

Глубинное обучение

Лекция 10
Сегментация изображений

Михаил Гущин

mhushchyn@hse.ru

НИУ ВШЭ, 2024



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

В предыдущей серии

Классификация изображений



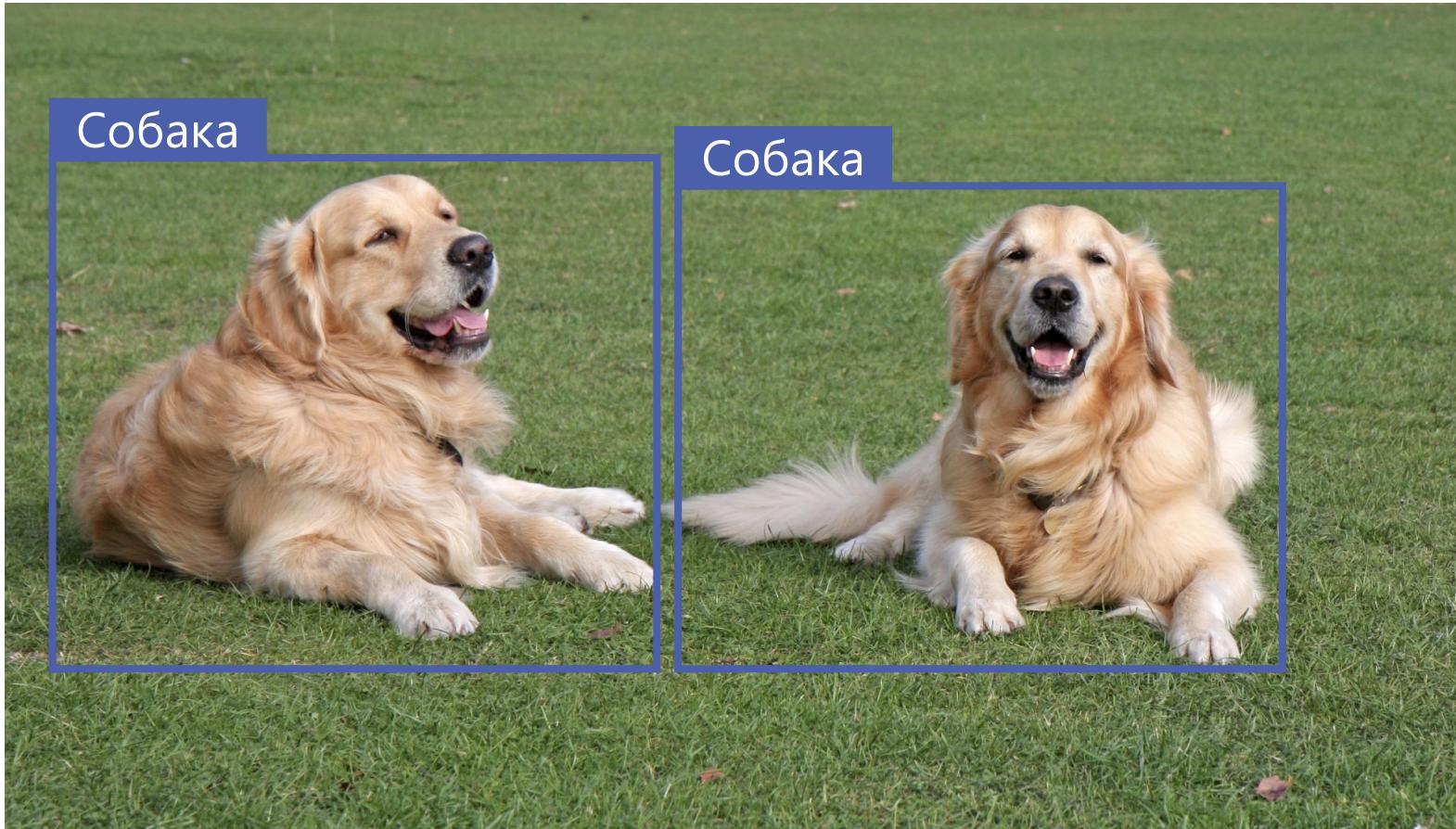
«Кот»

Классификация и локализация

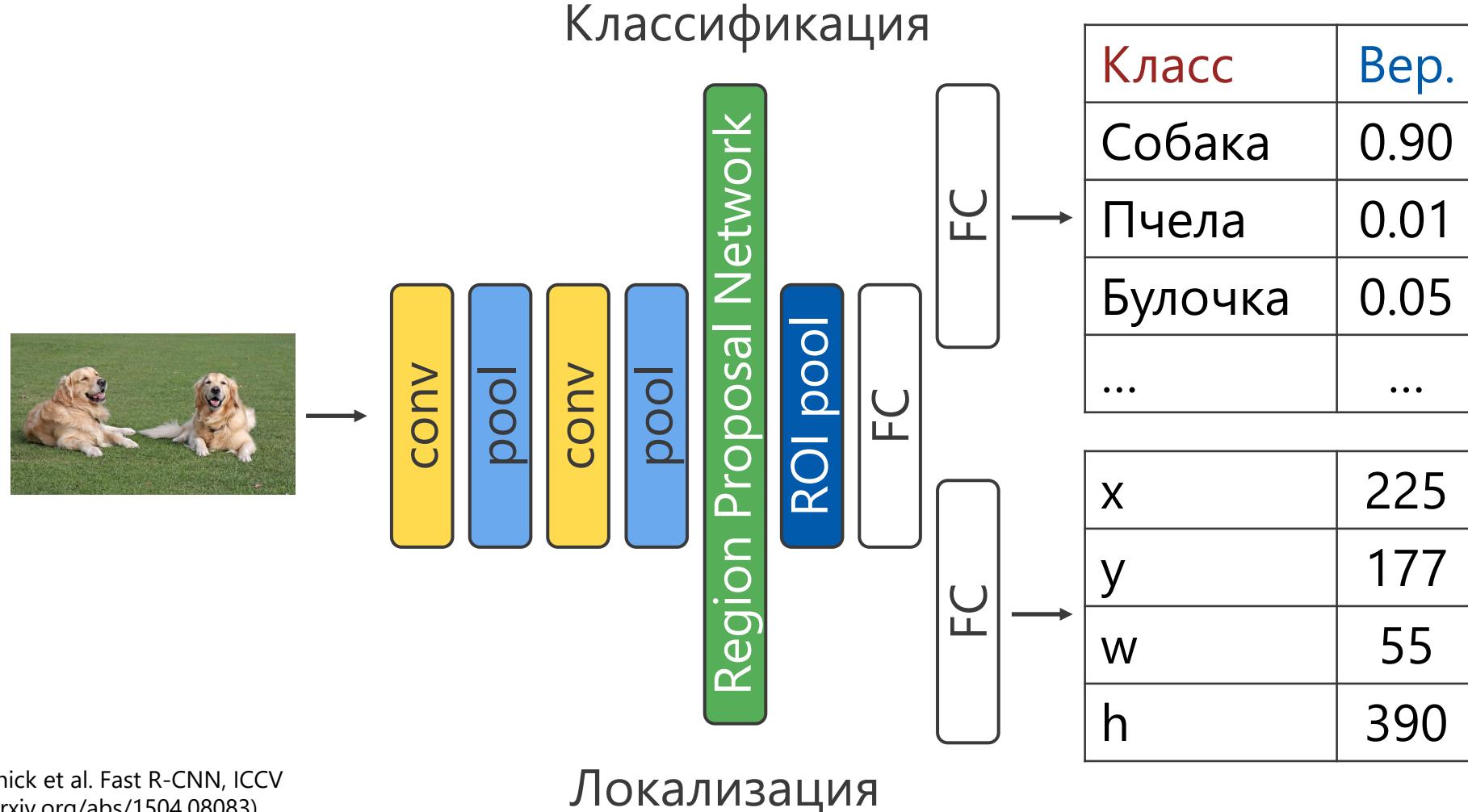


«Кот»

Детектирование объектов

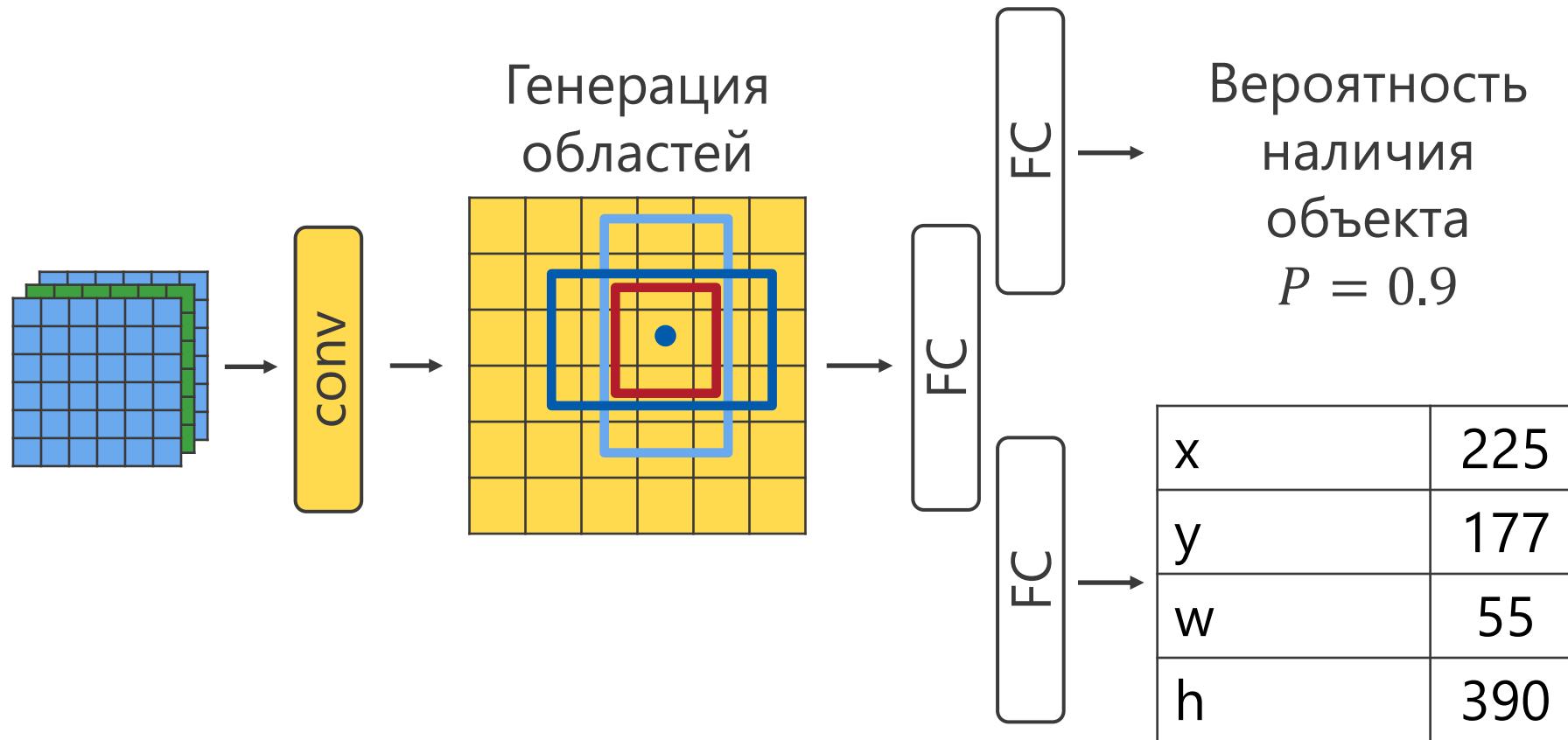


Faster R-CNN



Подробнее: R. Girshick et al. Fast R-CNN, ICCV
2015 (URL: <https://arxiv.org/abs/1504.08083>)

Region proposal network



* Обучаем эту сеть отдельно

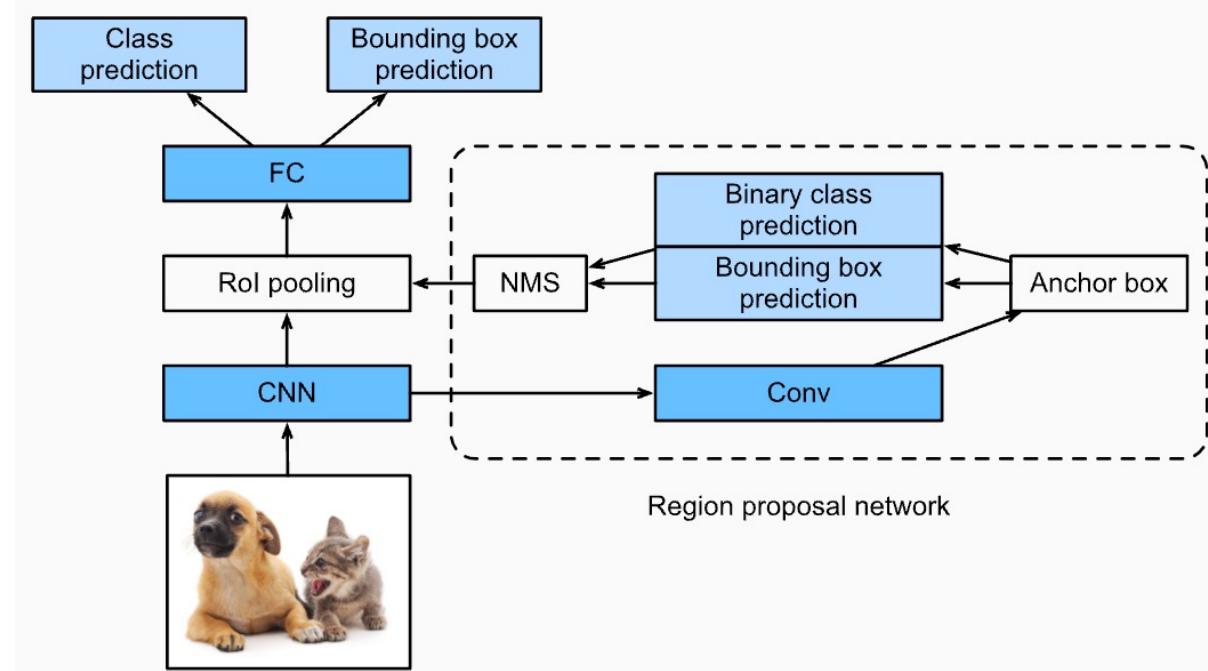
Region proposal network

- ▶ Простая генерация областей
- ▶ Оставляем только области, где предсказан объект
- ▶ Применяем non-maximum suppression для отбора лучших областей
- ▶ Выводим предсказанные локализации (x, y, w, h) в лучших областях
- ▶ Локализации (x, y, w, h) используем как ROI в Faster R-CNN

Вопрос

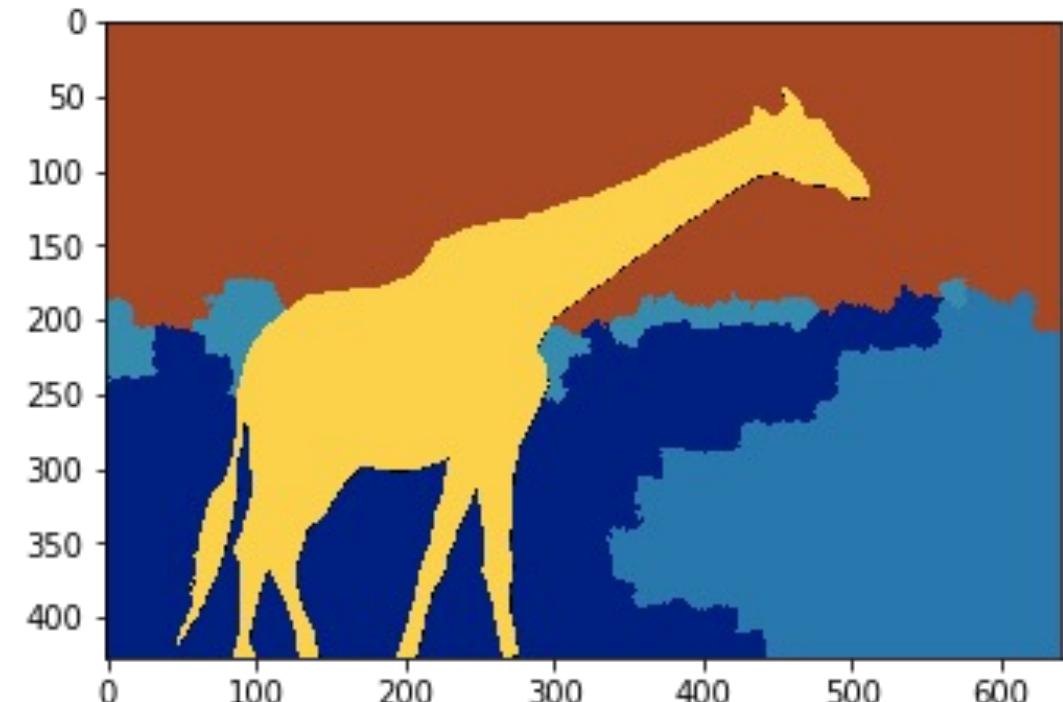
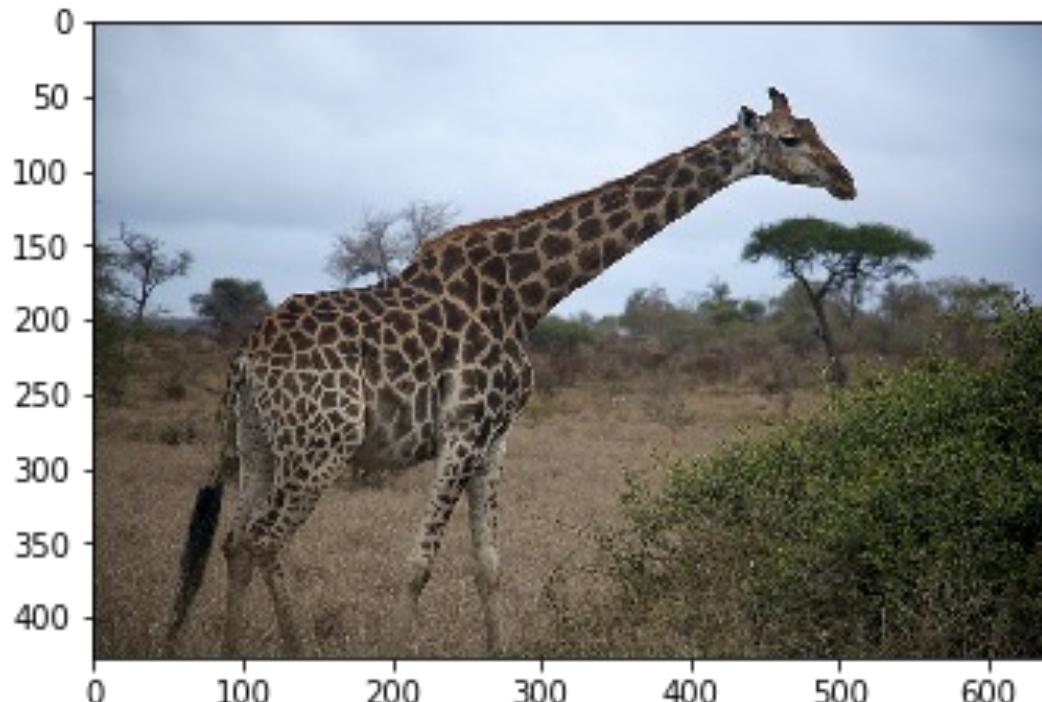
- ▶ Зачем мы дважды предсказываем прямоугольники?
- ▶ Можно это делать либо в region proposal network, либо на последнем слое faster r-cnn?

Ответ: прямоугольники нам нужны, чтобы убрать перекрывающиеся прямоугольники. Это уменьшает общее число ROI для дальнейшего анализа.



Семантическая сегментация

Семантическая сегментация



mxnet.apache.org

Семантическая сегментация

Входное изображение



Метка класса
для каждого пикселя

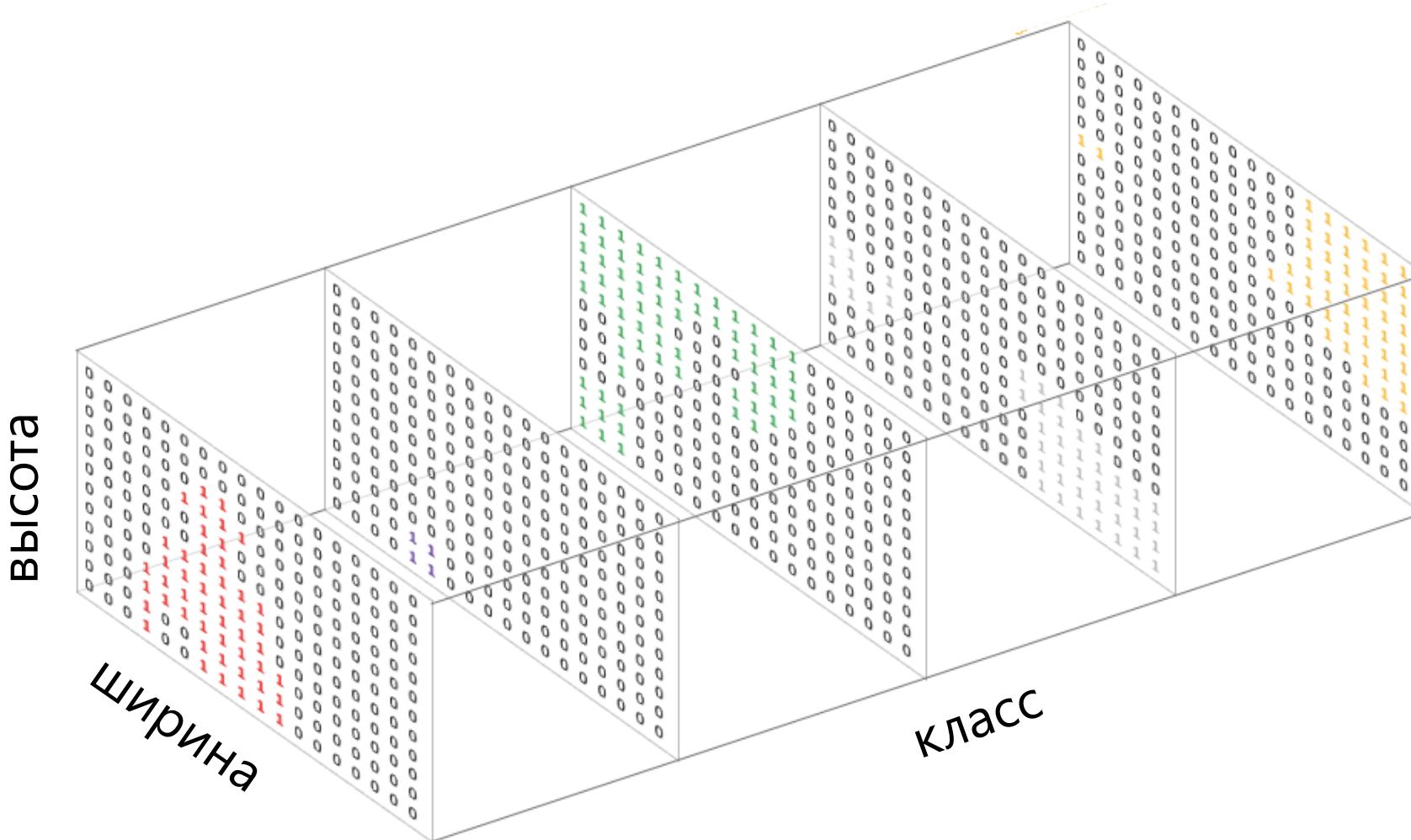
| | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 | |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 |
| 4 | 4 | 3 | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 5 | 5 | 5 |
| 4 | 4 | 3 | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 5 | 5 |
| 4 | 4 | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | 3 | 3 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | 3 | 3 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 |

jeremyjordan.me

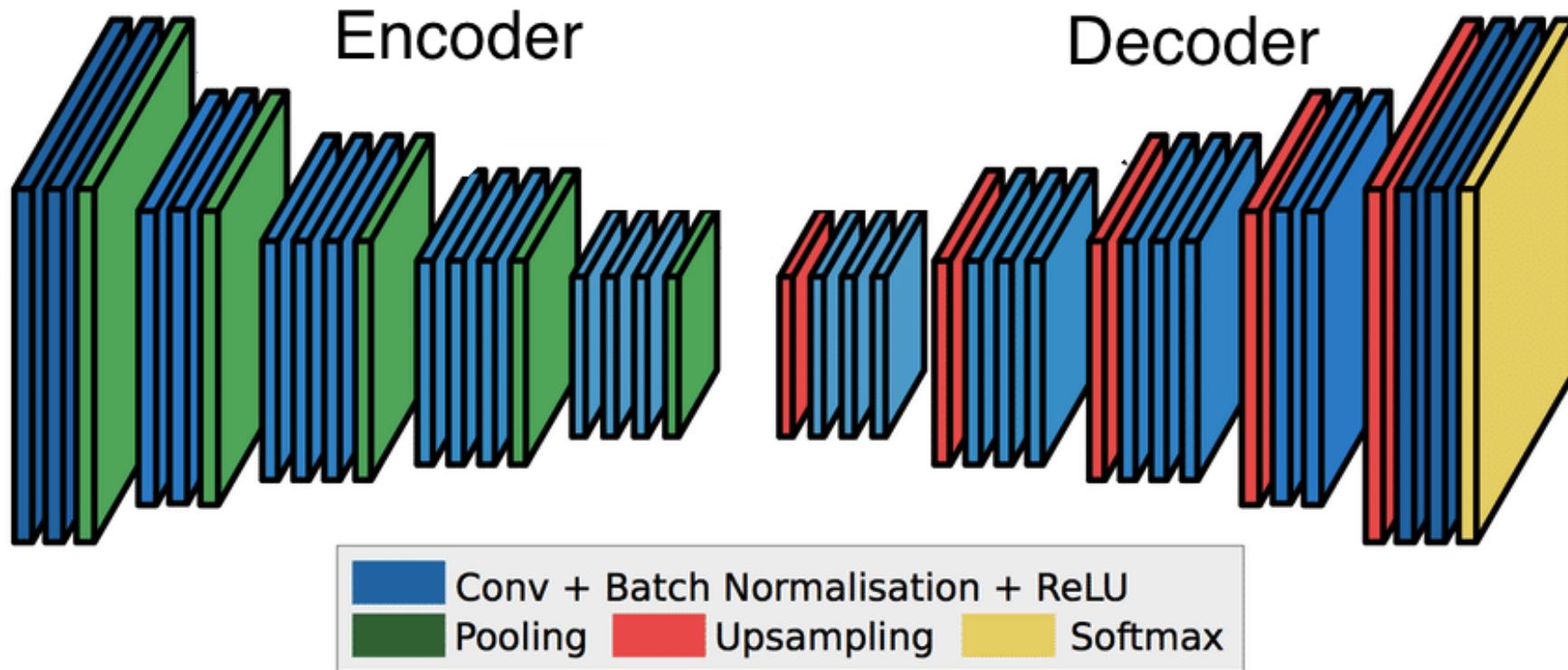
Семантическая сегментация

- ▶ При классификации изображений предсказываем одну метку класса для всего изображения
- ▶ При сегментации предсказываем метку класса для каждого пикселя изображения

Семантическая сегментация

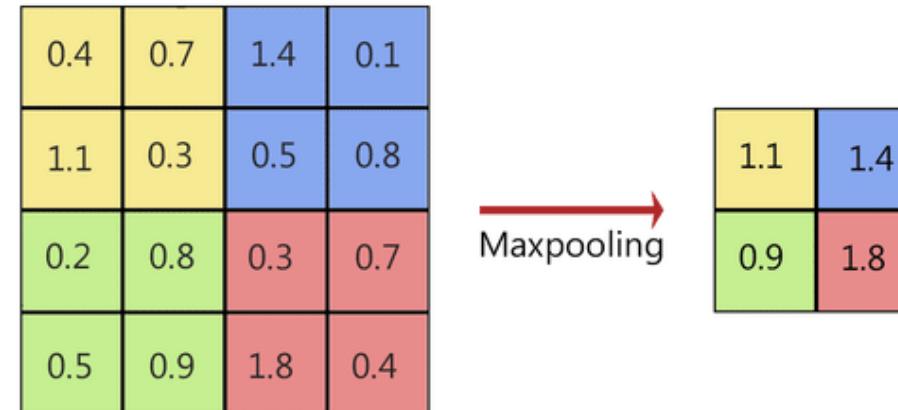


Типичная архитектура сети



semanticscholar.org

Upsampling



Выходное изображение

Входное изображение



$(3 \times H \times W)$

Метка класса
для каждого пикселя

| | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 | 5 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 | 5 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 | 5 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 | 5 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5 | 5 | 5 | 5 | 5 | 5 |
| 4 | 4 | 3 | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 5 | 5 |
| 4 | 4 | 3 | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 5 | 5 |
| 4 | 4 | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | 3 | 3 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | 3 | 3 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 |

$(C \times H \times W)$

C — Число классов

Функция потерь для обучения

Для одного изображения (categorical cross-entropy):

$$L = \sum_{i=1}^N \sum_{c=1}^C [y_i = c] \log p_{ic}$$

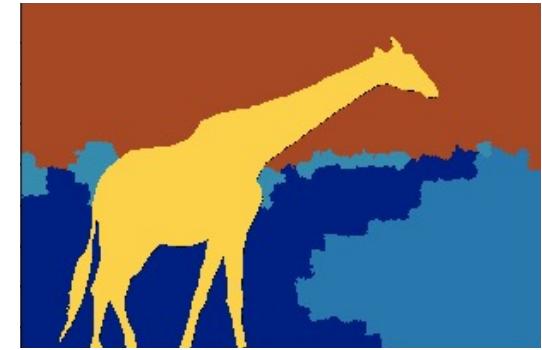
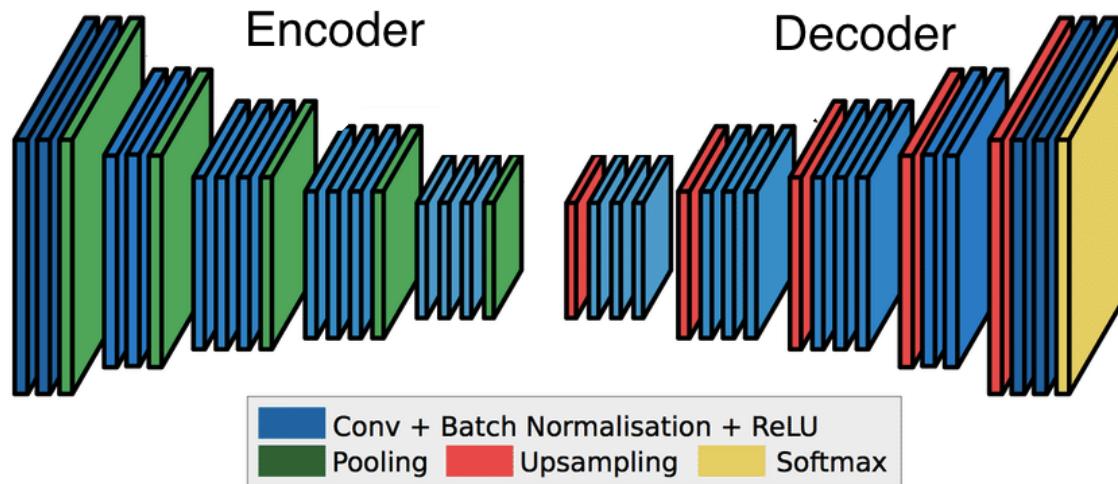
p_{ic} — вероятность c -го класса в i -ом пикселе;

y_i — истинная метка в i -ом пикселе;

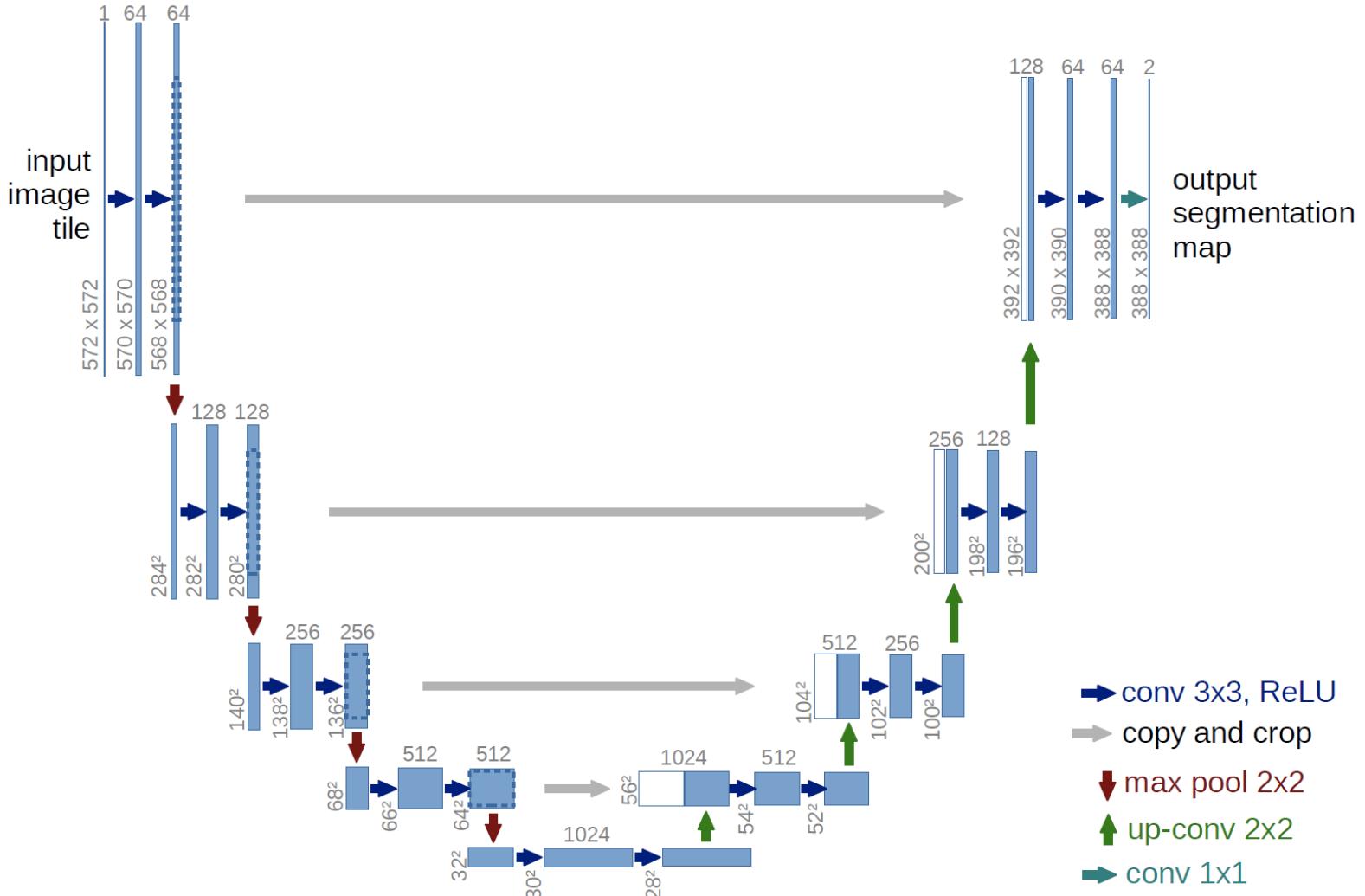
N — общее количество пикселей в изображении;

C — общее число классов.

Семантическая сегментация

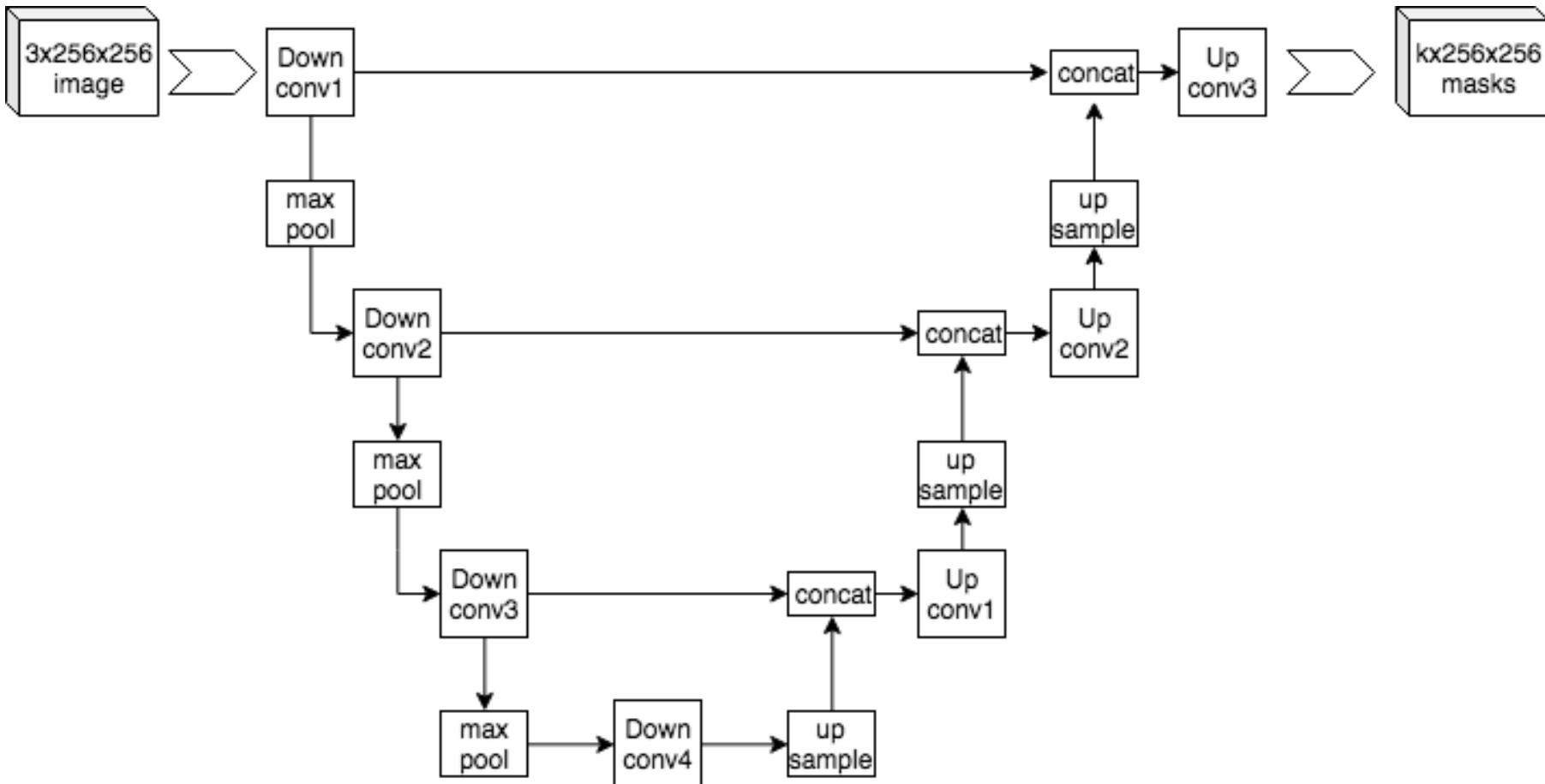


U-Net



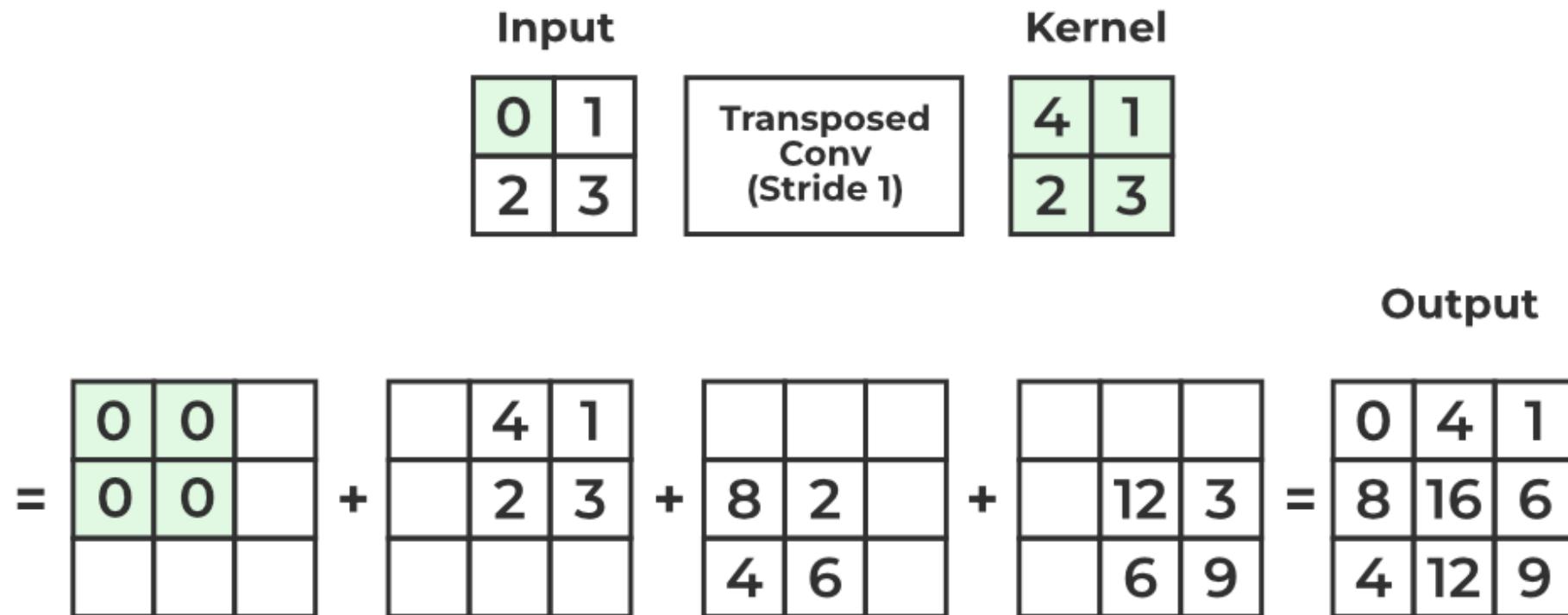
Подробнее: O. Ronneberger et al. U-Net: Convolutional Networks for Biomedical Image Segmentation, MICCAI 2015 (URL: <https://arxiv.org/abs/1505.04597>)

U-Net



Подробнее: By Mehrdad Yazdani - Own work, CC BY-SA 4.0
(URL: <https://commons.wikimedia.org/w/index.php?curid=81055729>)

ConvTranspose2d (upconv)



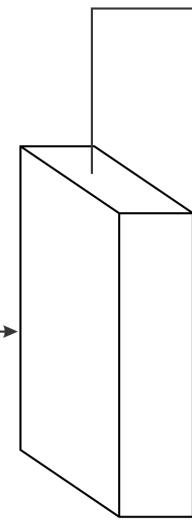
Pyramid Scene Parsing Network (PSPNet)

Pyramid Scene Parsing Network (PSPNet)

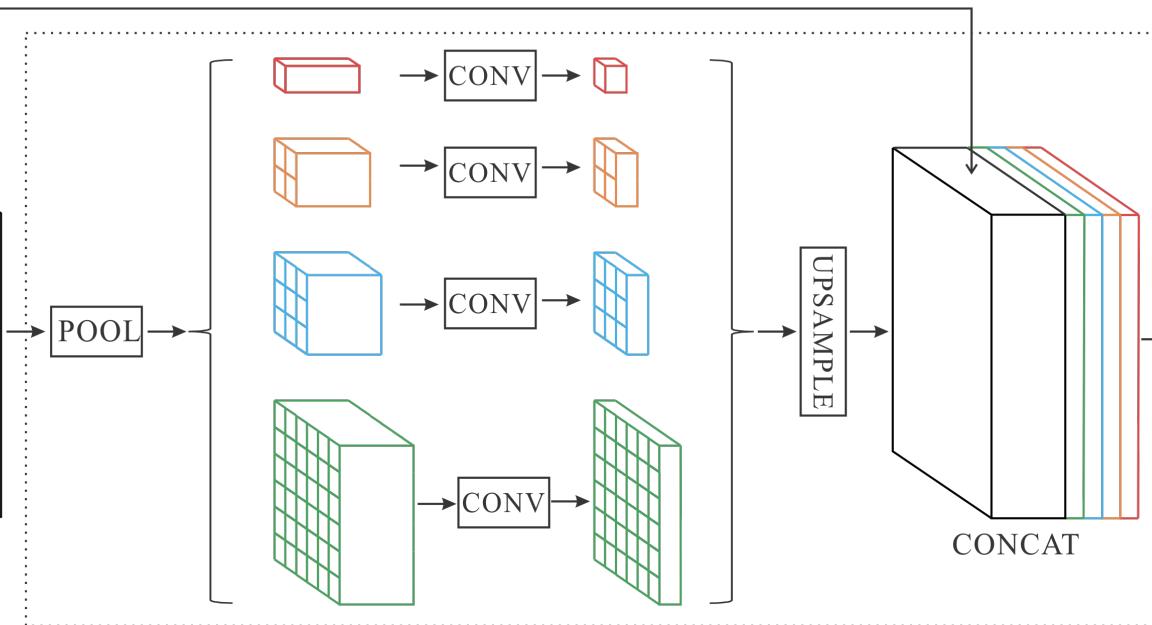
Для сегментации объектов разного размера



(a) Input Image



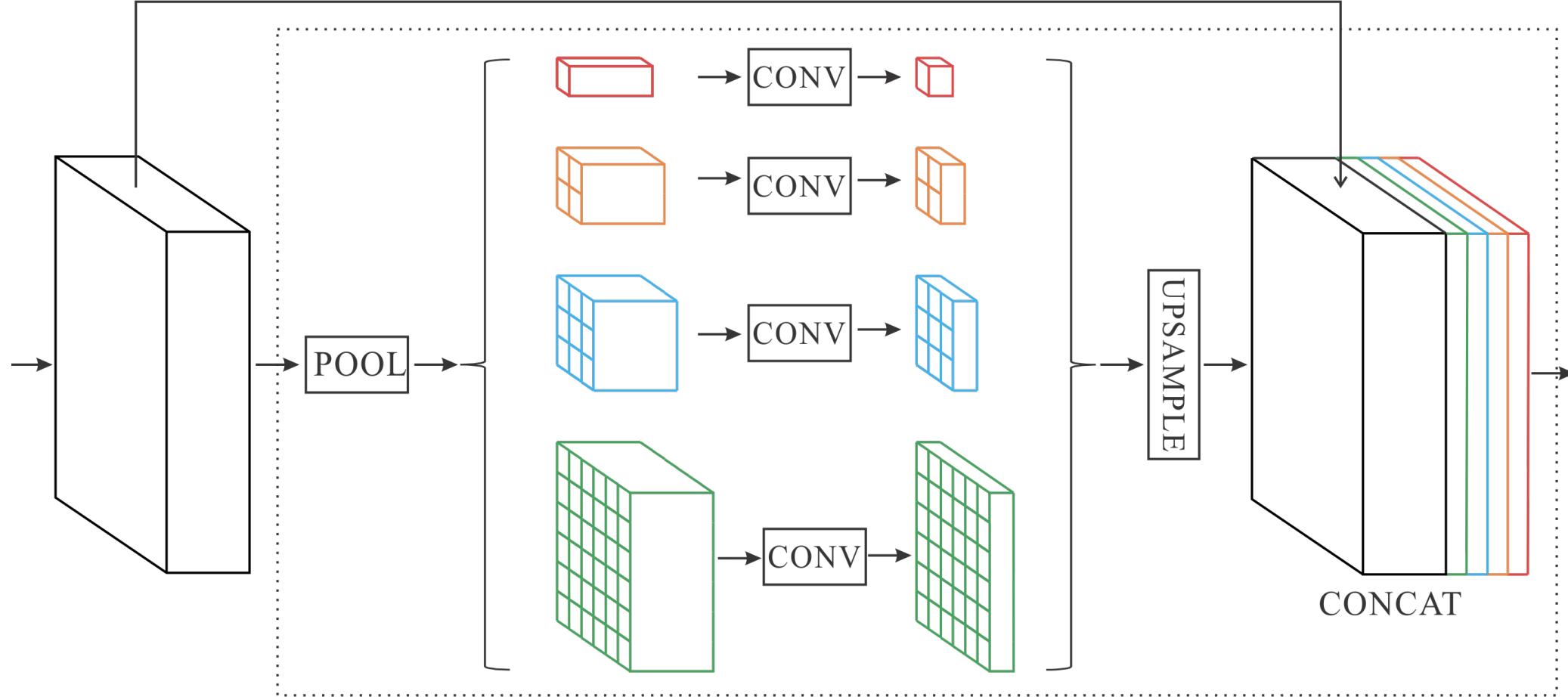
(b) Feature Map



(d) Final Prediction

Подробнее: H. Zhao et al. Pyramid Scene Parsing Network, CVPR 2017 (URL: <https://arxiv.org/abs/1612.01105v2>)

Pyramid Scene Parsing Network (PSPNet)

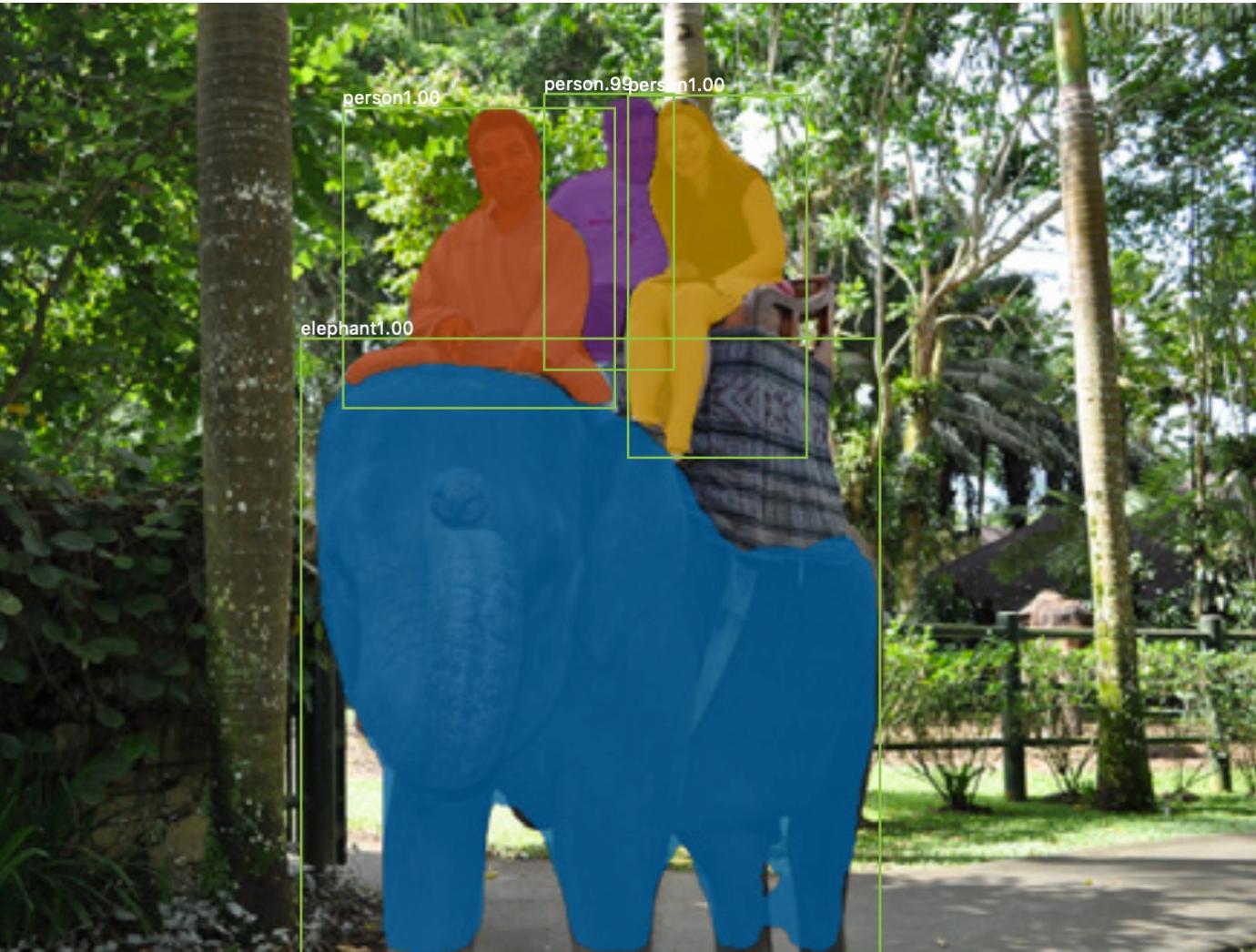


Другие алгоритмы

- ▶ DeepLab
- ▶ RefineNet
- ▶ LinkNet
- ▶ FCN
- ▶ LRASPP

Объектная сегментация

Объектная сегментация

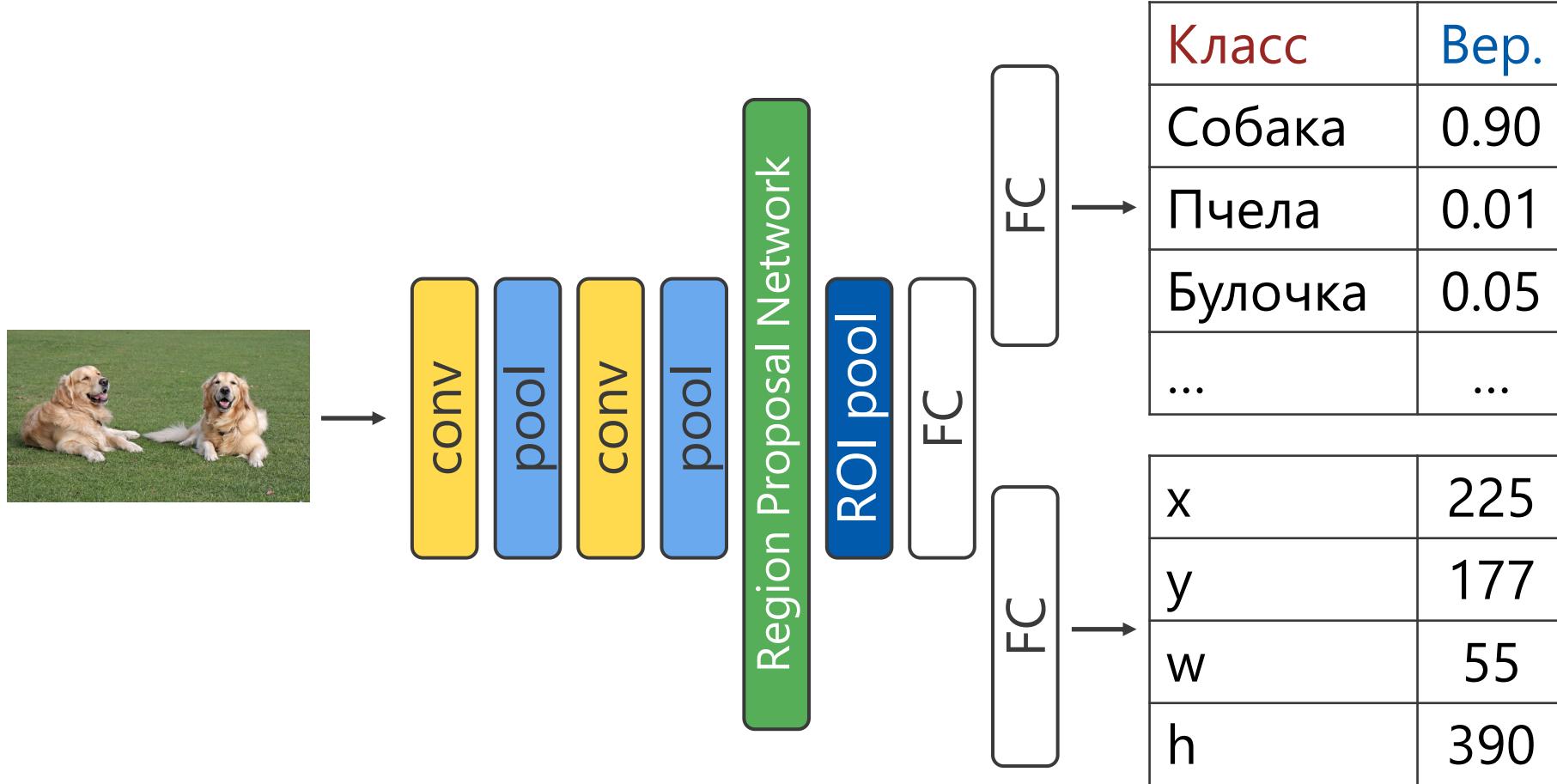


Подробнее: K. He et al. Mask R-CNN, 2017 (URL: <https://arxiv.org/abs/1703.06870>)

Объектная сегментация

- ▶ Найти прямоугольник с объектом
- ▶ Найти все пиксели объекта внутри прямоугольника

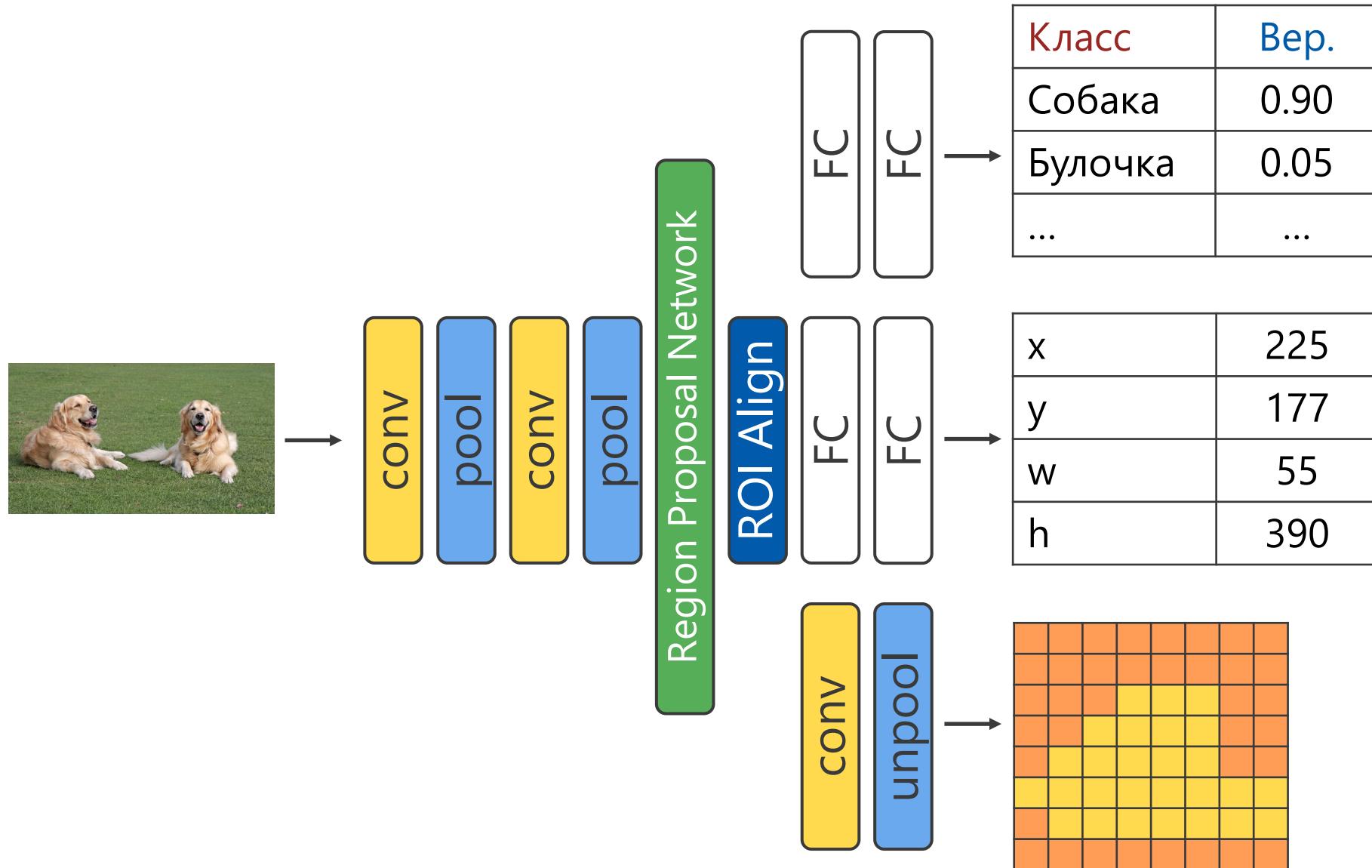
Faster R-CNN



Идея

- ▶ Возьмем Faster R-CNN за основу
- ▶ Для каждого ROI попросим найти пиксели объекта

Mask R-CNN



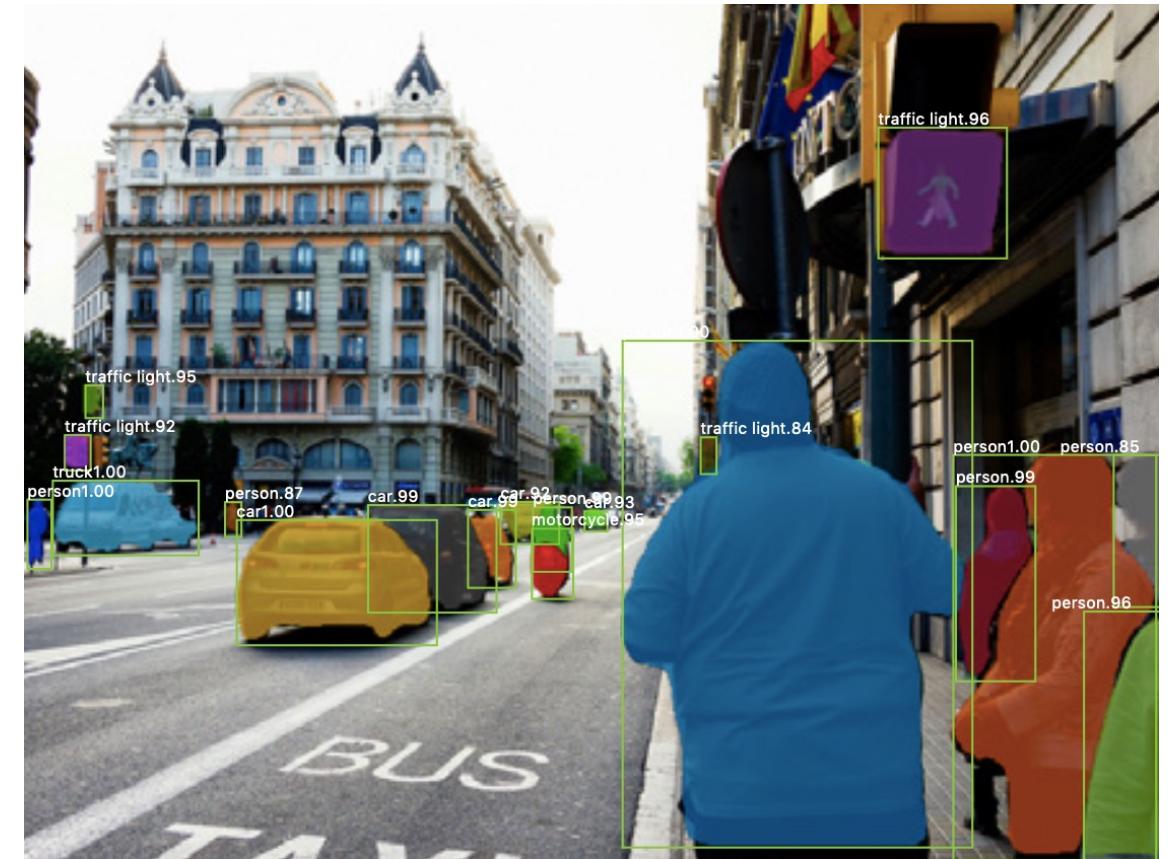
ФУНКЦИИ ПОТЕРЬ

Многозадачная функция потерь L :

$$L = \mathbf{L}_{\text{классификация}} + \alpha \mathbf{L}_{\text{локализация}} + \beta \mathbf{L}_{\text{маска}} \rightarrow \min$$

- ▶ $\mathbf{L}_{\text{классификация}}$ — функция кросс-энтропии для классификации изображения;
- ▶ $\mathbf{L}_{\text{локализация}}$ — MSE функция потерь для параметров прямоугольника;
- ▶ $\mathbf{L}_{\text{маска}}$ — категориальная кросс-энтропия для сегментации.

Примеры



Заключение

- ▶ Классификация изображений
- ▶ Классификация с локализацией объекта
- ▶ Детектирование объектов
- ▶ Семантическая сегментация
- ▶ Объектная сегментация