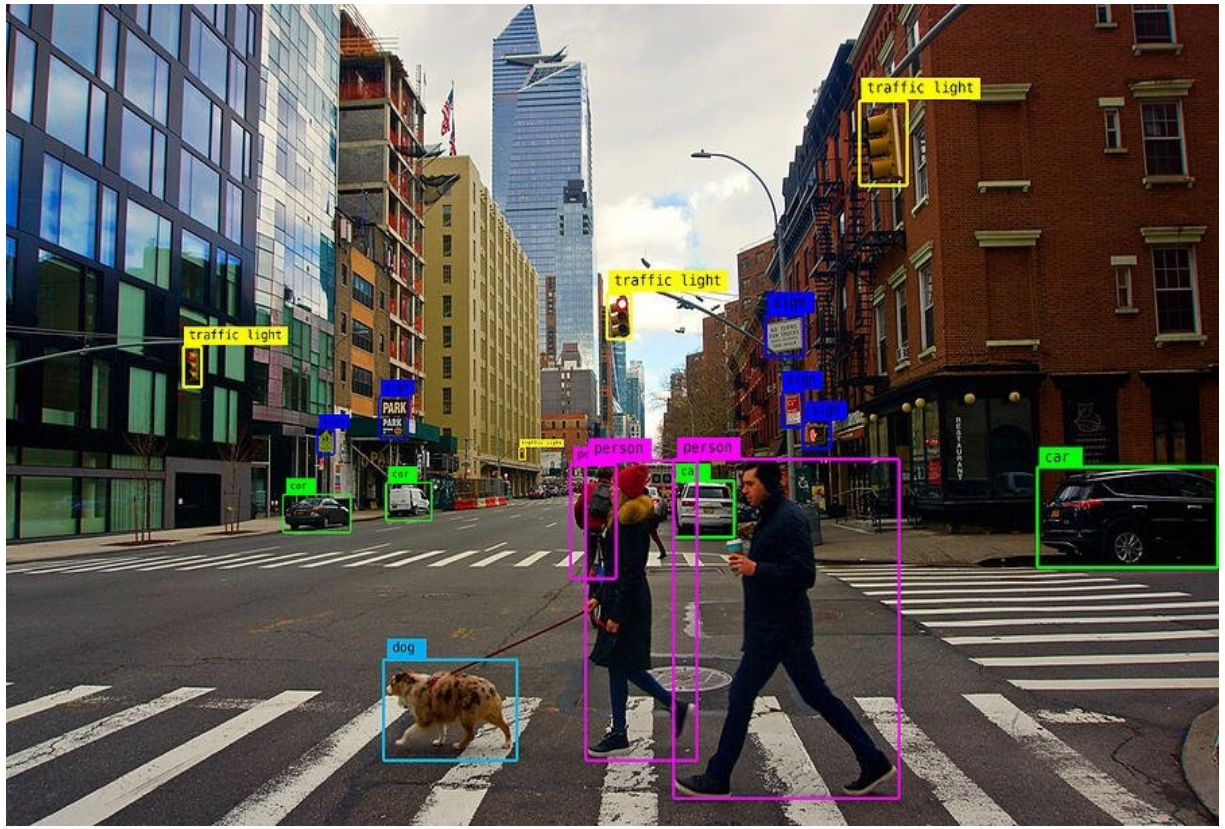
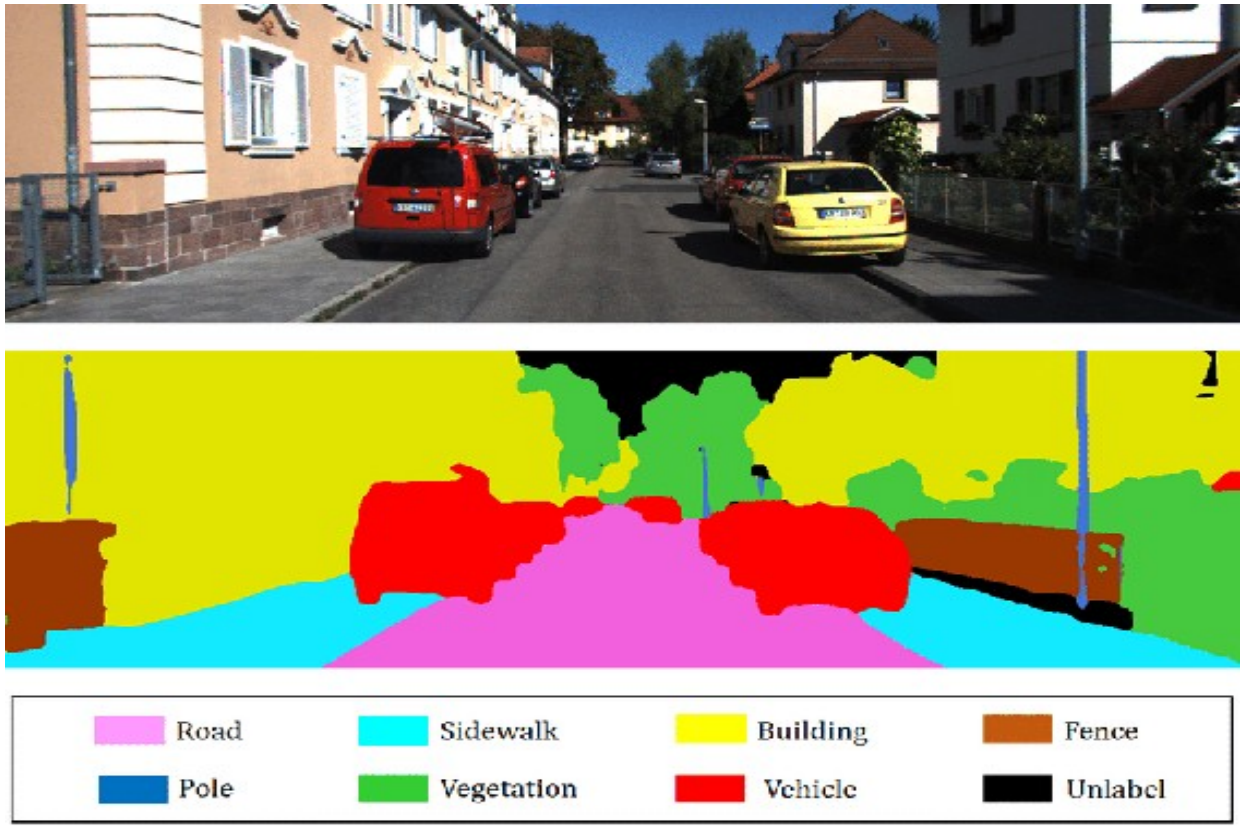
Глубокое обучение для обработки изображений

Лекция 8

Детекция и сегментация





Примеры задач детекции объектов (слева)
и семантической сегментации (справа)

Детекция объектов (Object Detection)

Самый простой способ: рассматривать небольшие части изображения и для каждого из них делать предсказание, есть ли объект или нет

Детекция объектов (Object Detection)

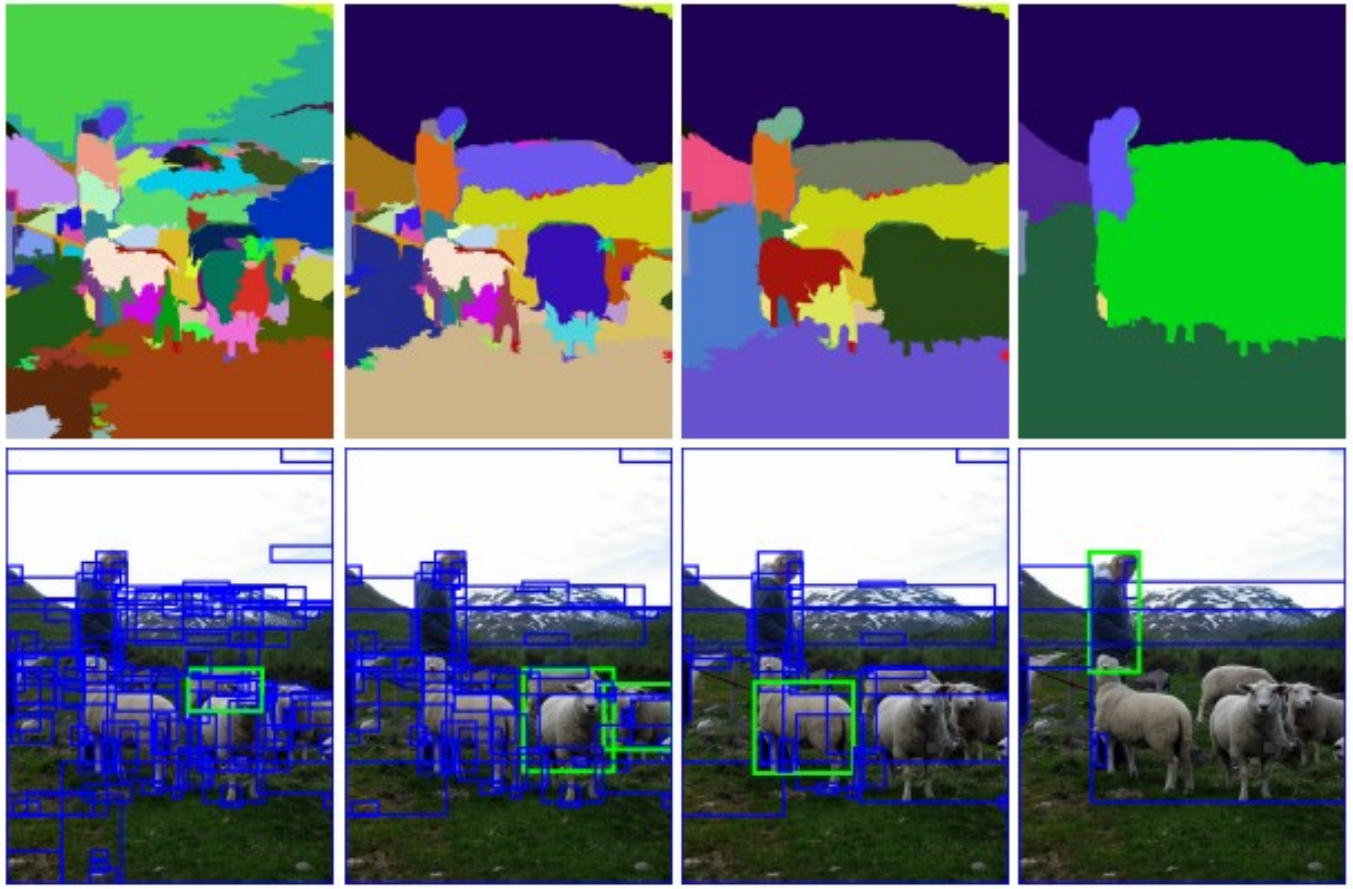
Самый простой способ: рассматривать небольшие части изображения и для каждого из них делать предсказание, есть ли объект или нет

Детекция объектов (Object Detection)

Самый простой способ: рассматривать небольшие части изображения и для каждого из них делать предсказание, есть ли объект или нет

Объекты могут быть разного размера, их может быть много и т.д., поэтому нам потребуется использовать тысячи таких областей разного масштаба

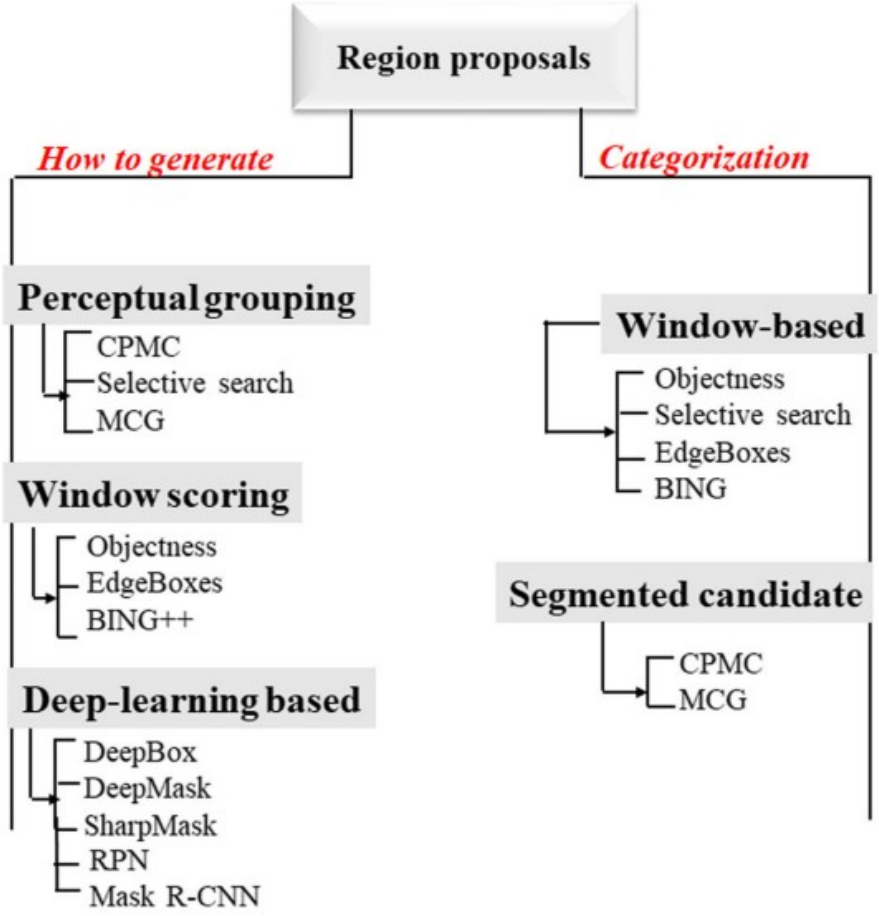
Region proposal



Идея состоит в том, чтобы с помощью, например, алгоритма Selective Search найти области на картинке, которые могут содержать объекты. Это позволит не перебирать множество различных областей простым перебором.

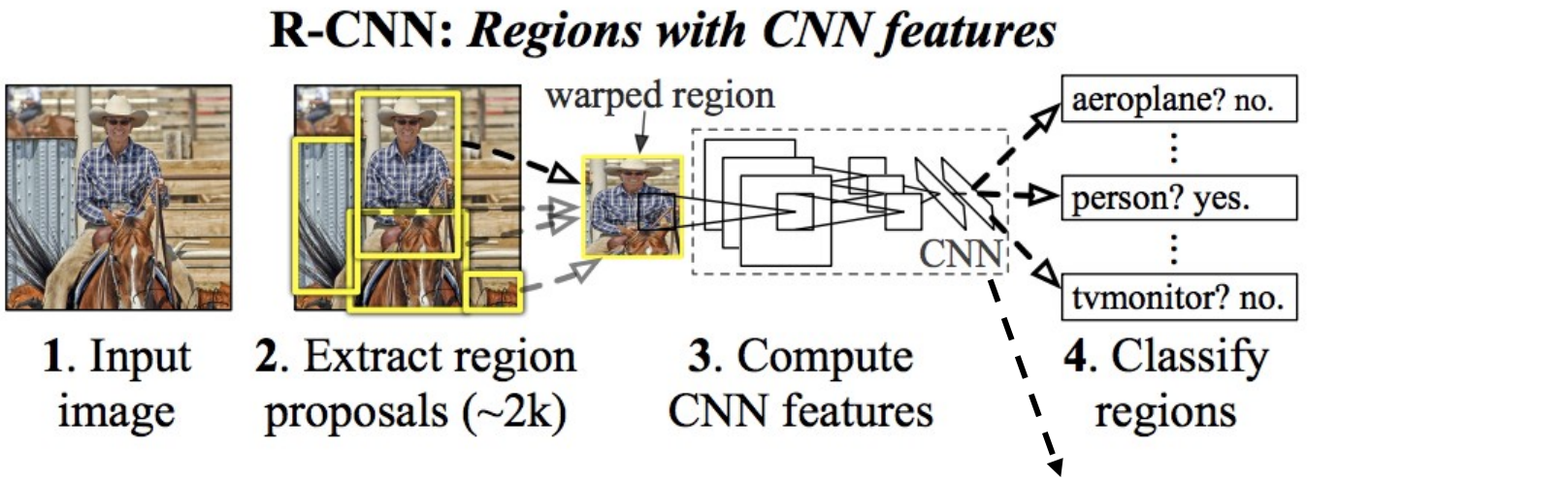
Selective Search for Object Recognition

Region proposal



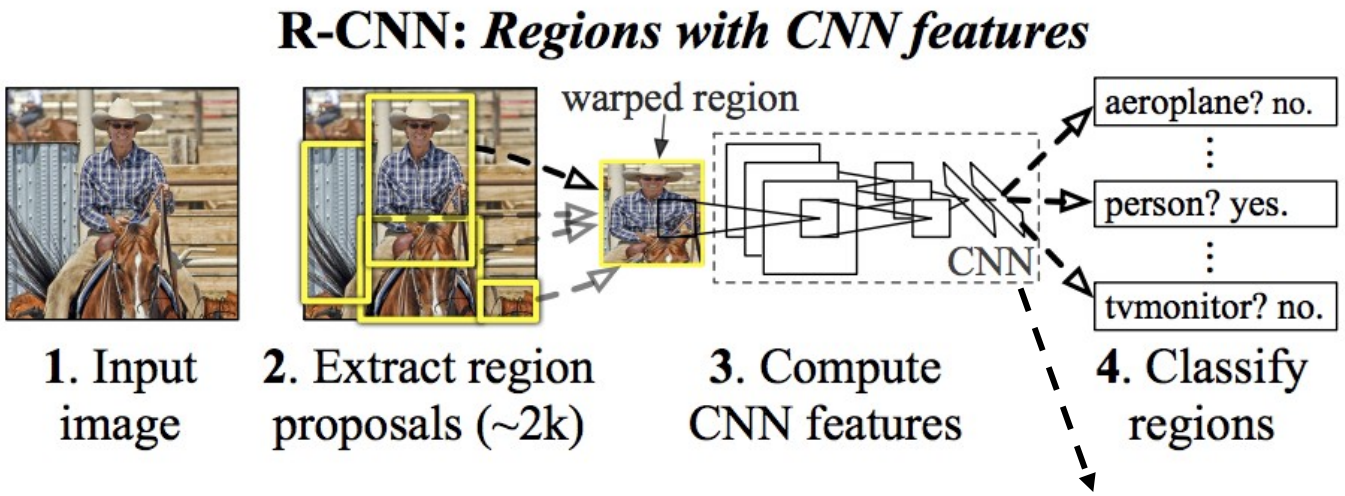
A class-independent flexible algorithm to generate region proposals

R-CNN (Region-based CNN)



[Rich feature hierarchies for accurate object detection and semantic segmentation]

R-CNN (Region-based CNN)

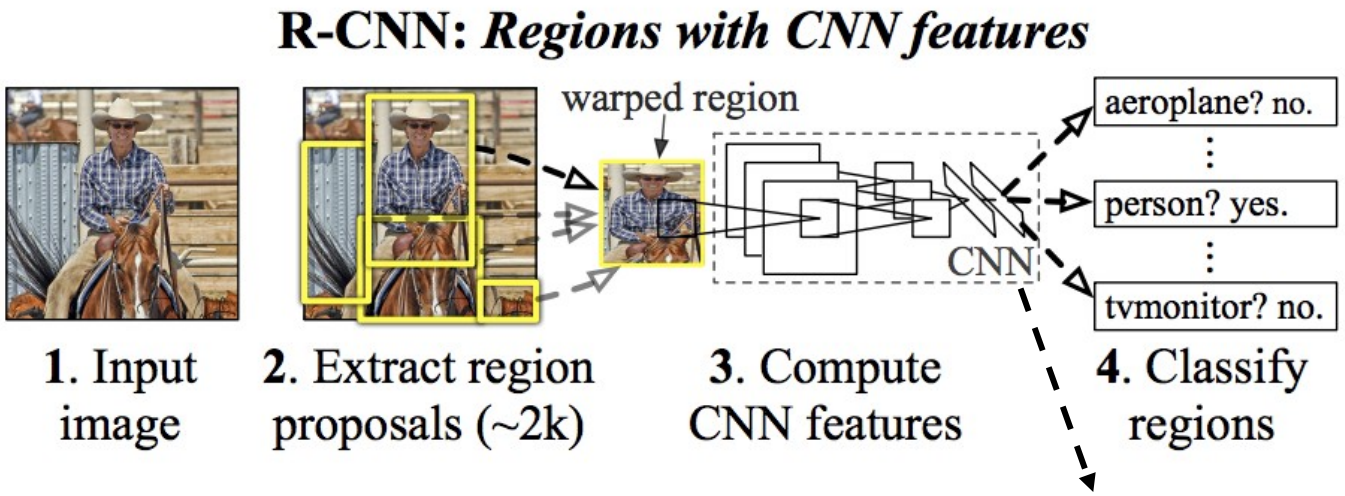


5. Bounding Box Regressor
[x, y, w, h]

Предсказывает 4 числа – уточняют расположение бокса внутри области, предсказанной на region proposal stage

[Rich feature hierarchies for accurate object detection and semantic segmentation]

R-CNN (Region-based CNN)



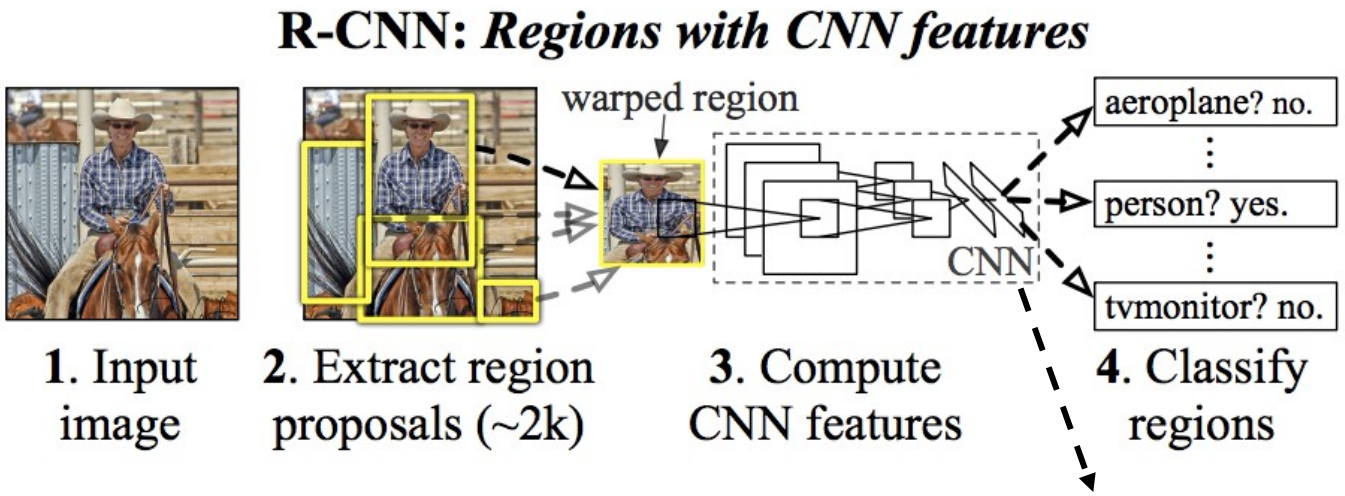
Комбинированная **функция потерь**:
предсказание класса (**cross-entropy loss**) +
SVM классификатор (**Hinge loss**) +
предсказание координат бокса (**MSE**)

5. Bounding Box Regressor
[x, y, w, h]

Предсказывает 4 числа – уточняют расположение бокса внутри области, предсказанной на region proposal stage

[Rich feature hierarchies for accurate object detection and semantic segmentation]

R-CNN (Region-based CNN)



Комбинированная **функция потерь**:
предсказание класса (**cross-entropy loss**) +
SVM классификатор (**Hinge loss**) +
предсказание координат бокса (**MSE**)

5. Bounding Box Regressor
[x, y, w, h]

Предсказывает 4 числа – уточняют расположение бокса внутри области, предсказанной на region proposal stage

Время предсказания ~ 1 минута на картинку

[Rich feature hierarchies for accurate object detection and semantic segmentation]

Fast R-CNN

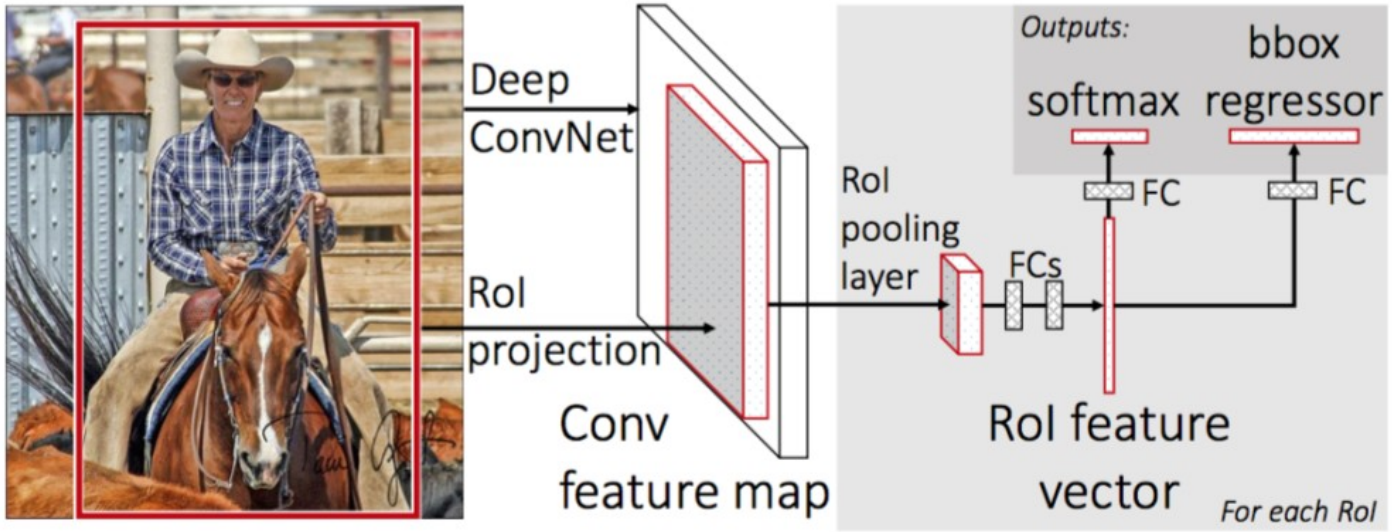


Figure 1. Fast R-CNN architecture. An input image and multiple regions of interest (RoIs) are input into a fully convolutional network. Each RoI is pooled into a fixed-size feature map and then mapped to a feature vector by fully connected layers (FCs). The network has two output vectors per RoI: softmax probabilities and per-class bounding-box regression offsets. The architecture is trained end-to-end with a multi-task loss.

Fast R-CNN

Fast R-CNN

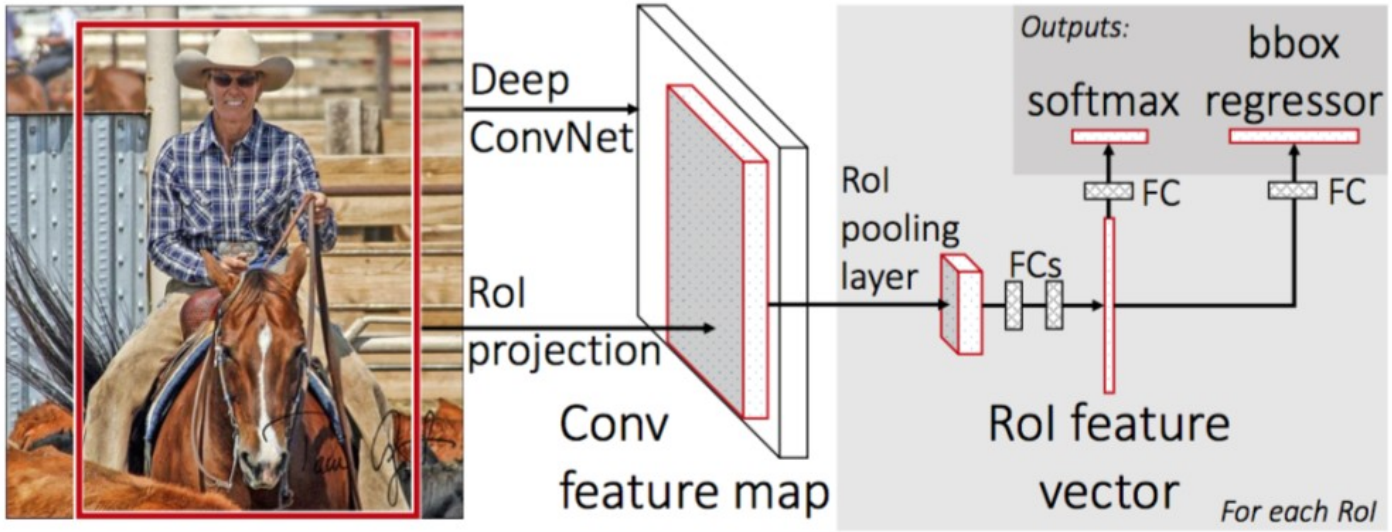
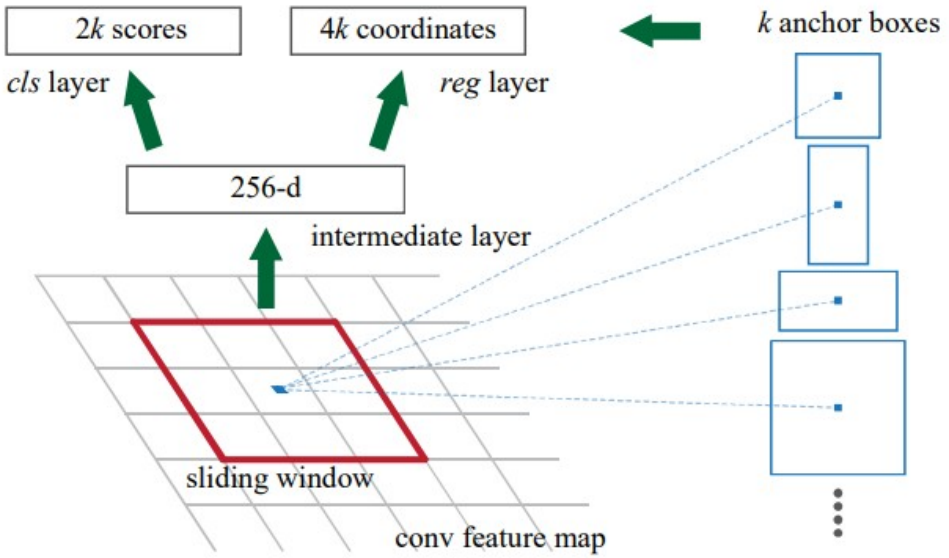


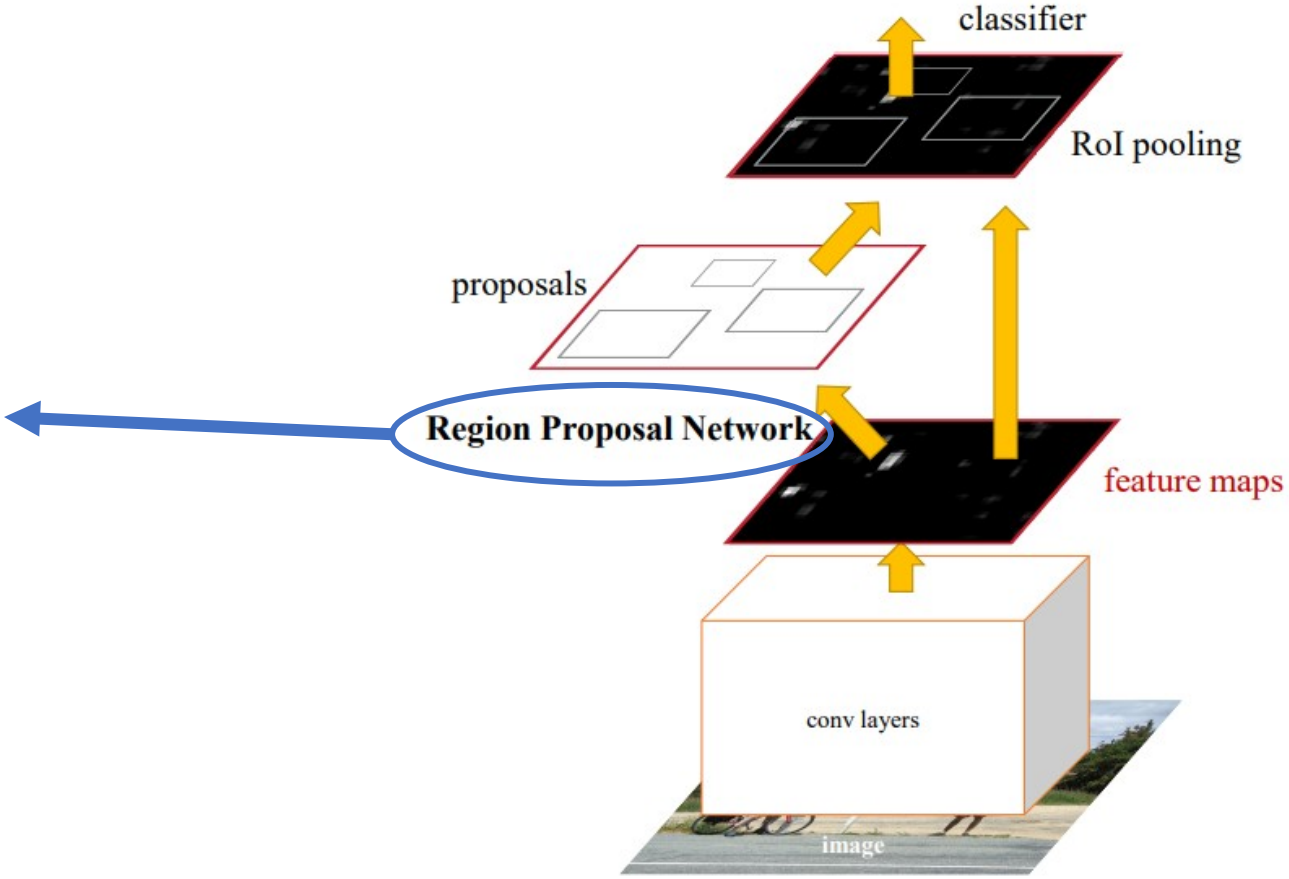
Figure 1. Fast R-CNN architecture. An input image and multiple regions of interest (RoIs) are input into a fully convolutional network. Each RoI is pooled into a fixed-size feature map and then mapped to a feature vector by fully connected layers (FCs). The network has two output vectors per RoI: softmax probabilities and per-class bounding-box regression offsets. The architecture is trained end-to-end with a multi-task loss.

Время предсказания ~ 2-3 секунды на картинку. Теперь поиск регионов занимает большую часть времени.

[Fast R-CNN]

Faster R-CNN

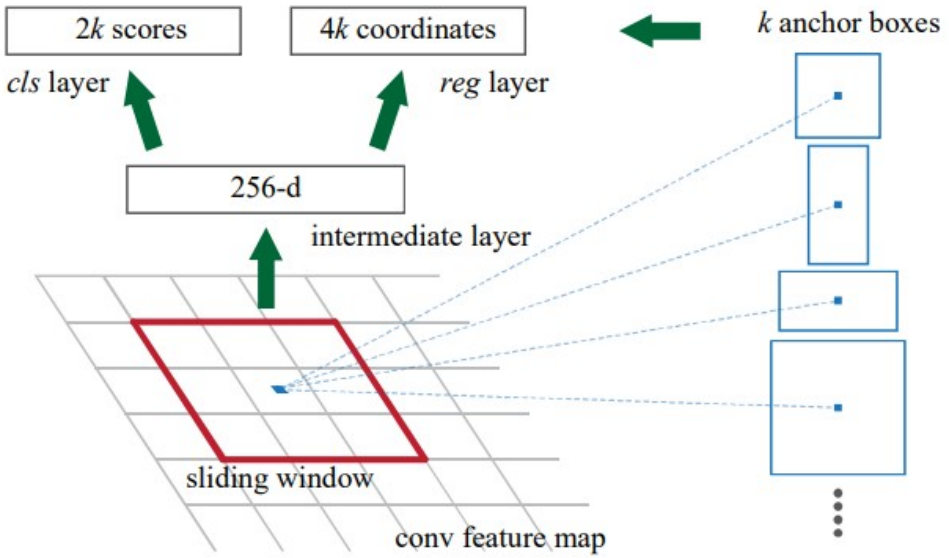


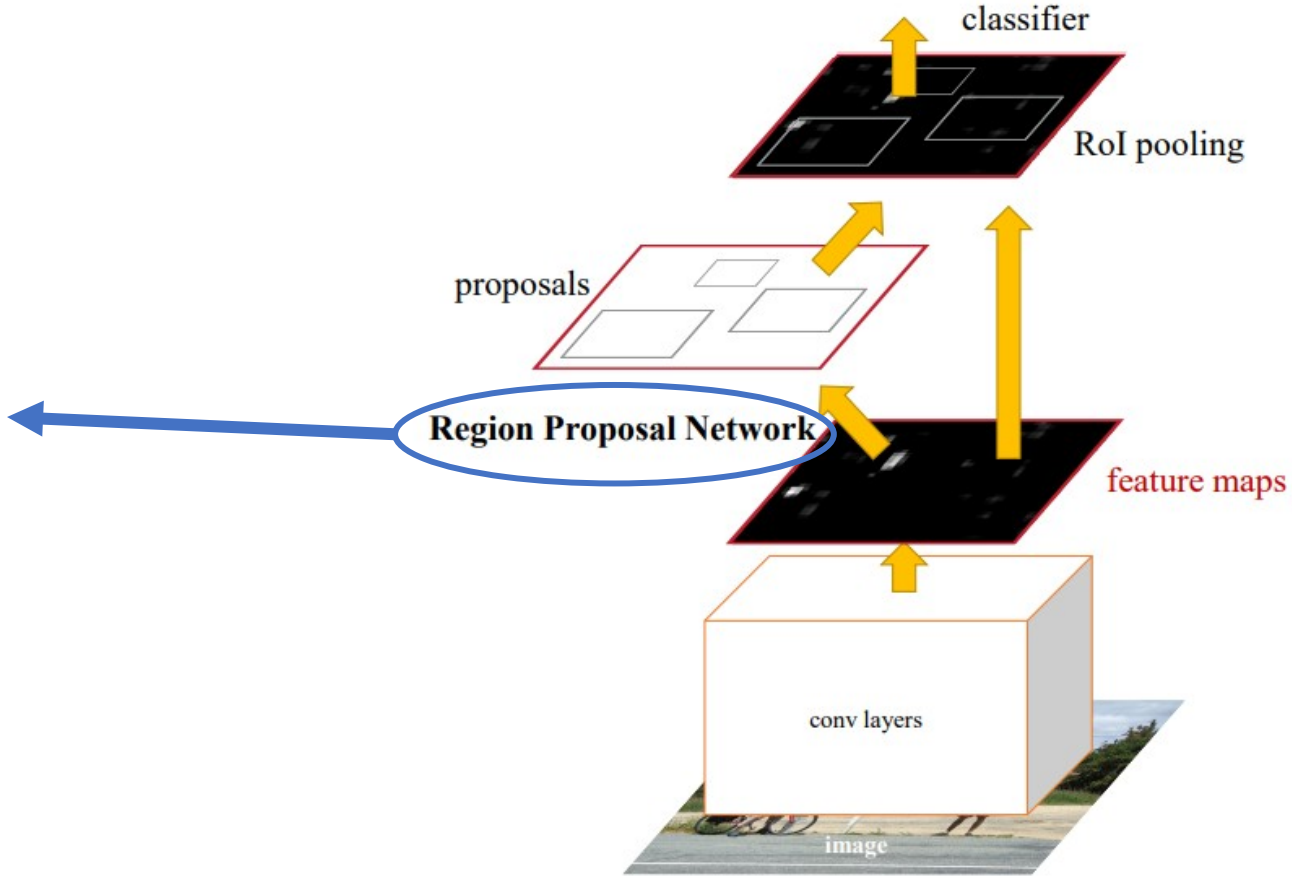


Теперь region proposals предсказываются с помощью Region Proposal Network

[Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks]

Faster R-CNN





Теперь region proposals предсказываются с помощью Region Proposal Network

Время предсказания ~ 0.2 секунды на картинку.

[Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks]

YOLO (You Only Look Ones)

[You Only Look Once: Unified, Real-Time Object Detection]

Осуществляем детекцию без region proposals

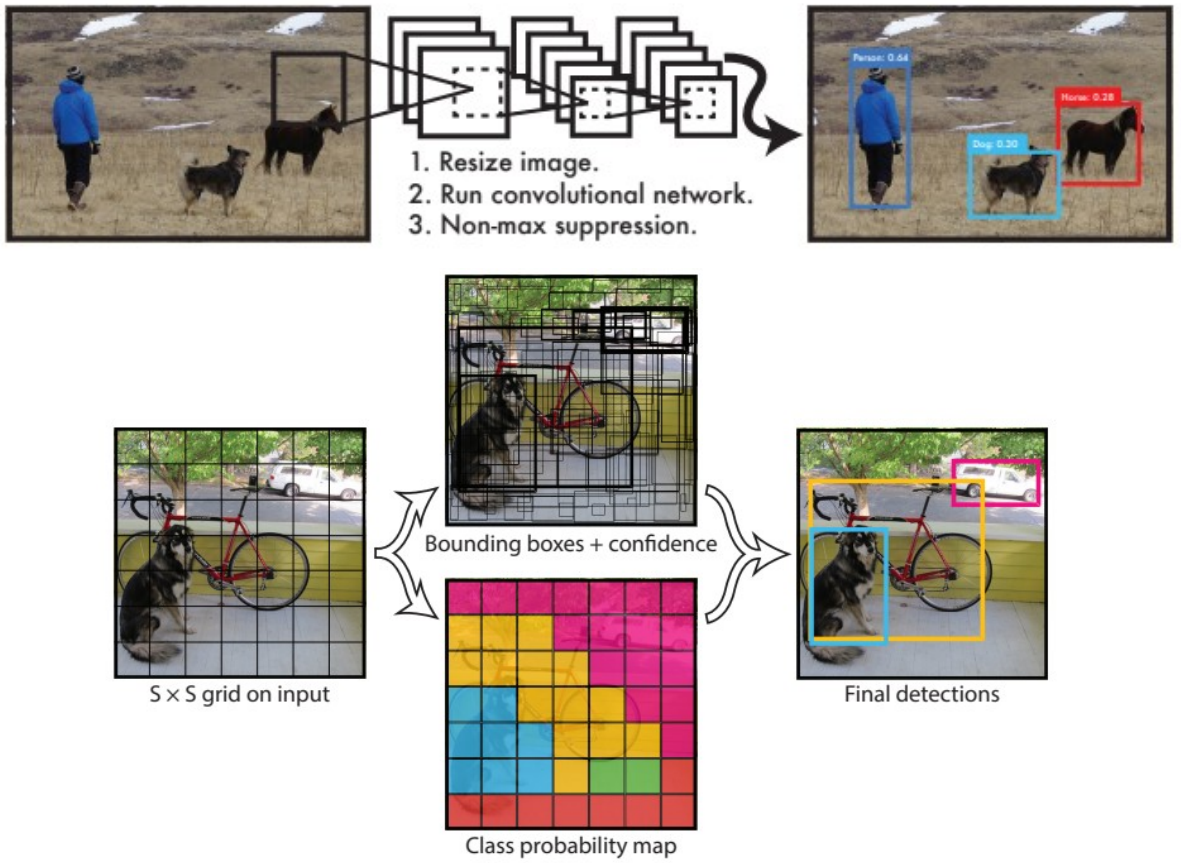
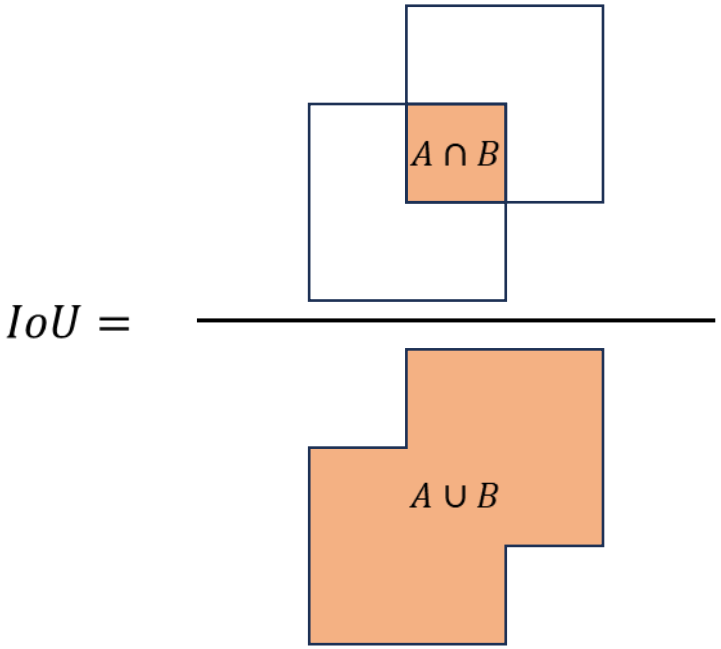


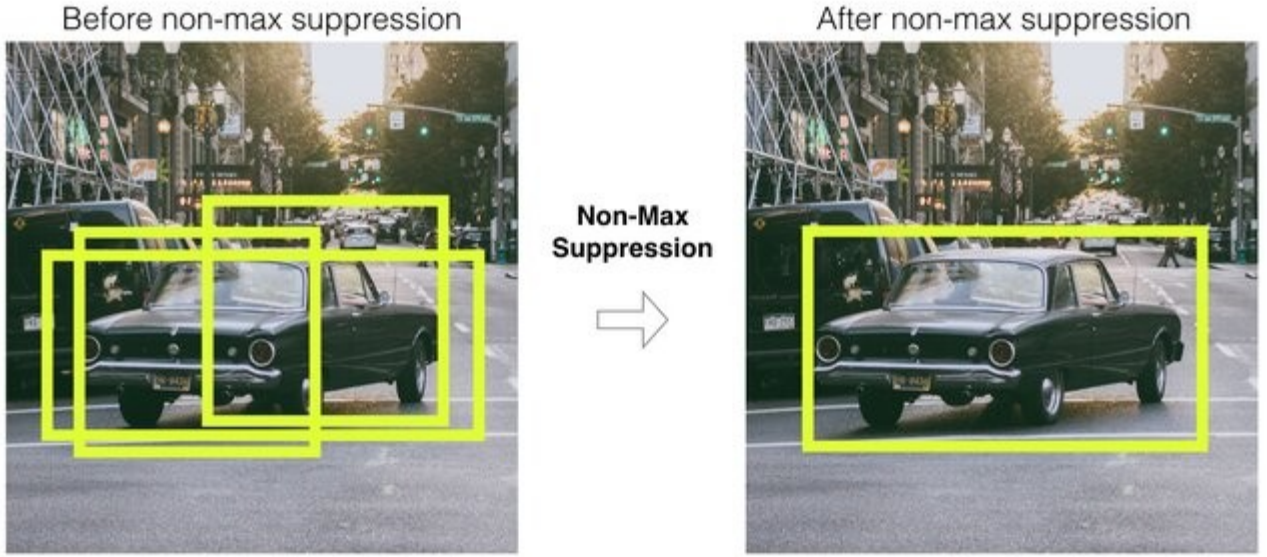
Figure 2: The Model. Our system models detection as a regression problem. It divides the image into an $S \times S$ grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. These predictions are encoded as an $S \times S \times (B * 5 + C)$ tensor.

Non-maximum suppression

$$IoU - Intersection\ over\ Union$$

$$IoU = \frac{\text{Площадь пересечения}}{\text{Площадь объединения}} = \frac{A \cap B}{A \cup B}$$





Non-maximum suppression алгоритм:

Пусть S – массив, элементы которого содержат информацию о каждом боксе (x_1, y_1, x_2, y_2, *confidence*)

1. Выбрать бокс с наибольшим *confidence* и удалить его из S.
2. Найти IoU выбранного бокса со всеми остальными.
3. Удалить боксы с IoU большим порогового значения (часто берут 0.5) из S.
4. Повторять 1-3, пока в S есть боксы

One-stage и two-stage детекторы

Детекторы можно разделить на две большие группы:

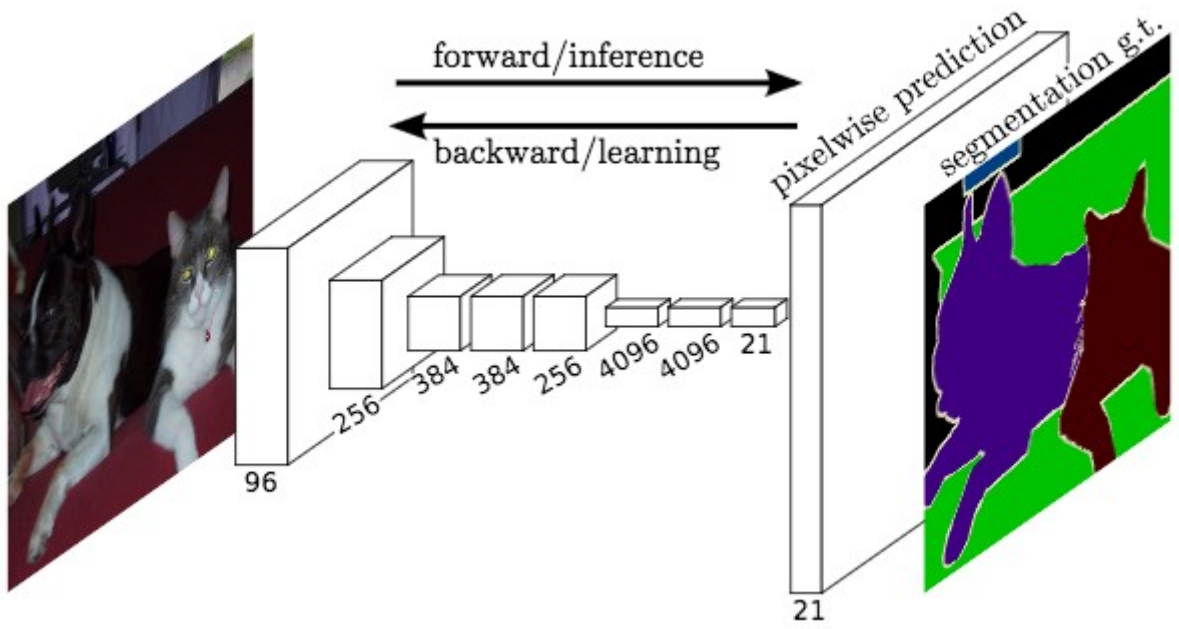
- **Двухстадийные детекторы (two-stage detectors)**
  На первом этапе находят region proposals, которые используются для нахождения объекта на втором этапе.

  Выше качество, Больше время предсказания
  Примеры: семейство R-CNN

- **Одностадийные детекторы (one-stage detectors)**
  Этап region proposals отсутствует.

  Ниже качество, меньше время предсказания
  Примеры: семейство YOLO, SSD, RetinaNet, EfficientDet

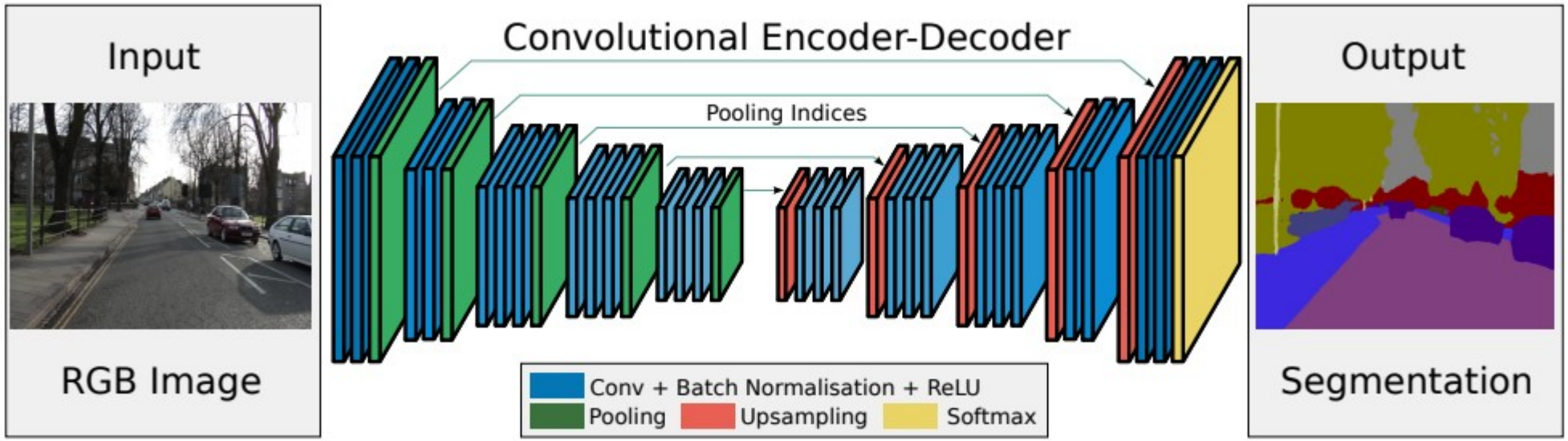
Семантическая сегментация (Semantic Segmentation)



Можем использовать стандартную архитектуру из раздела классификации изображений, только без полносвязного слоя в конце.

Слишком “резкий” upsampling до исходного размера картинки – границы областей получаются не четкими.

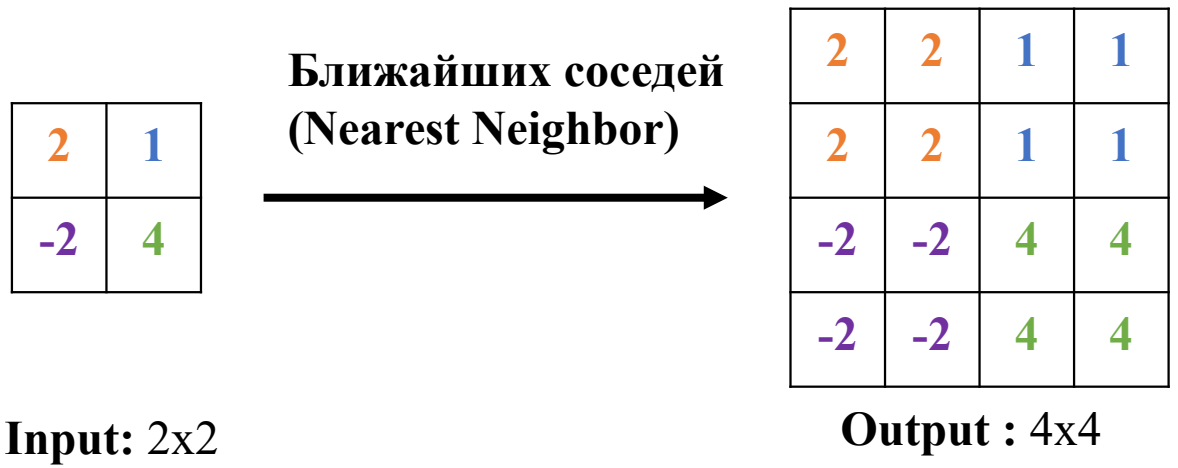
Семантическая сегментация (Semantic Segmentation)

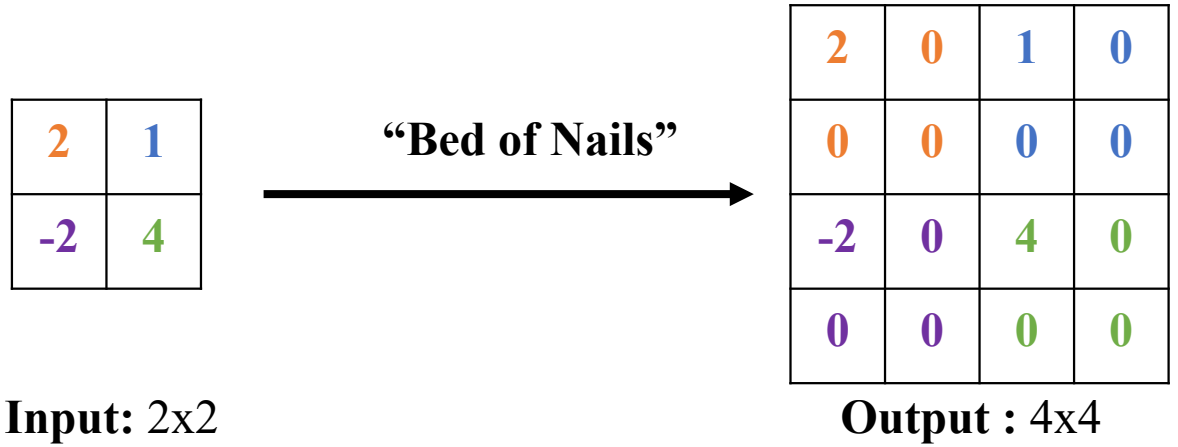


Лучшие результаты показывает так называемая Hourglass (песочные часы) архитектура

SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation

Как можно делать upsampling?





Как можно делать upsampling: Max Unpooling

Индексы максимальных значений после **Max Pooling** сохраняются и используются при **Max Unpooling**

2	-4	2	1
0	1	0	-1
2	1	4	2
5	-3	1	0

Max pool, 2x2 filter →

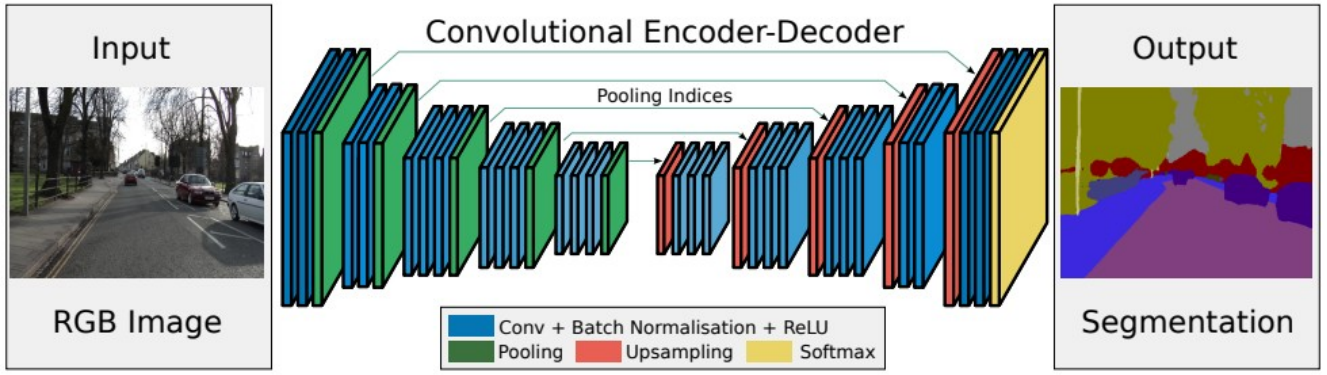
2	2
5	4

→ • • • →

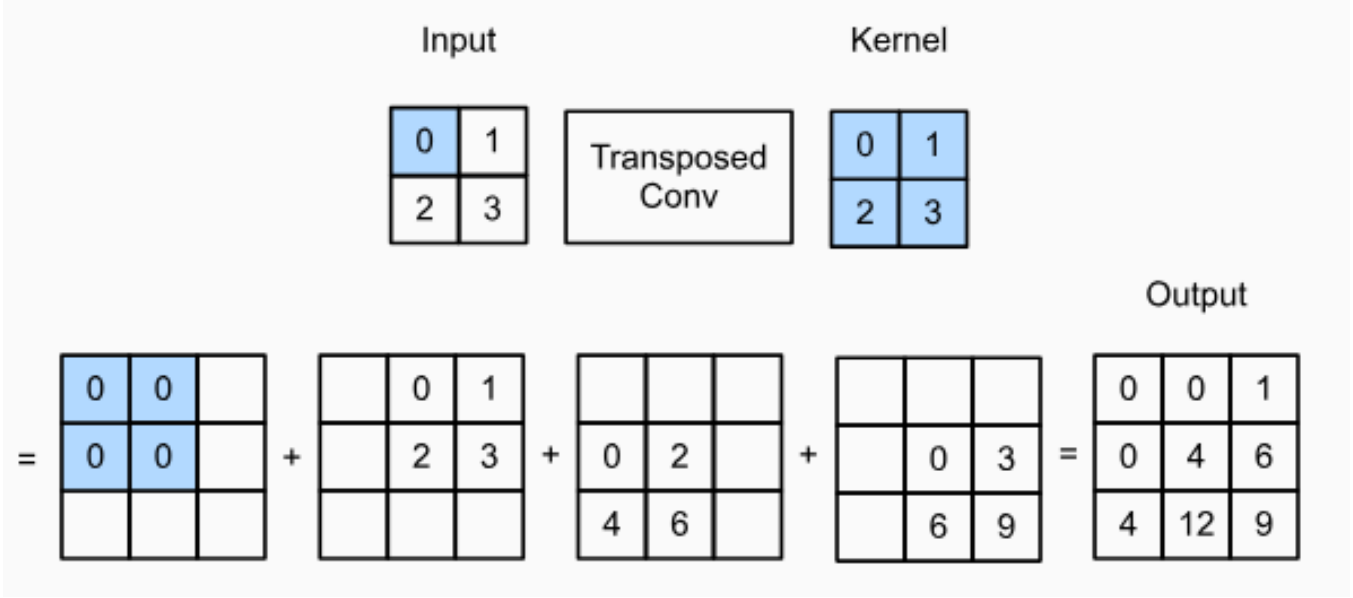
3	-2
4	1

Max Unpooling →

3	0	-2	0
0	0	0	0
0	0	1	0
4	0	0	0

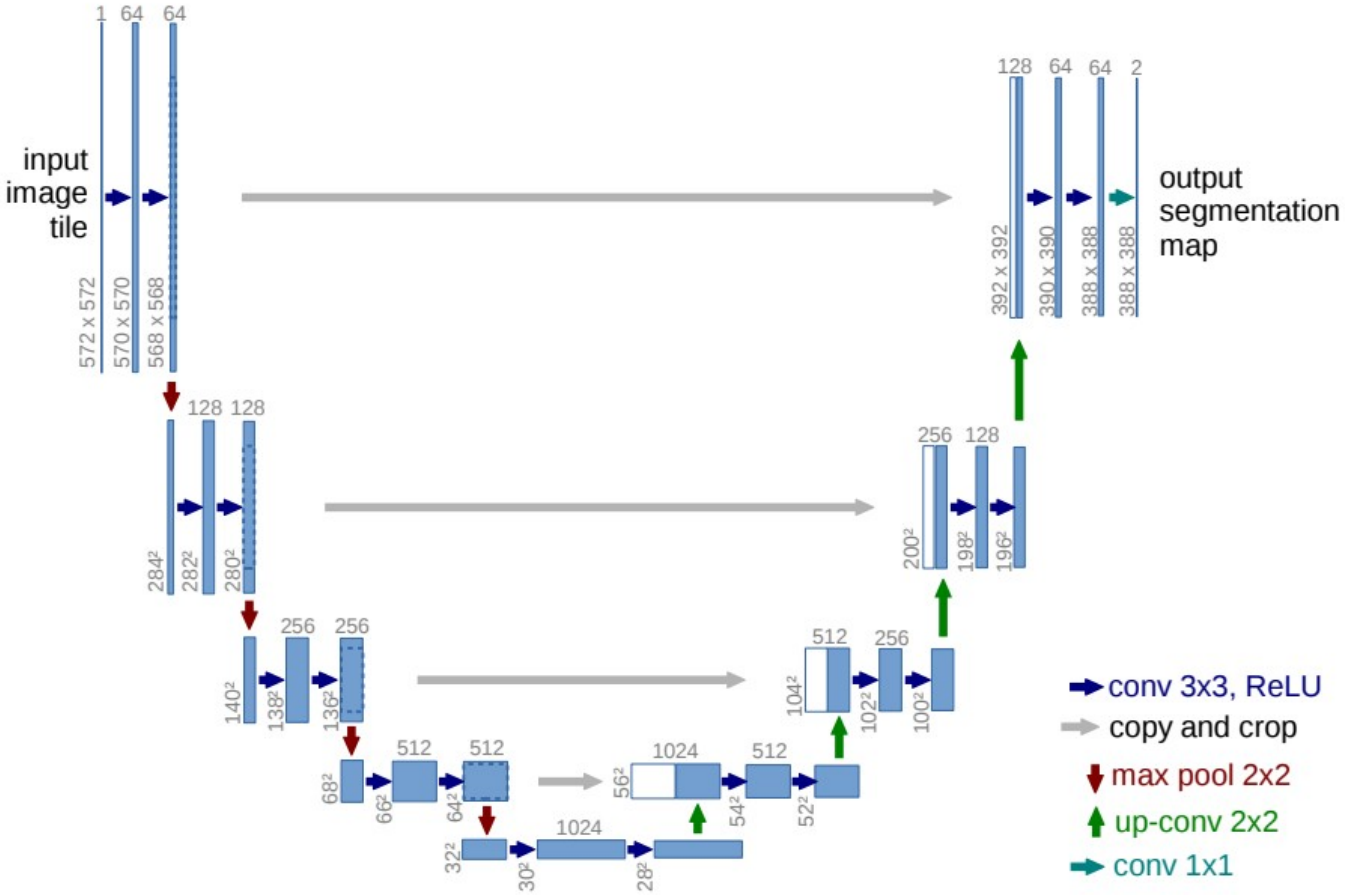


Как можно делать upsampling: Transposed convolution



[Dive into Deep Learning]

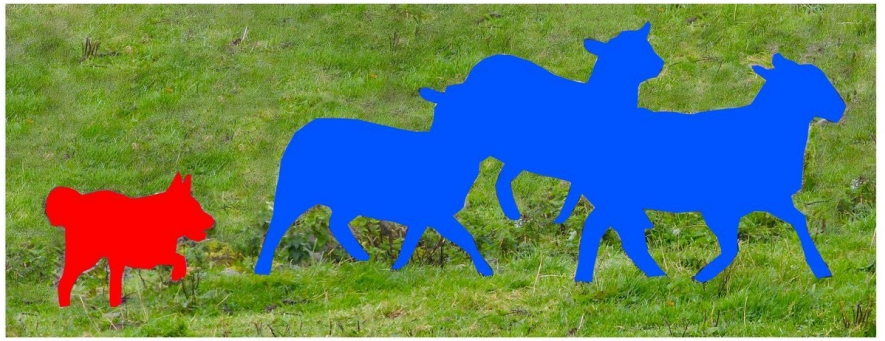
U-net



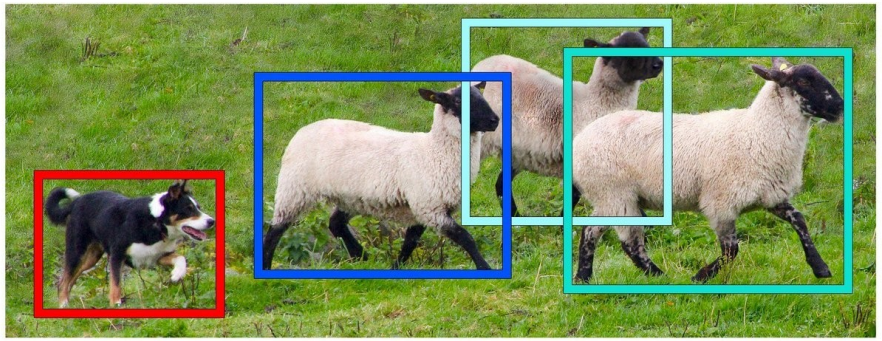
[U-Net: Convolutional Networks for Biomedical Image Segmentation]

Сегментация объектов (Instance Segmentation)

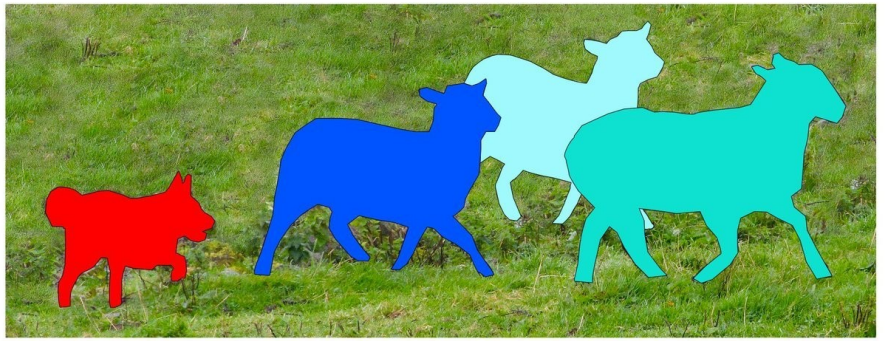
Image Recognition



Semantic Segmentation

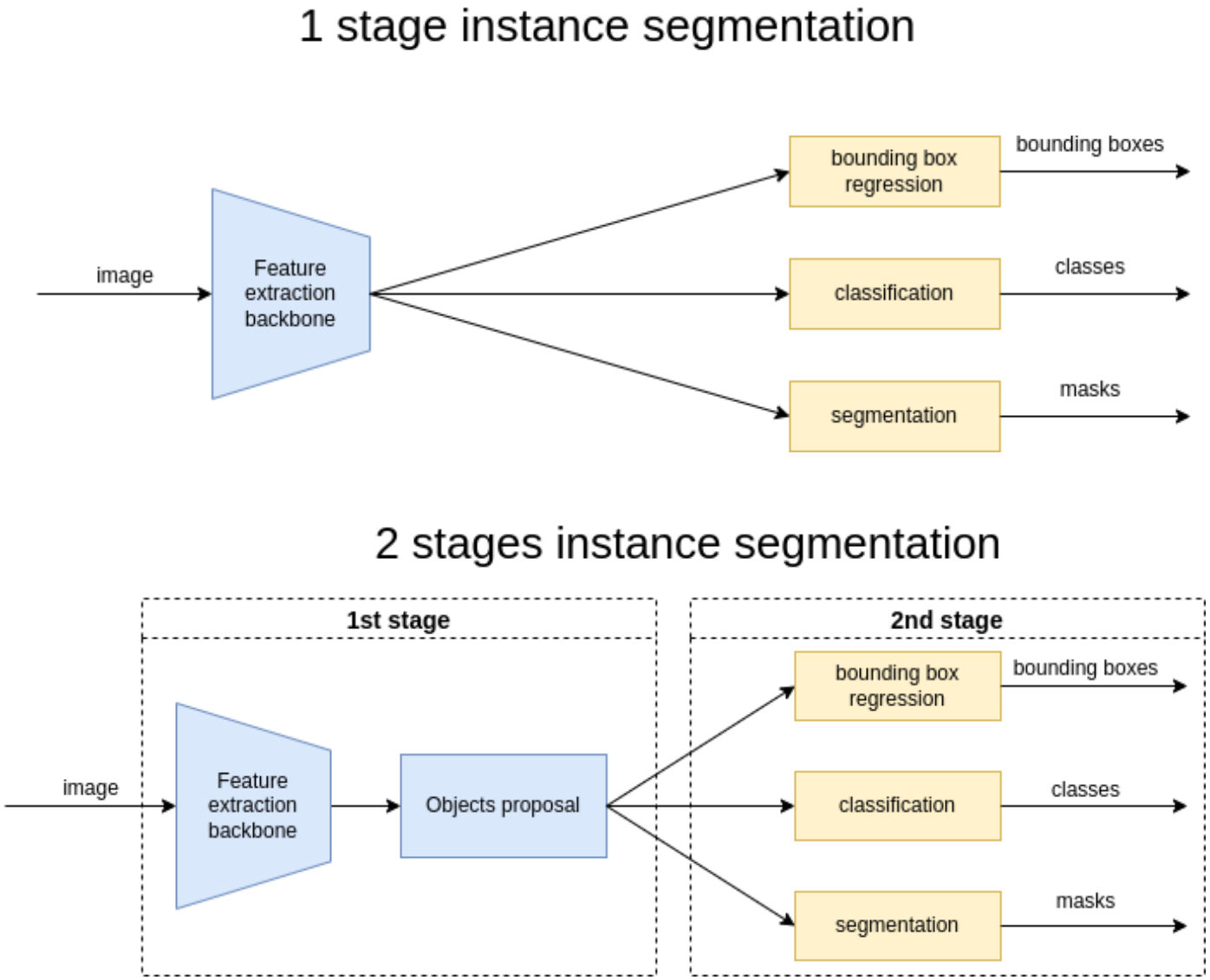


Object Detection

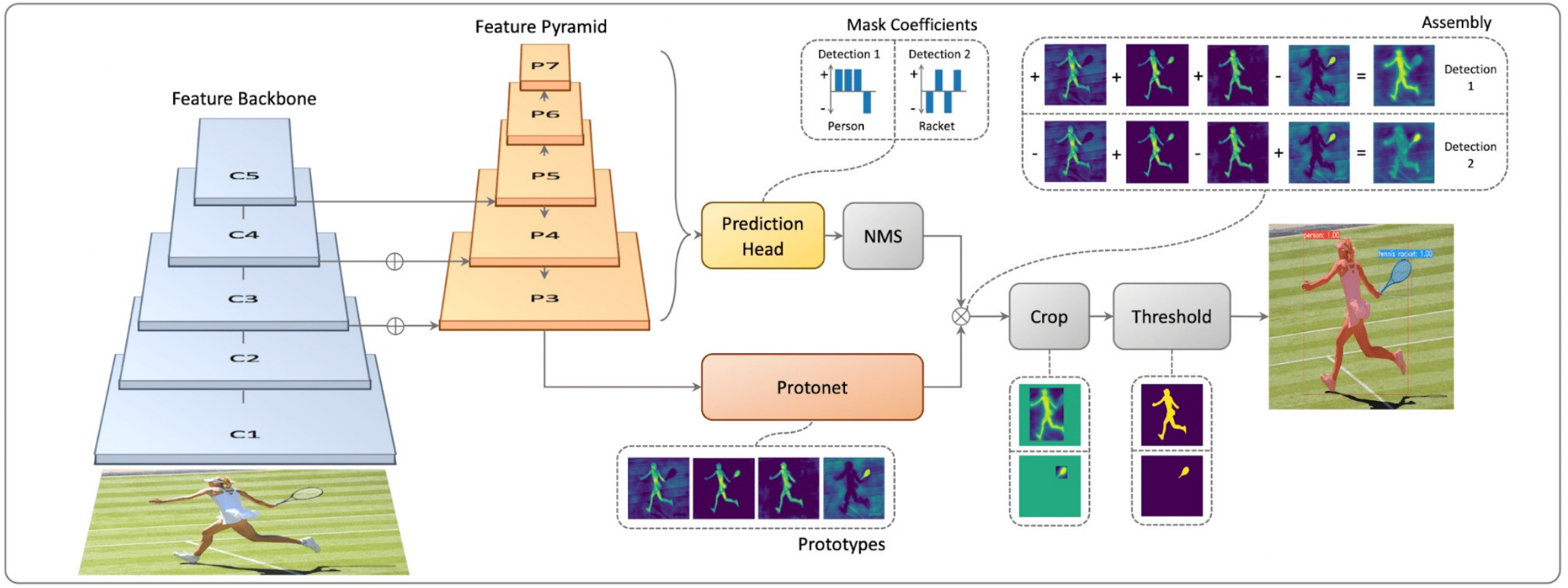


Instance Segmentation

Сегментация объектов (Instance Segmentation)



Сегментация объектов (Instance Segmentation)



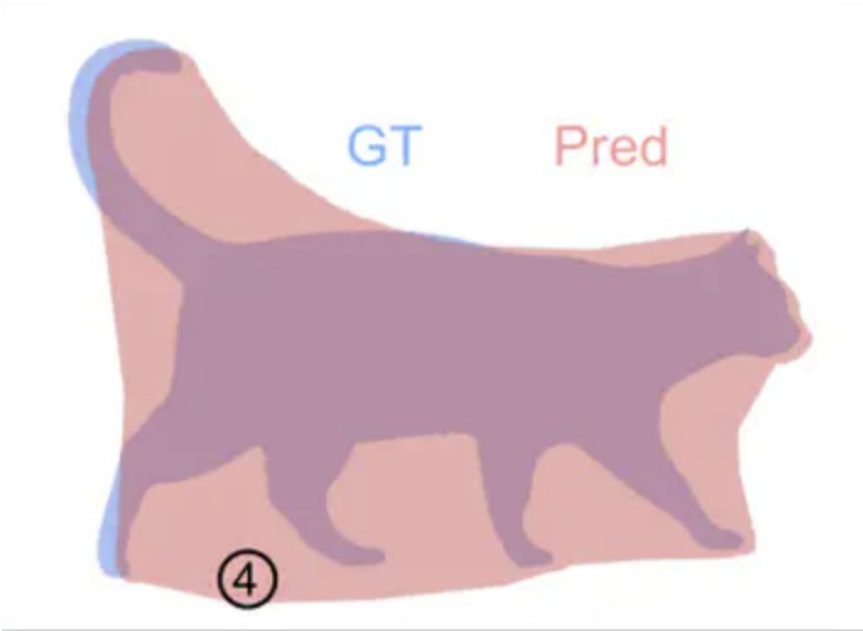
https://arxiv.org/pdf/1904.02689

Метрики в сегментации и детекции

$$IoU = \frac{A \cap B}{A \cup B}$$

← Если работаем с **боксами**
Если работаем с **сегм. масками** →

$$IoU = Jaccard\ Coefficient = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$



$$Dice = 2\frac{A \cap B}{A + B}$$

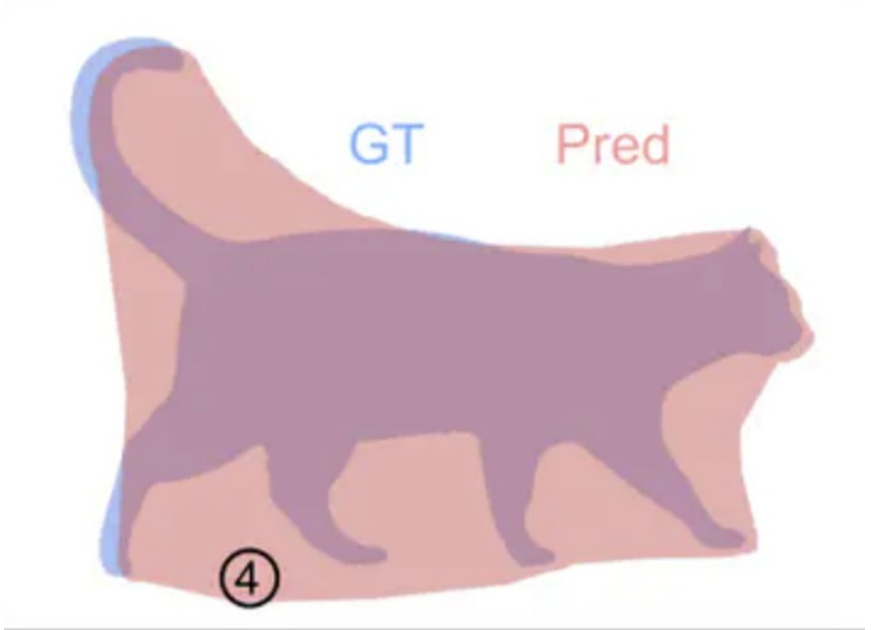
$$Dice\ Coefficient = 2\frac{|A \cap B|}{|A| + |B|}$$

Метрики в сегментации и детекции

$$IoU = \frac{A \cap B}{A \cup B}$$

← Если работаем с **боксами**
Если работаем с **сегм. масками** →

$$IoU = Jaccard\ Coefficient = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$



$$Dice = 2\frac{A \cap B}{A + B}$$

$$Dice\ Coefficient = 2\frac{|A \cap B|}{|A| + |B|}$$

В задаче **semantic segmentation** IoU вычисляется для каждого класса, а затем находится среднее значение — mean IoU (**mIoU** метрика)

Метрики в сегментации и детекции: mean Average Precision (mAP)

mAP показывает, насколько модель хорошо локализует объекты на изображении и правильно предсказывает классы (подробнее на практике)

Используется, например, в **object detection**, **instance segmentation**