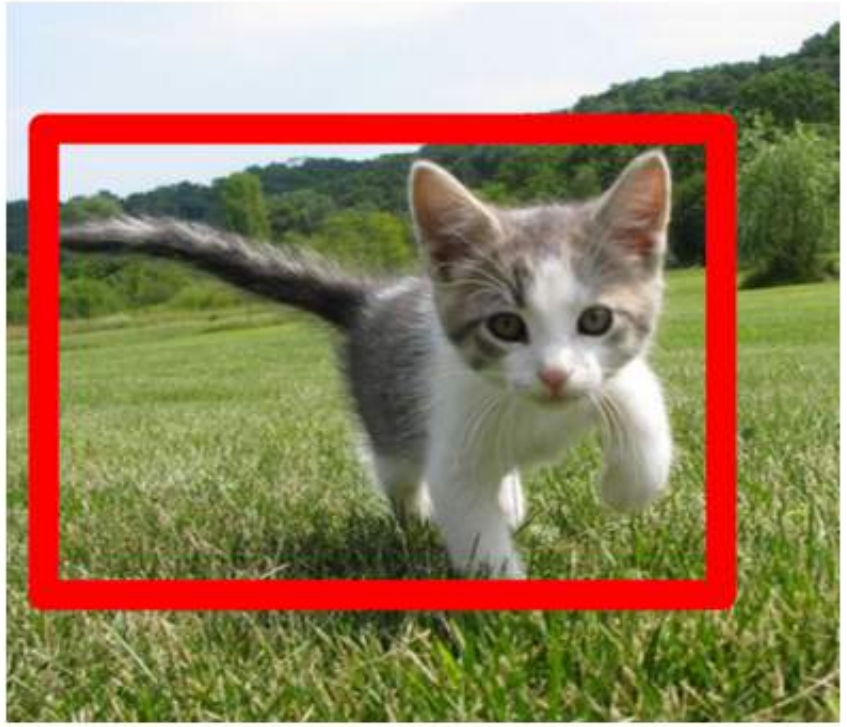
Classification + Localization

Задача:

Обвести объект рамкой и классифицировать его.

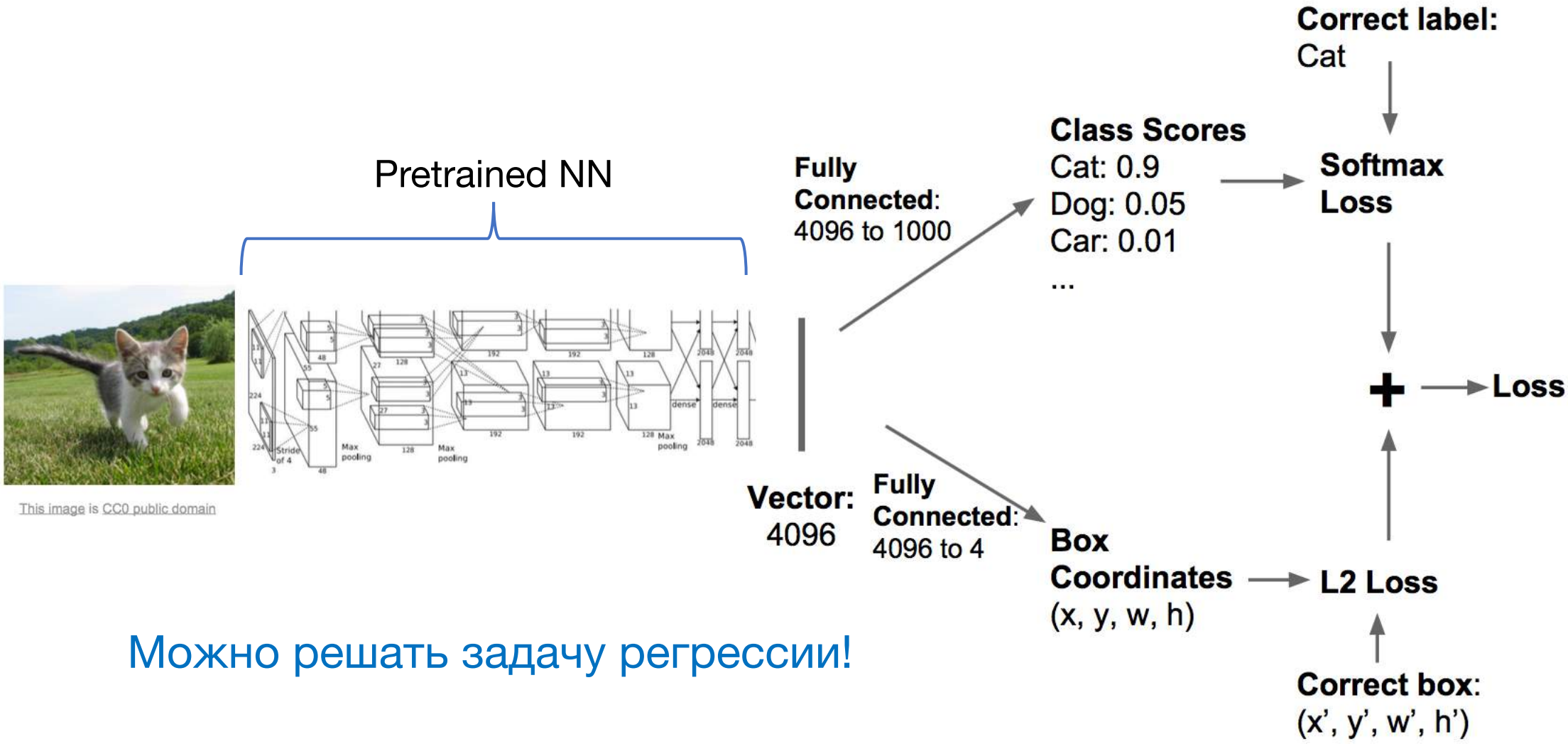
Важная деталь:

Мы всегда имеем дело только с одним объектом на картинке.



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

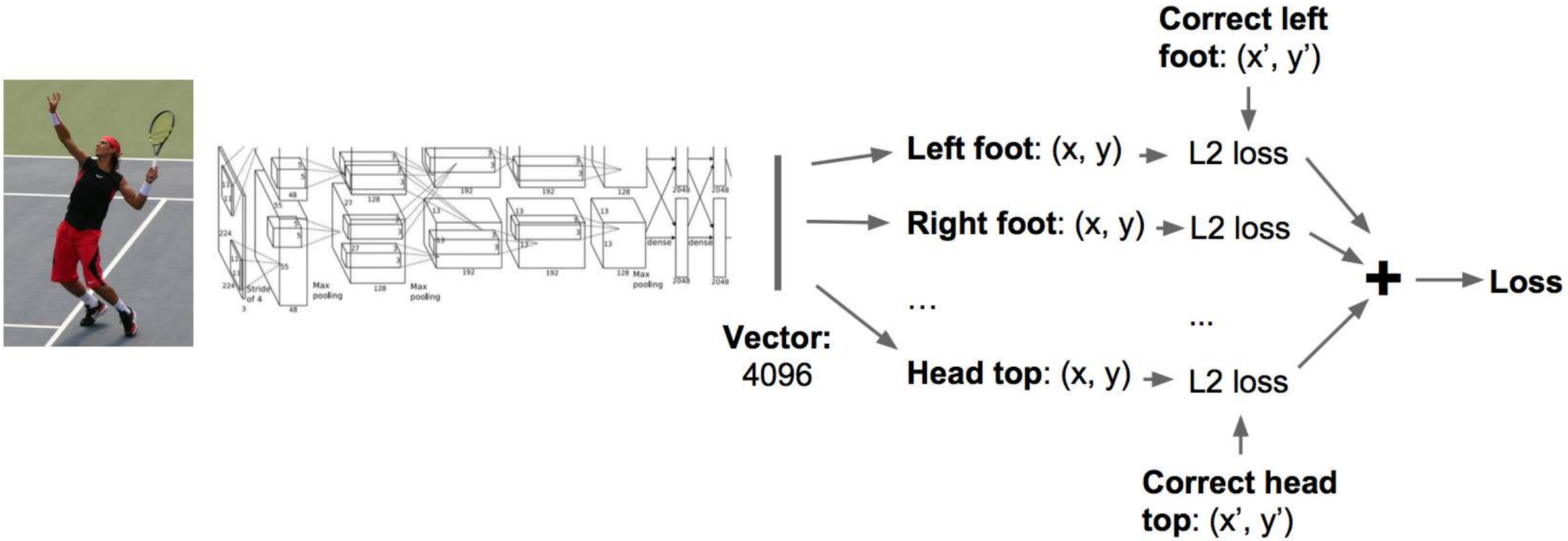
Classification + Localization



Можно решать задачу регрессии!

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Human Pose Estimation



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

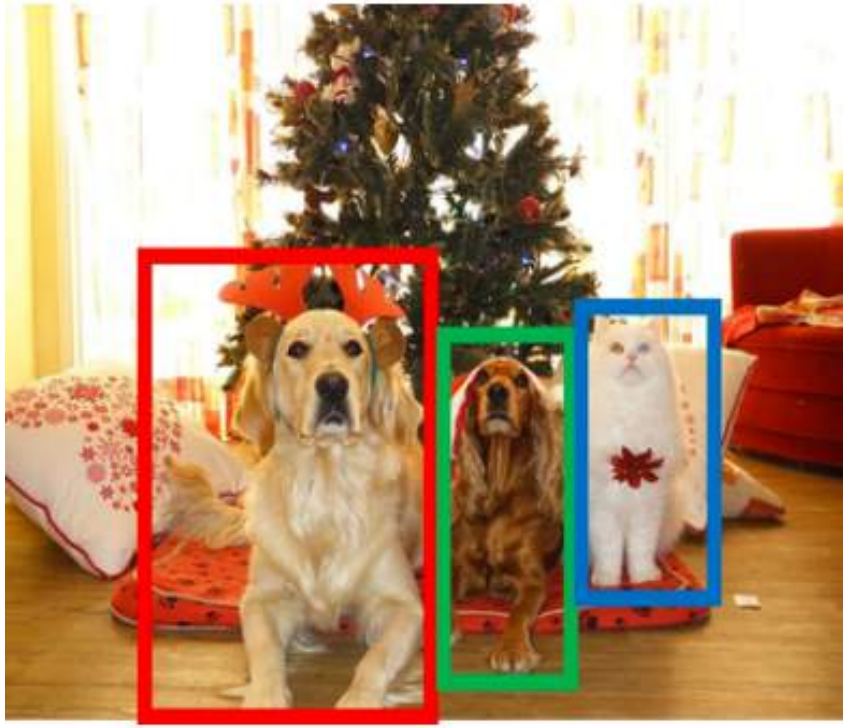
Object Detection

Задача:

Обвести все объекты рамкой
и классифицировать их.

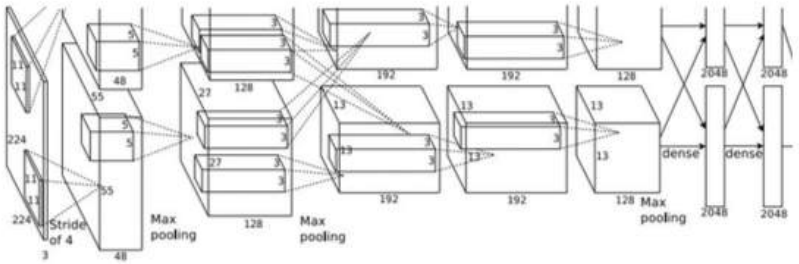
Важная деталь:

Мы имеем дело с
неизвестным количеством
объектов.

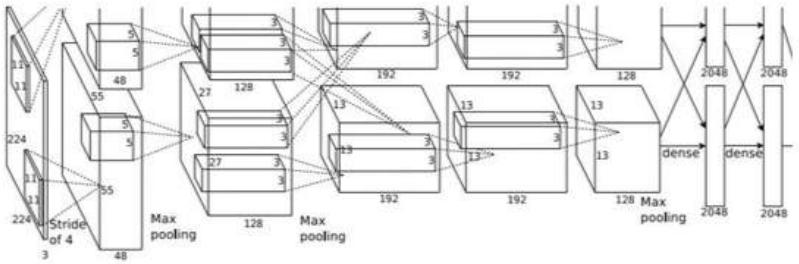


Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

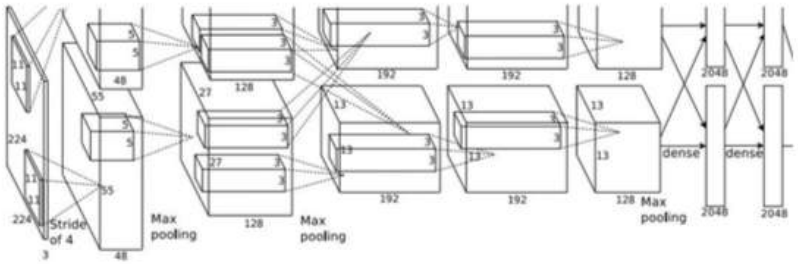
Object Detection



CAT: (x, y, w, h)



DOG: (x, y, w, h)
DOG: (x, y, w, h)
CAT: (x, y, w, h)

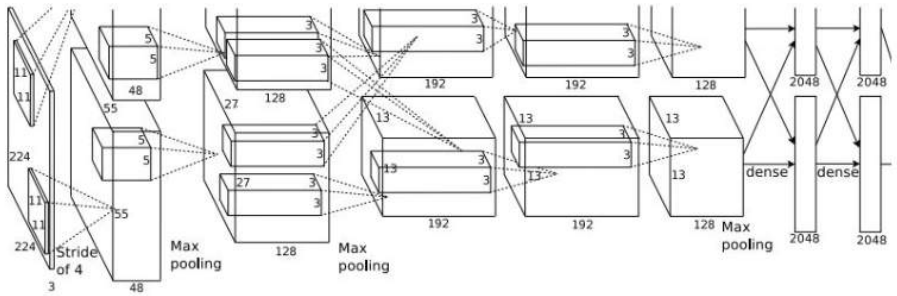


DUCK: (x, y, w, h)
DUCK: (x, y, w, h)
....

Разное количество выходов

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

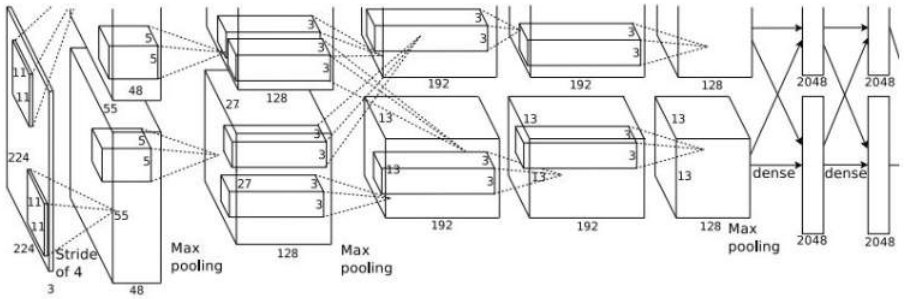
Object Detection: Sliding Window



Dog? NO
Cat? NO
Background? YES

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

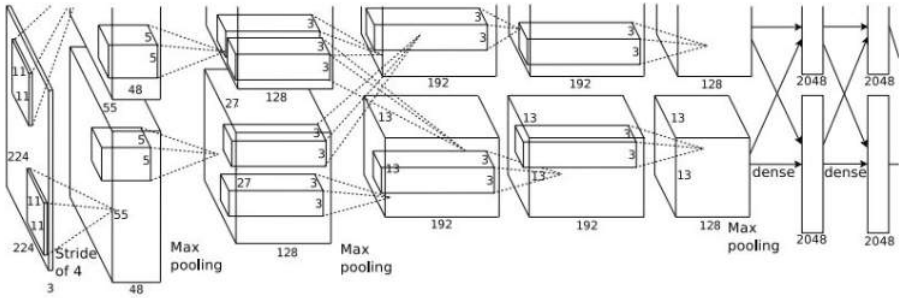
Object Detection: Sliding Window



Dog? YES
Cat? NO
Background? NO

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

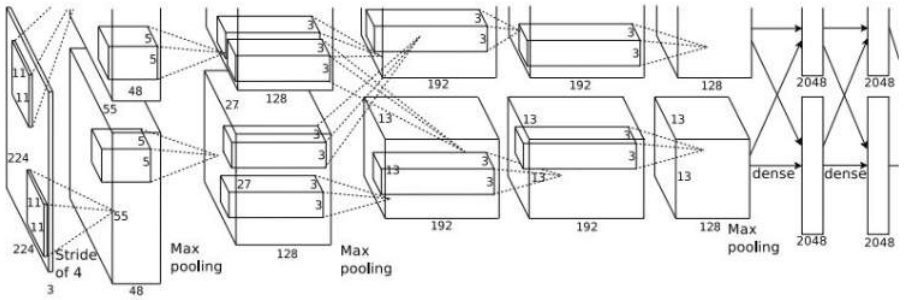
Object Detection: Sliding Window



Dog? YES
Cat? NO
Background? NO

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Object Detection: Sliding Window



Dog? NO
Cat? YES
Background? NO

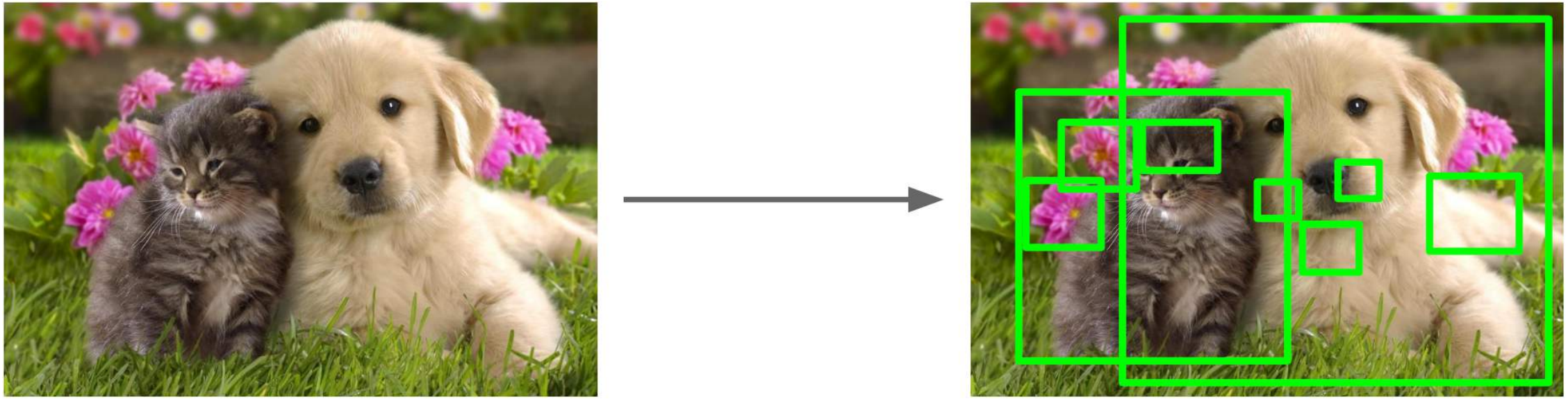
Q: В чем проблема?

Очень дорого считать.

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

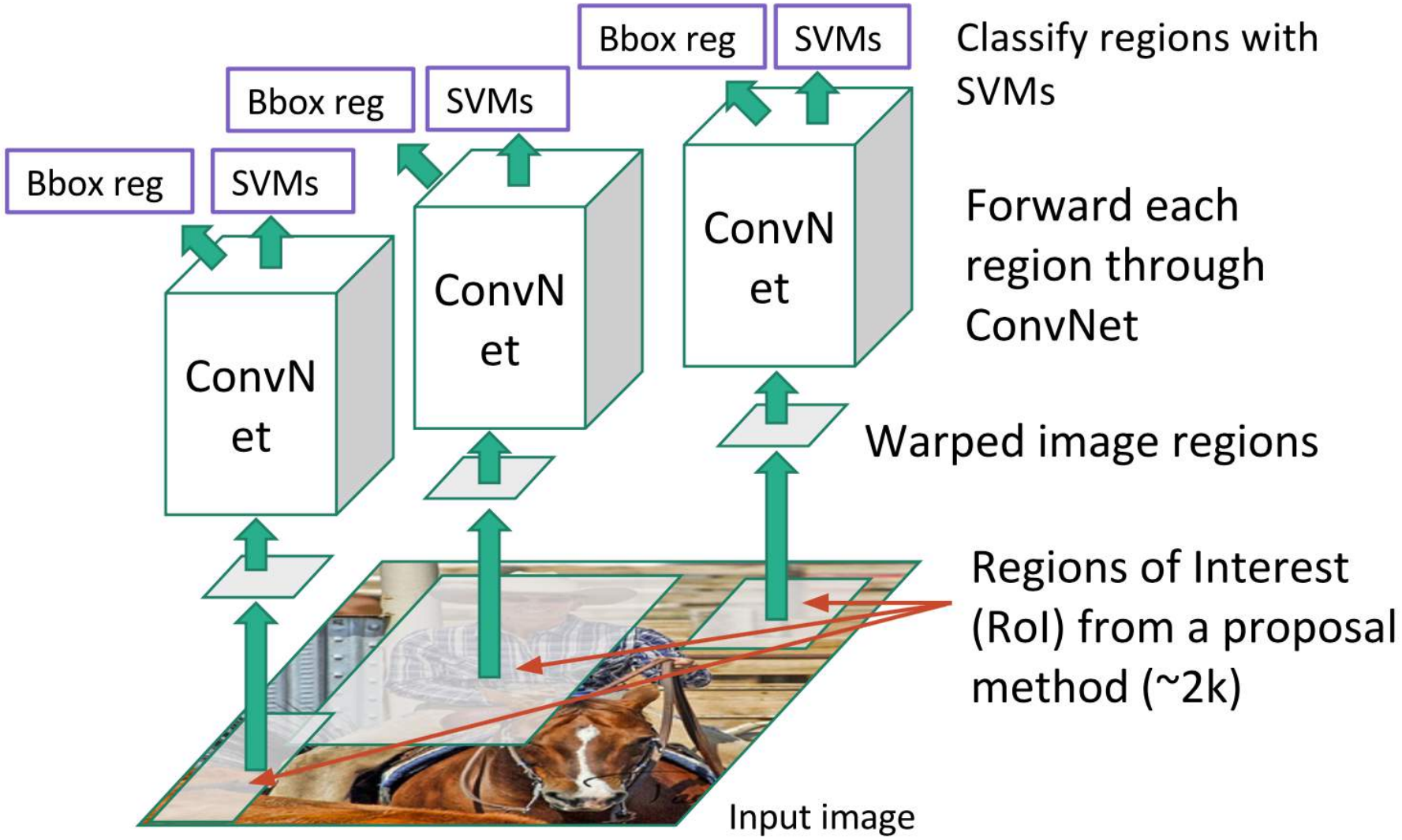
Region Proposals

- Можно использовать методы, не связанные с DL. Например, Selective Search.
- Работает с нормальной скоростью, примерно 2000 регионов за пару секунд.



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Object Detection: R-CNN



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

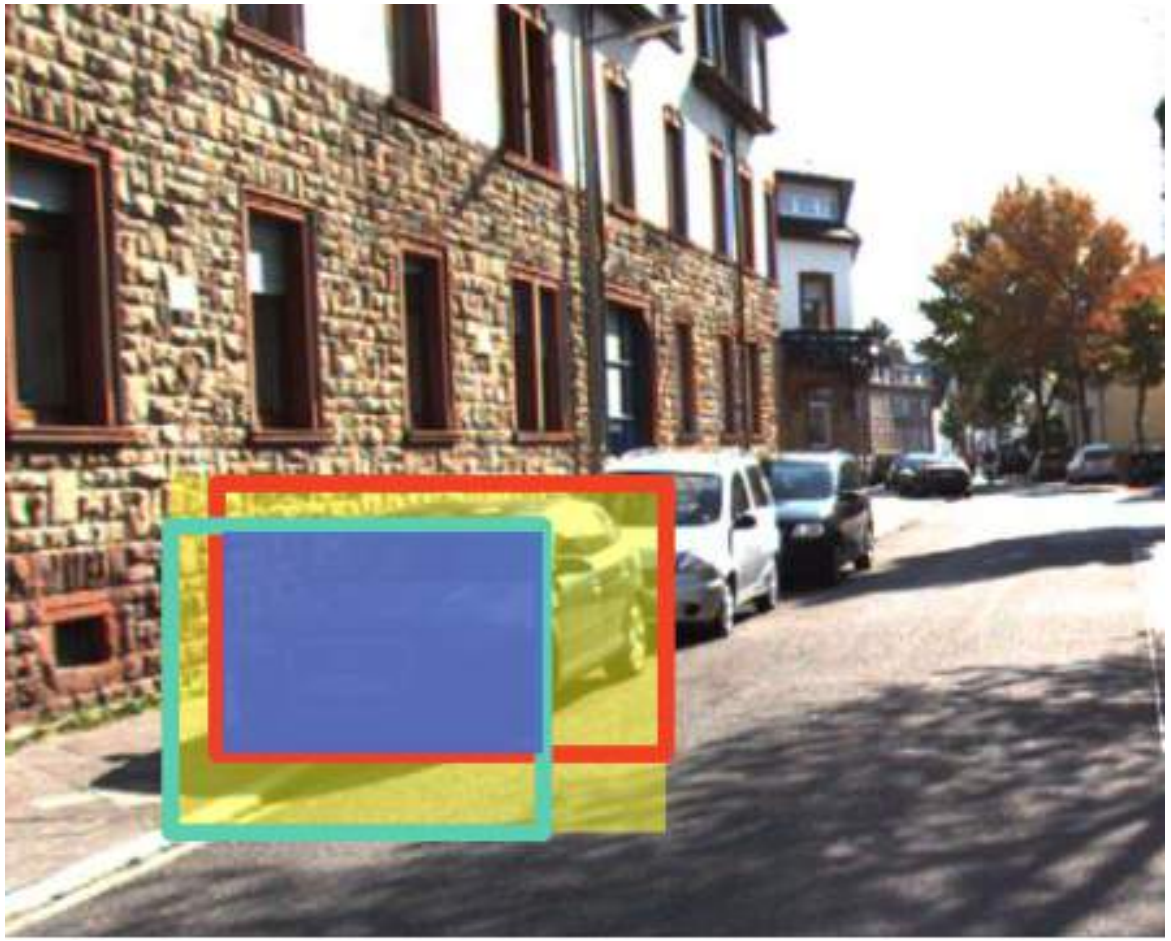
Object Detection: Non-Maximum Suppression

От детектора получается много пересекающихся bbox, покрывающие одни и те же объекты:

1. Фильтруем по убыванию вероятностей

2. Идём по порядку и удаляем все прочие bbox, пересекающие с текущим на >50%

Object Detection: IoU



intersection

union

IoU = intersection / Union

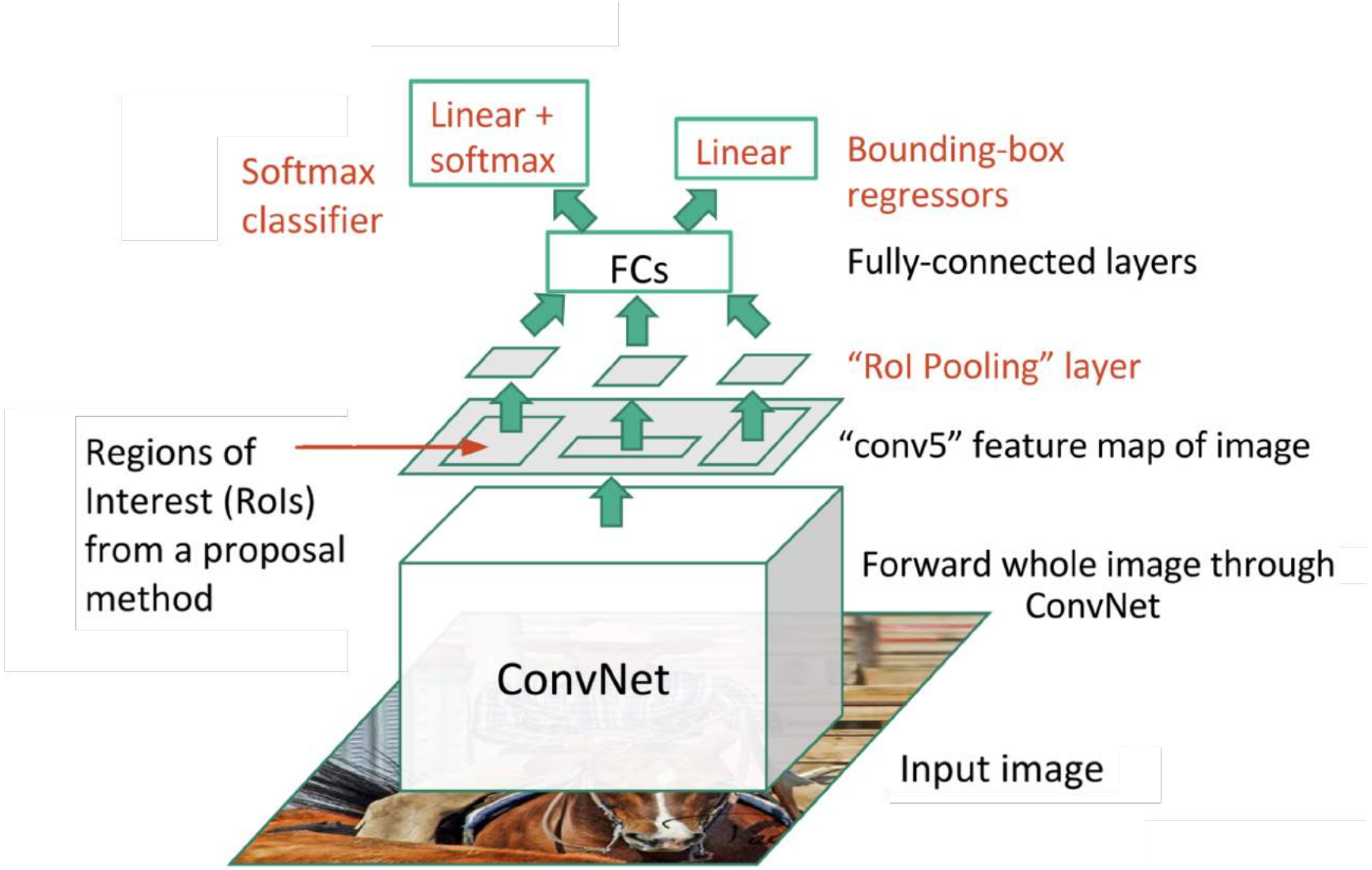
Source: DL course at YDS 2018, lecture 5.

Object Detection: R-CNN Problems

- Очень много разных моделей в одном фреймворке.
  - CNN, которую мы доучиваем в процессе.
  - SVM, который мы тоже доучиваем.
  - BBox Regression, опять же учим.
- Долго учится и требует много памяти.
- Долго работает на inference стадии (47s на картинку).

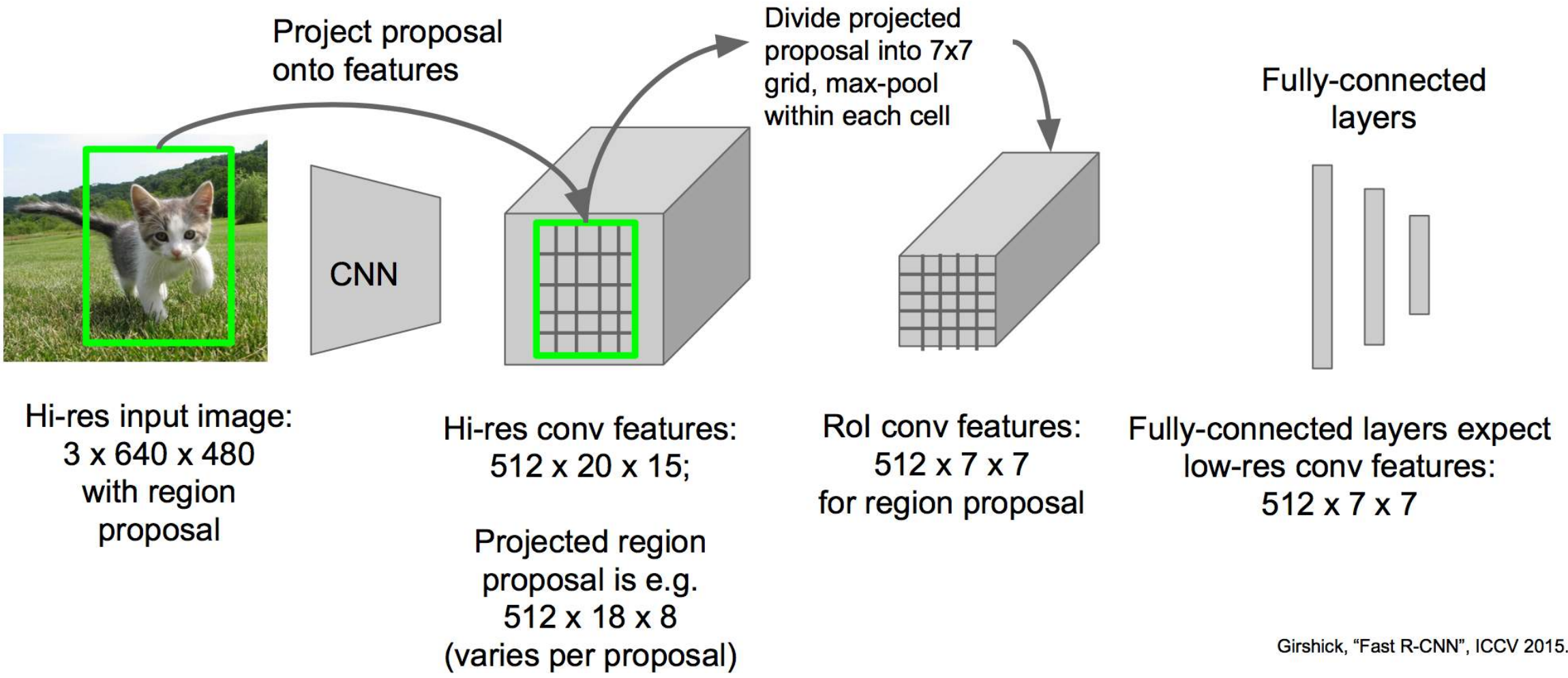
Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Object Detection: Fast R-CNN



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

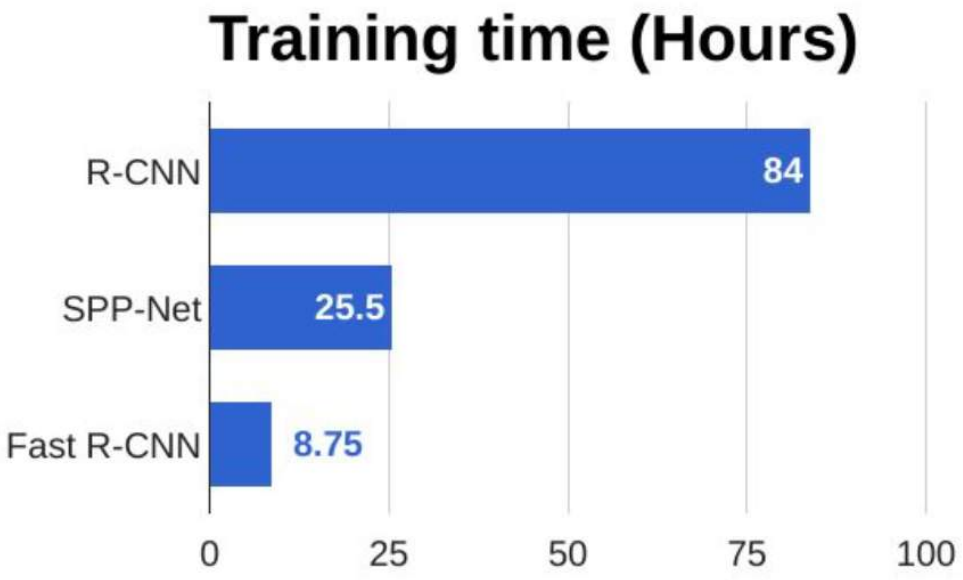
Object Detection: RoI Pooling

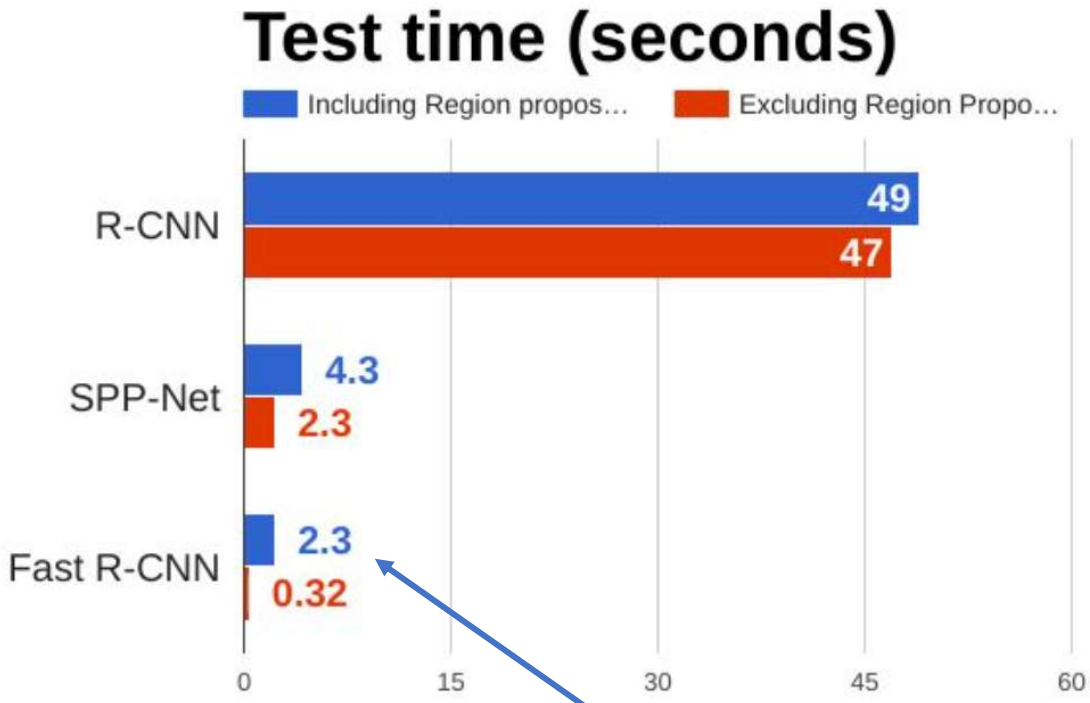


Girshick, "Fast R-CNN", ICCV 2015.

Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Object Detection: Runtimes

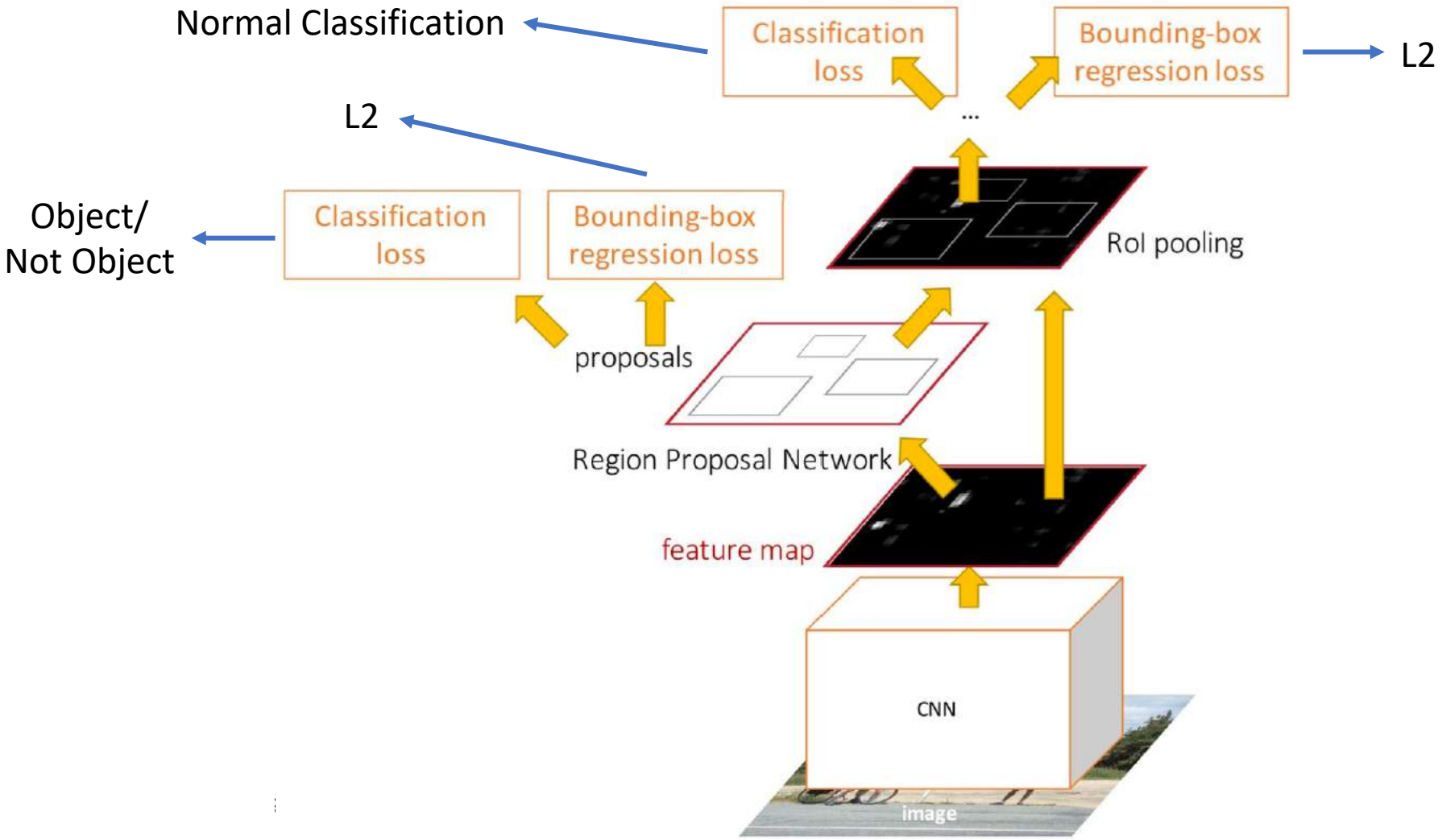




Медленно из-за алгоритма выделения RoI.

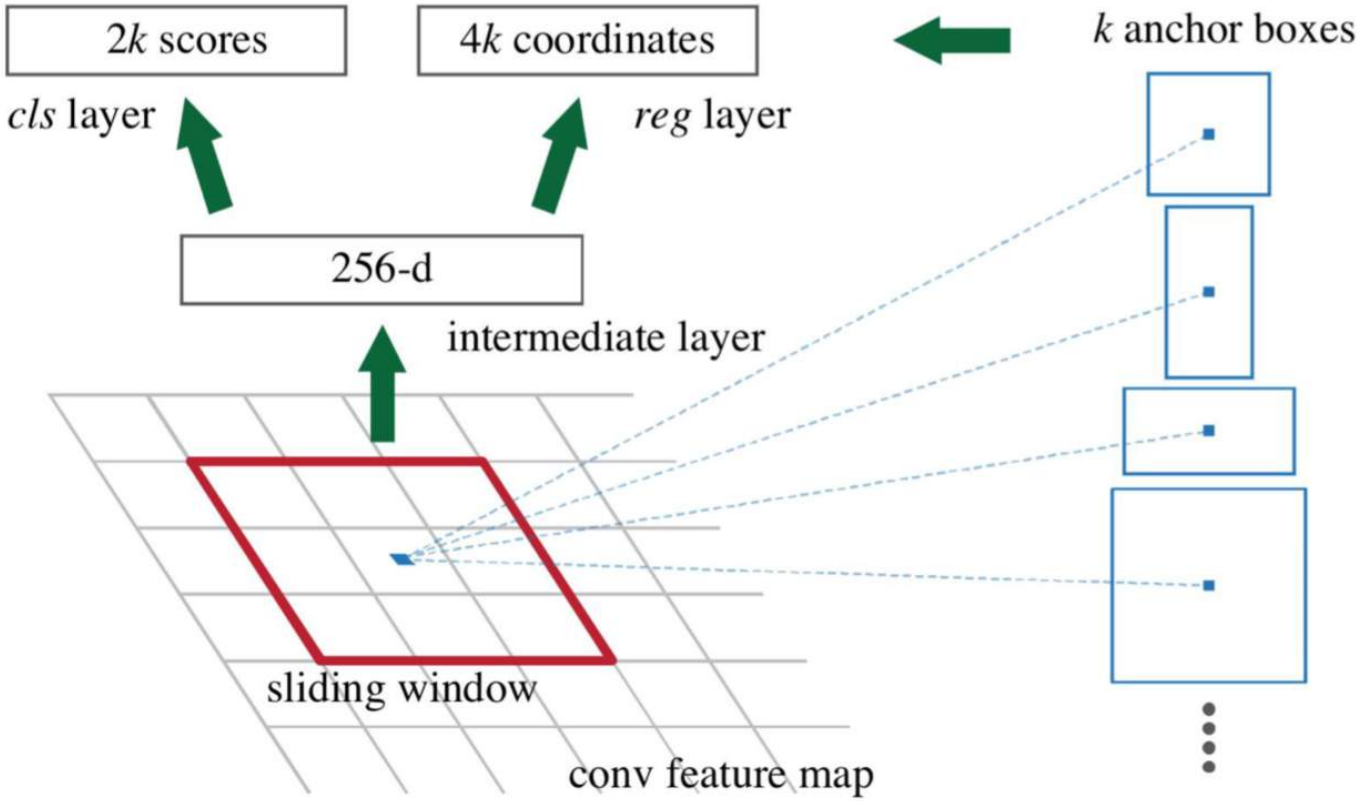
Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Object Detection: Faster R-CNN



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

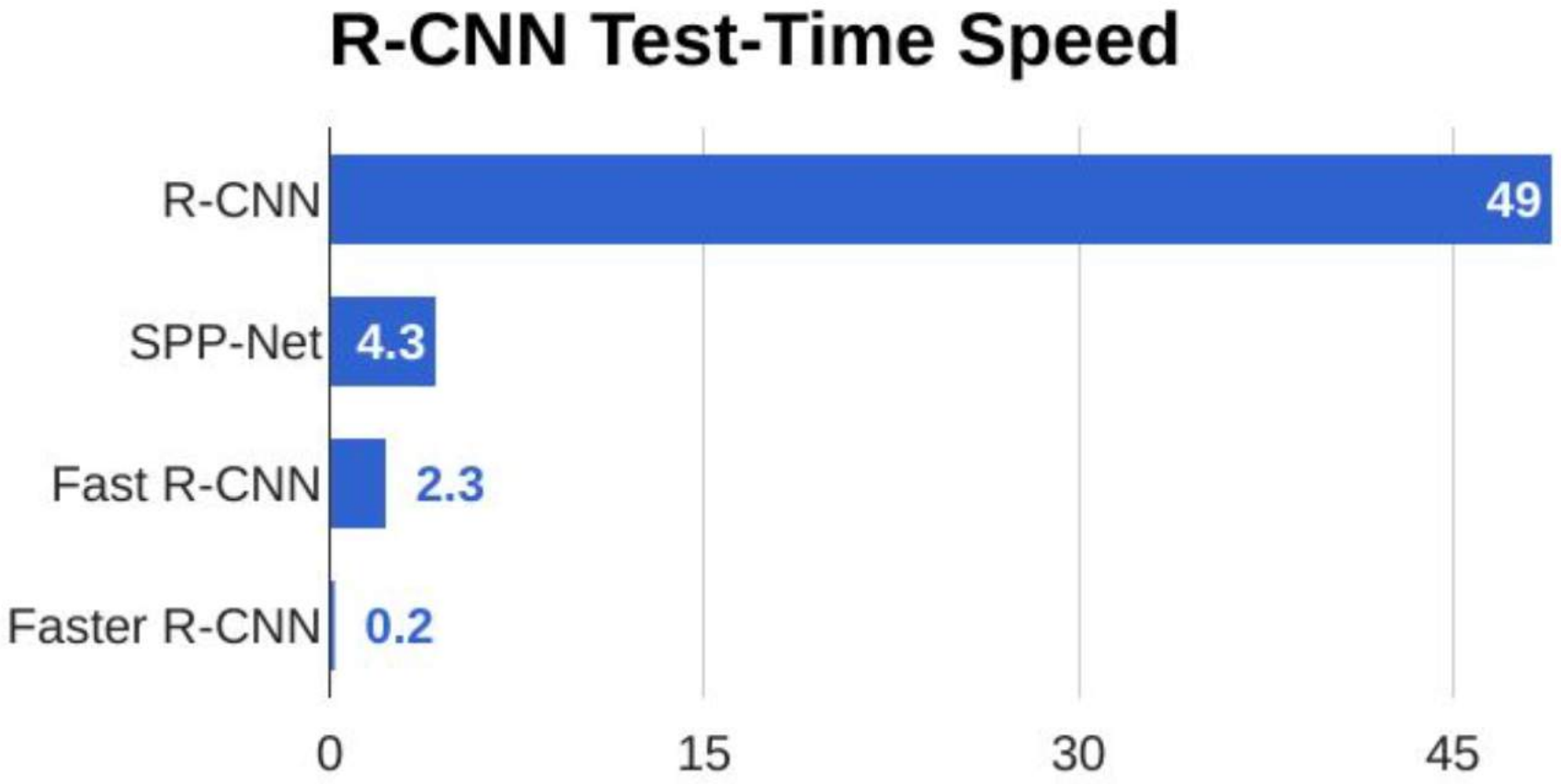
Object Detection: Region Proposal Network



1. Небольшим скользящим окном двигаемся по признакам-активациям
2. Оцениваем шансы на объект + корректируем рамки
3. Окно разных размеров и соотношений
4. Из 17000 оставим 300 proposal

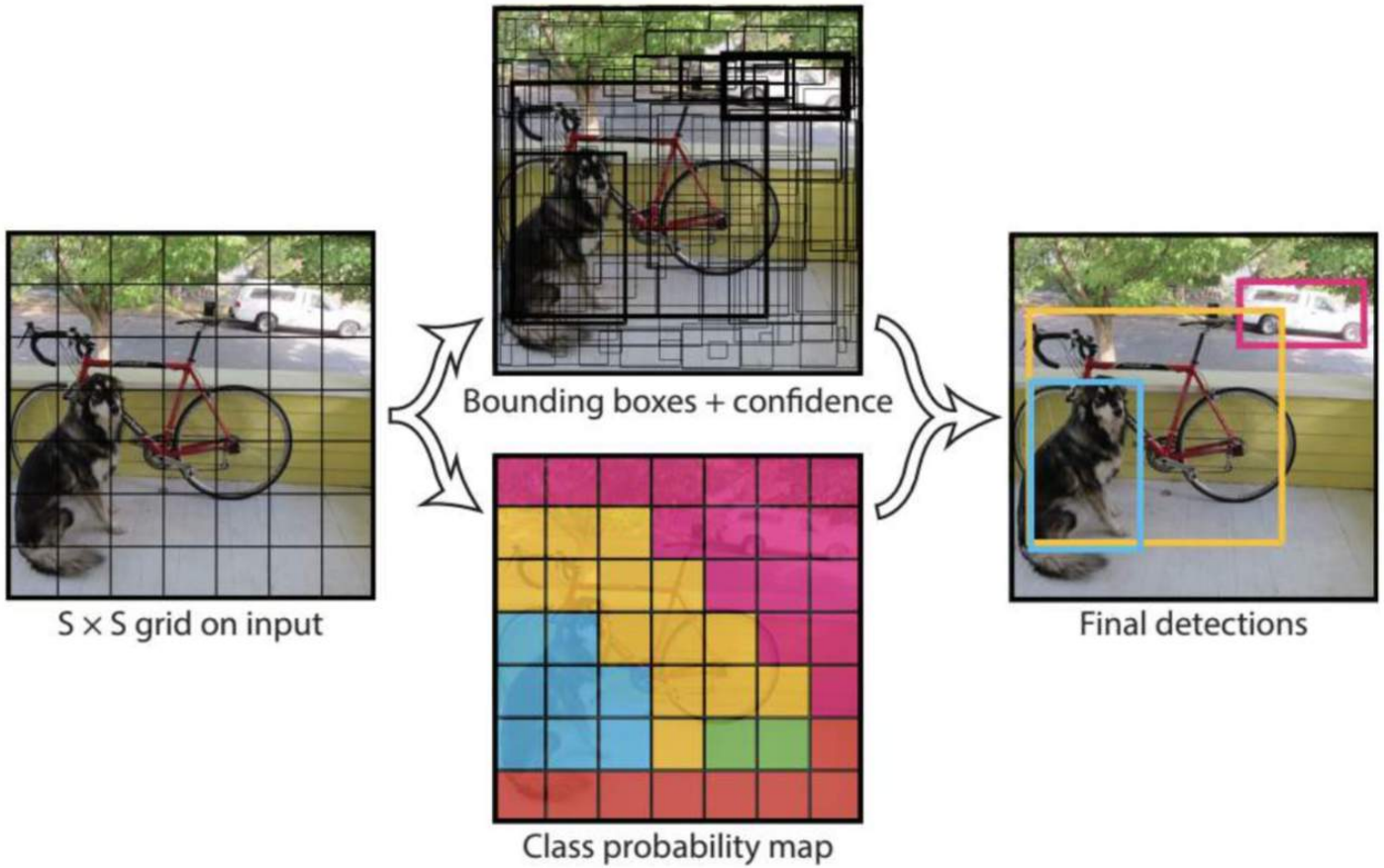
Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Object Detection: Runtimes



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

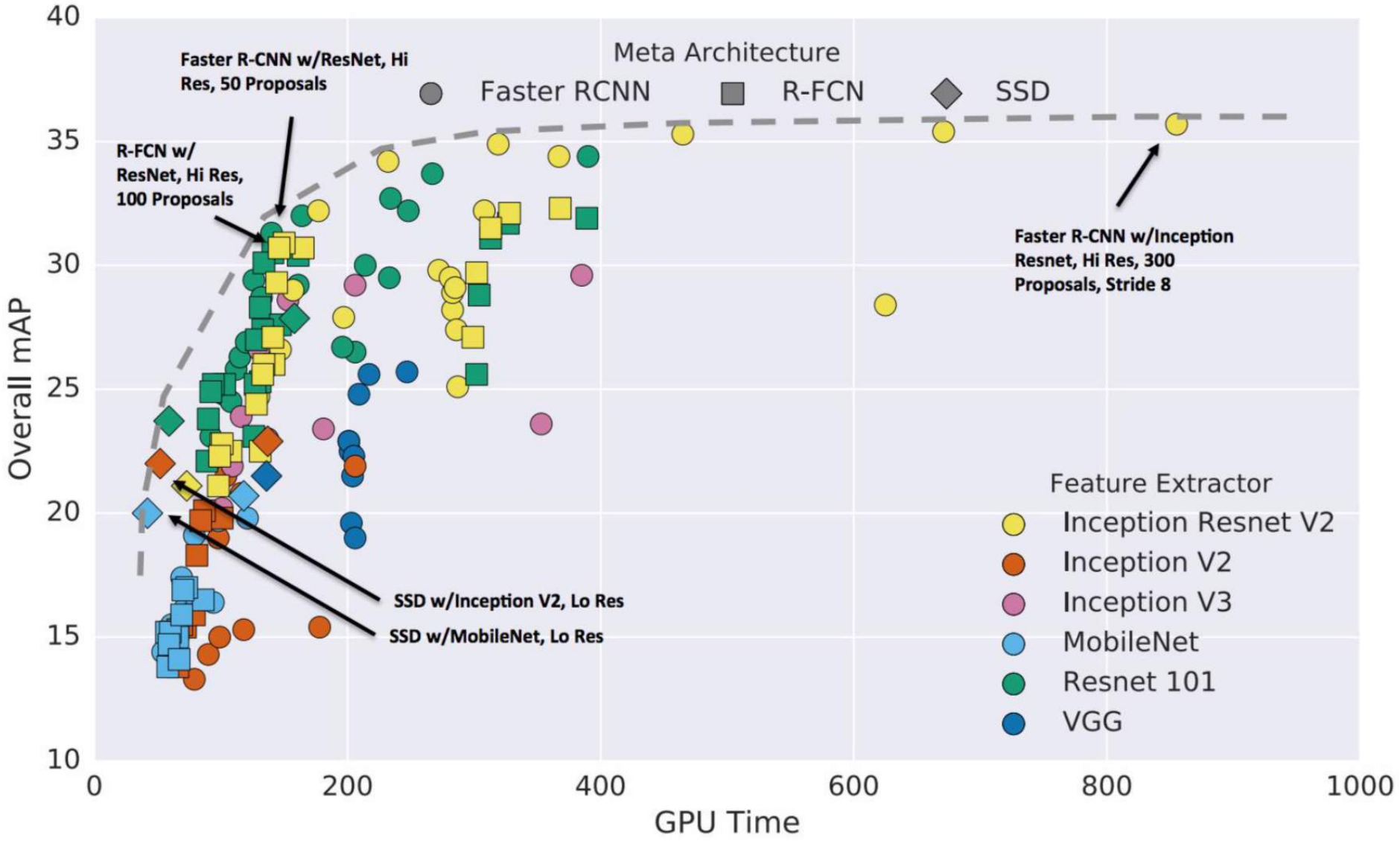
Object Detection: YOLO



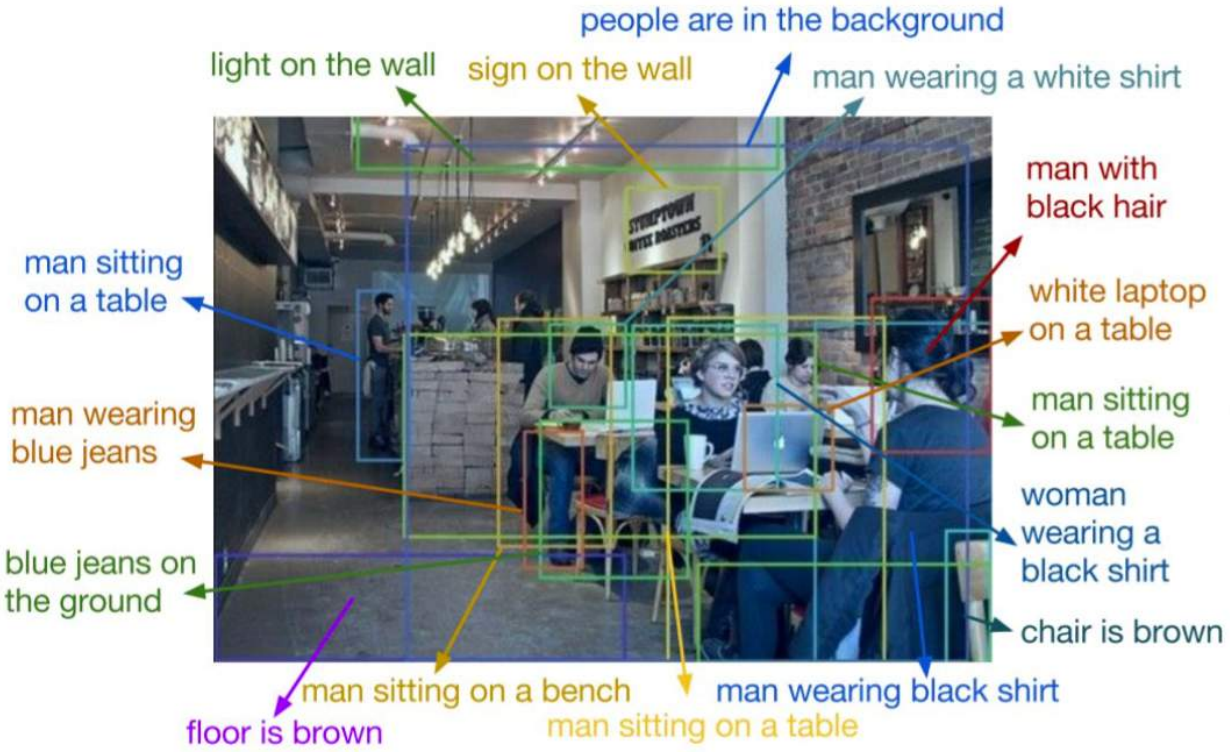
- Делим изображение на блоки
- Для каждого блока предсказывается распределение классов + координаты B штук bbox с уверенностями
- Фильтруем и находим основные

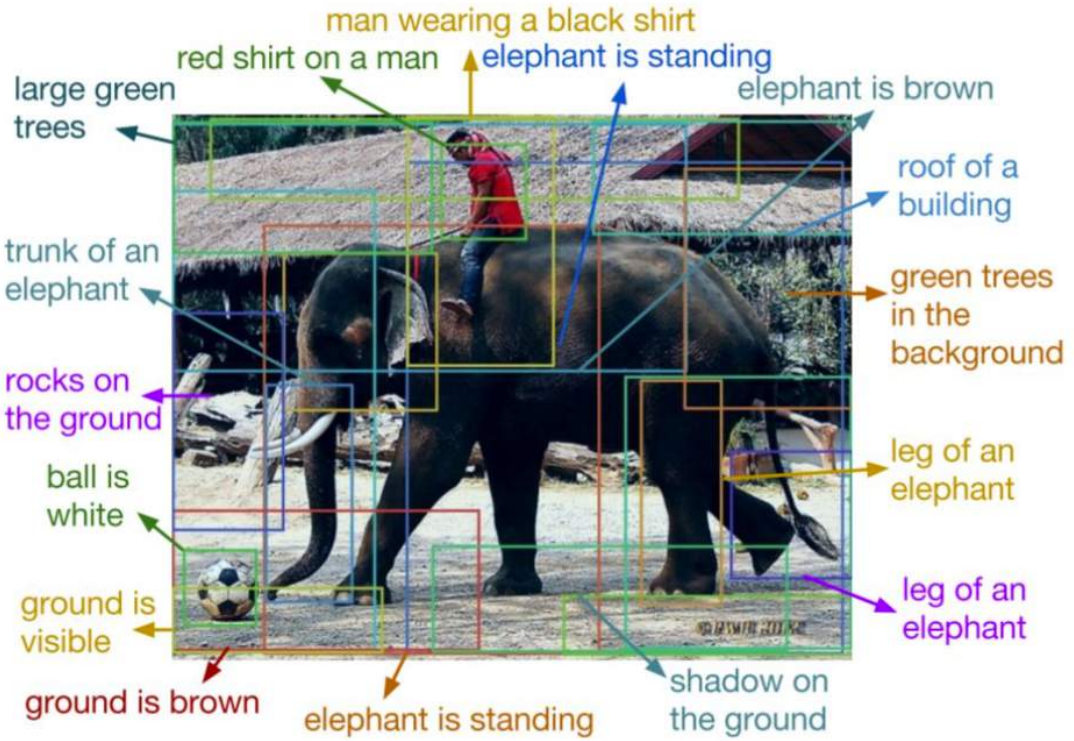
Главное преимущество в том, что это быстрее, чем Real-Time.
Хотя и не так точно.

Object Detection: Architectures



Dense Captioning





Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Instance Segmentation

Задача:

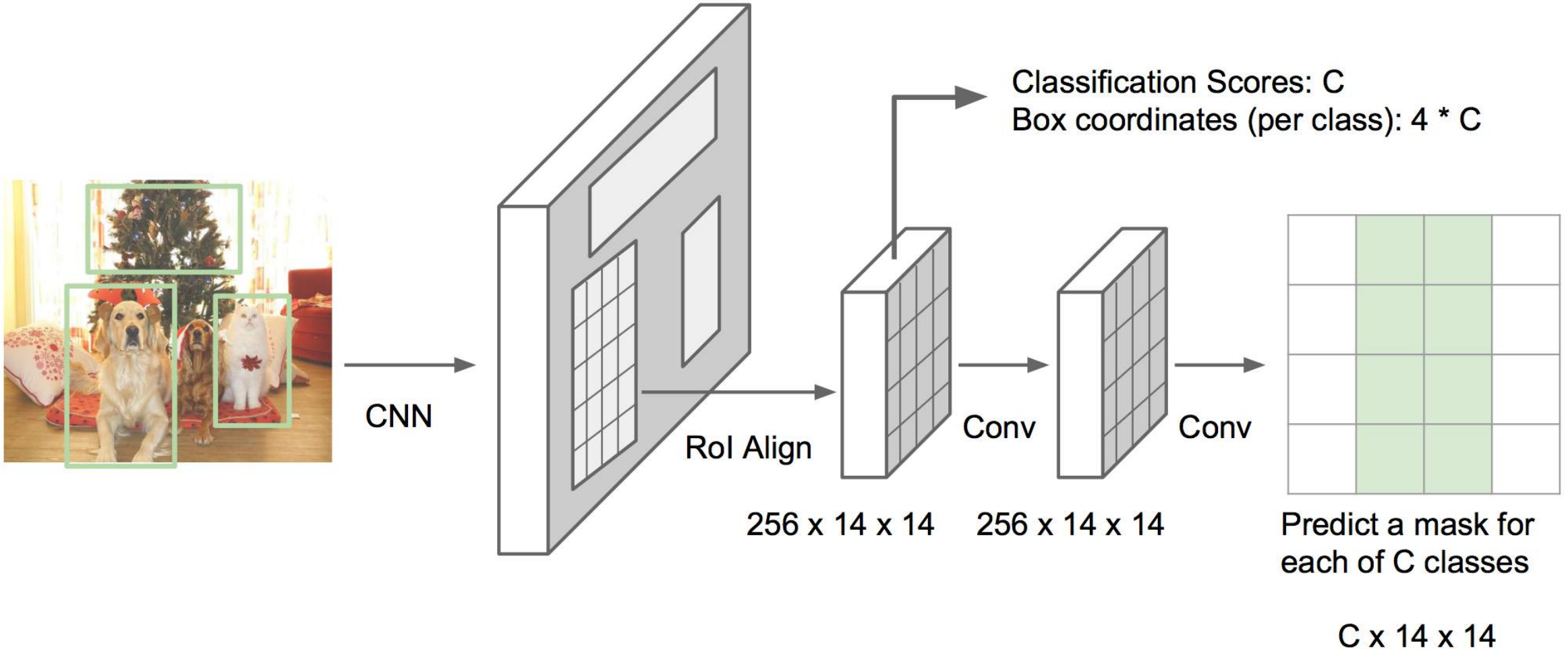
Дать лейбл каждому пикселю на картинке. При этом отличая объекты одного класса друг от друга.

Важная деталь:

Мы имеем дело с неизвестным количеством объектов.

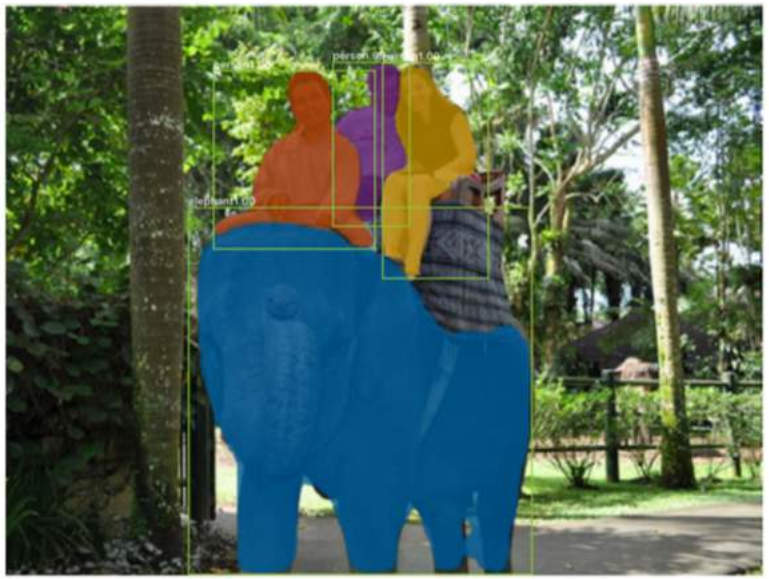
Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

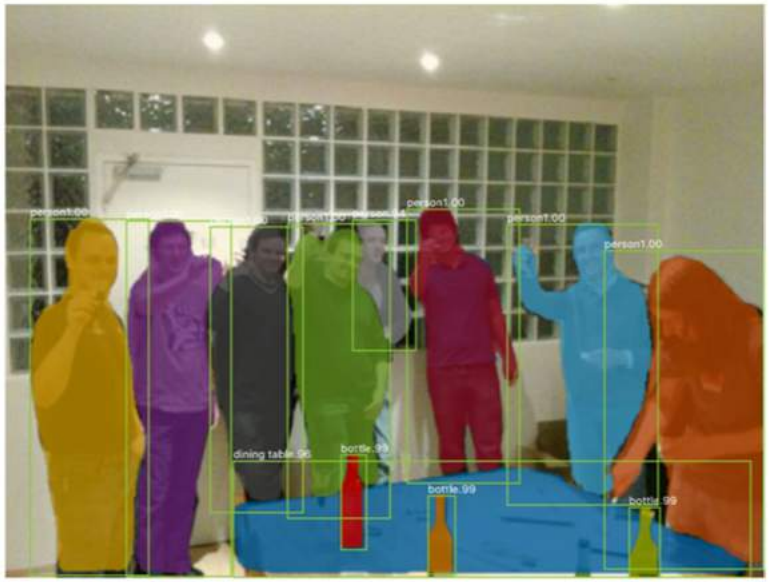
Instance Segmentation: Mask R-CNN



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

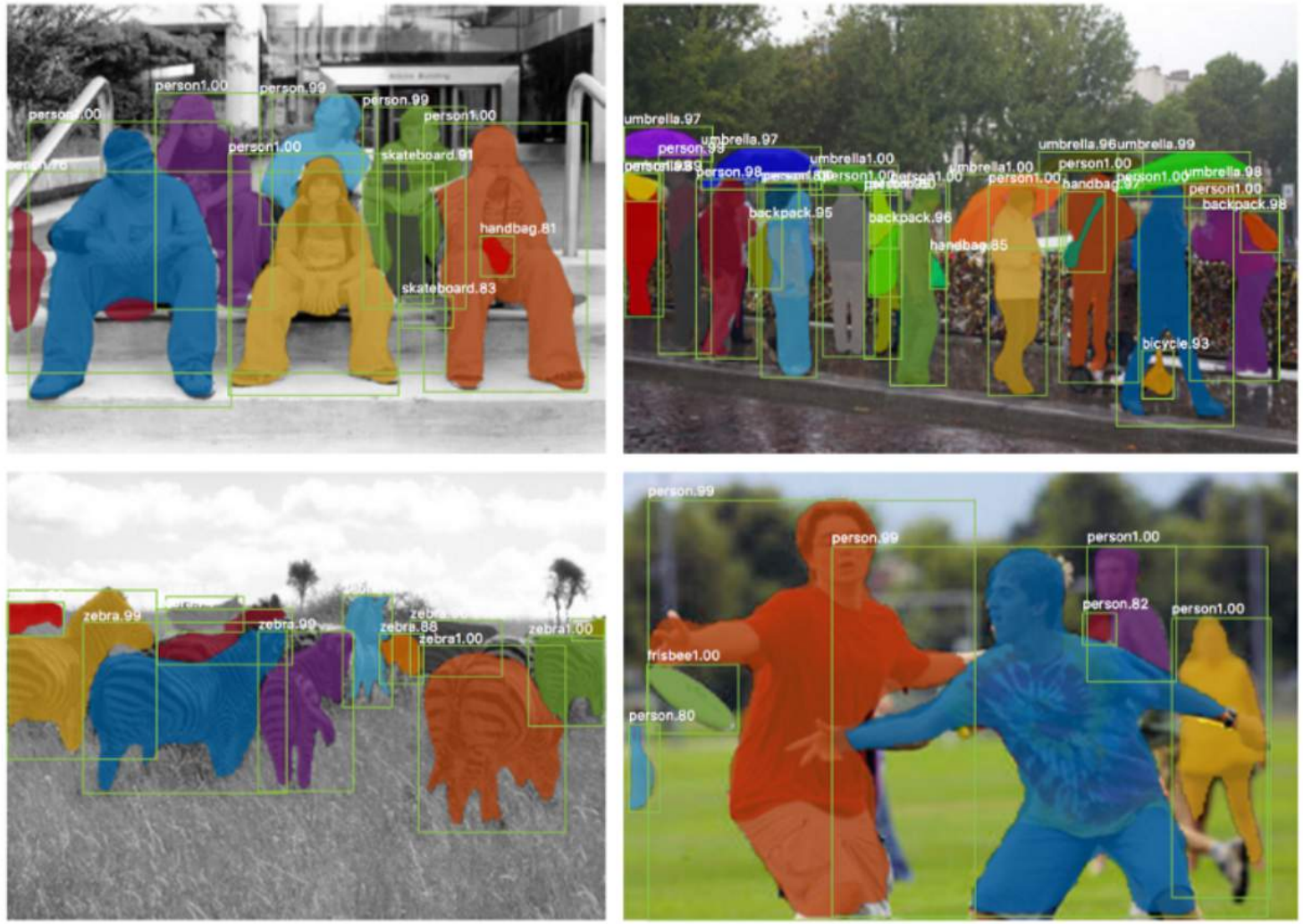
Instance Segmentation: Mask R-CNN





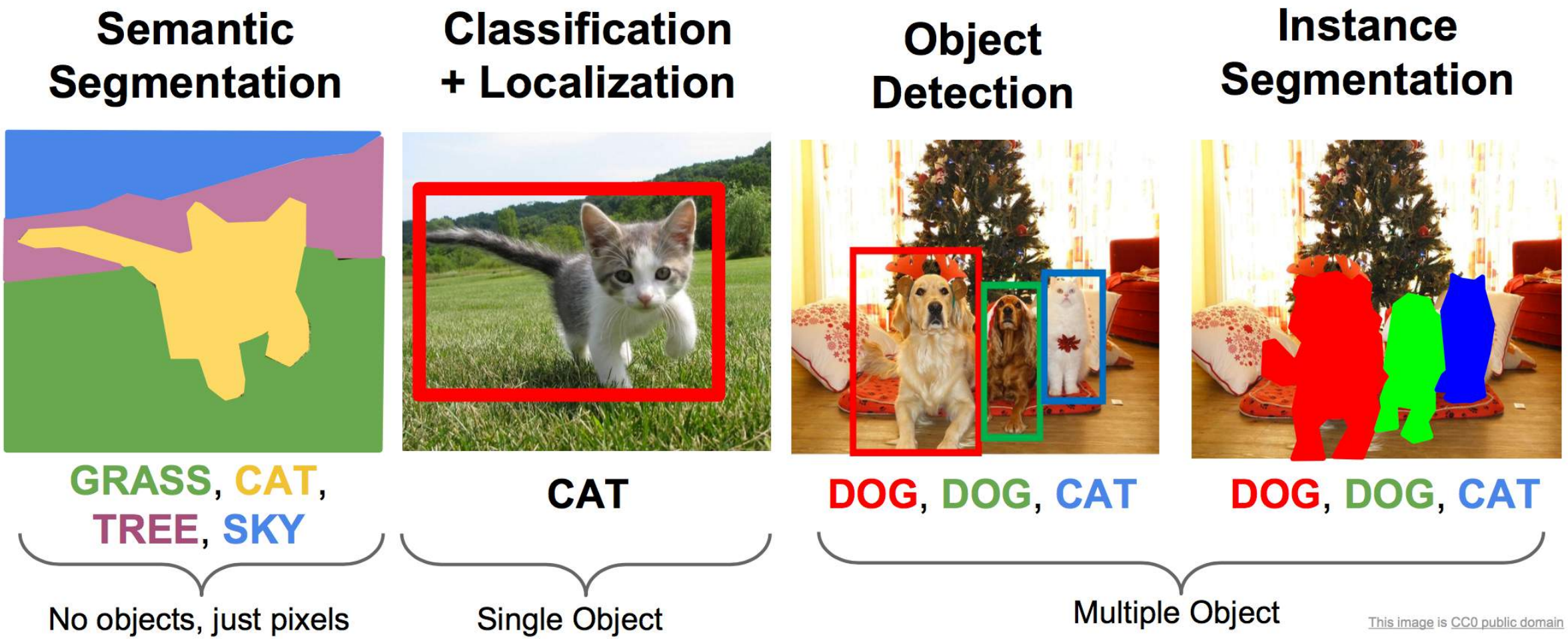
Source: https://arxiv.org/pdf/1703.06870.pdf

Instance Segmentation: Mask R-CNN



Source: https://arxiv.org/pdf/1703.06870.pdf

Recap



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Useful Materials

Лекции

- cs231n 2017 Lecture 11 video: https://www.youtube.com/watch?v=nDPWywWRIRo&list=PL3FW7Lu3i5JvHM8ljYj-zLfQRF3EO8sYv
- cs231n 2017 Lecture 11 Slides: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Статьи

- Mask-RCNN: https://arxiv.org/pdf/1703.06870.pdf
- U-Net: https://arxiv.org/pdf/1505.04597.pdf)和%5bTiramisu%5d(https://arxiv.org/abs/1611.09326.pdf
- Faster R-CNN: http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-networks.pdf
- YOLO-9000: http://openaccess.thecvf.com/content_cvpr_2017/papers/Redmon_YOLO9000_Better_Faster_CVPR_2017_paper.pdf