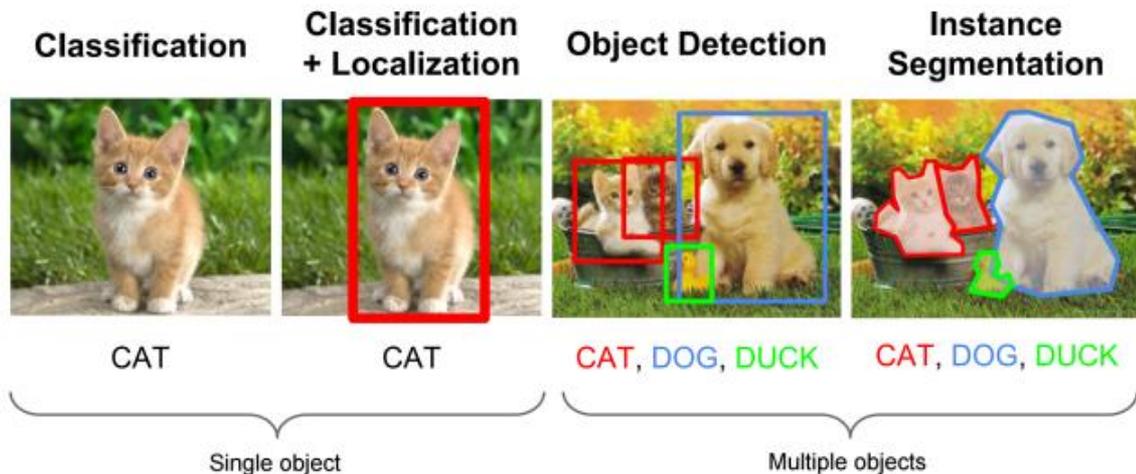


Нейронные сети для задач детекции и сегментации

Задача детекции



Различия с задачей классификации изображений:

Одно изображение может содержать объекты нескольких классов или может не содержать никаких.

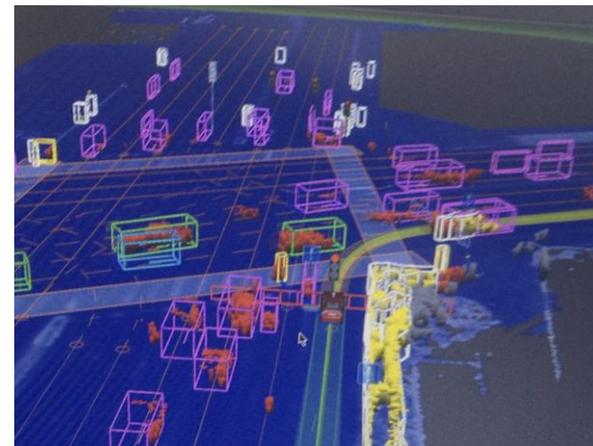
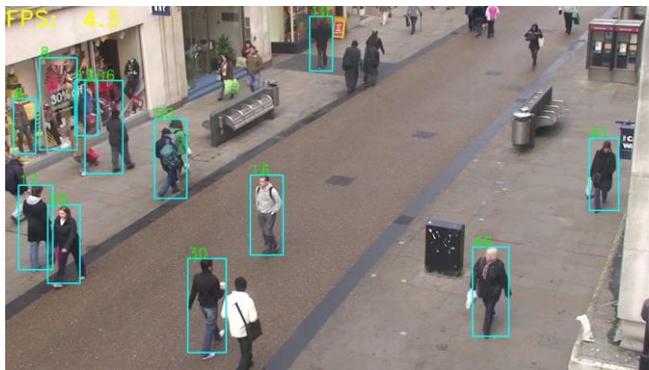
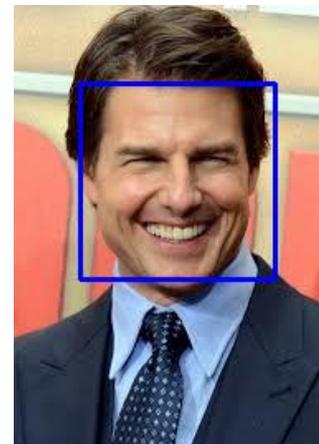
Одно изображение может содержать несколько объектов одного класса.

Нам нужно локализовать объект на изображении, предсказывая его Bounding Box (bbox).

Приложения

- 1) Распознавание лиц
- 2) Поиск людей на фото
- 3) Самостоятельные автомобили / роботы / беспилотники:

- Обнаружение автомобиля
- Обнаружение пешеходов (автоматическое наблюдение)

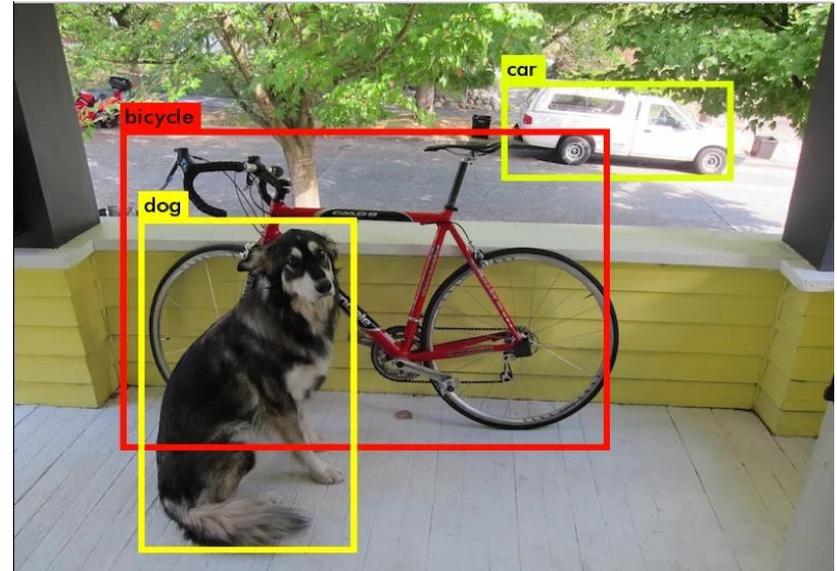


Обнаружение объектов: Постановка задачи

Входные данные: Изображение

Выход: “ограничительные рамки с метками классов”_Каждая рамка:

- Верхний левый угол: (x_{tl}, y_{tl})
 - Правый нижний угол: (x_{br}, y_{br})
 - Метка класса
 - Достоверность (вероятность этого класса)
- ⇒ 6 чисел для каждой рамки



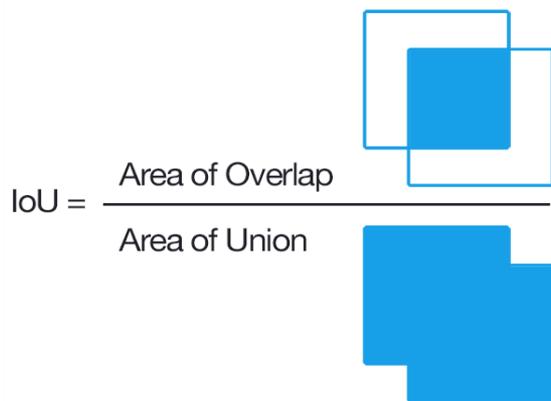
Трудности

- 1) Как локализовать объекты на изображении?
Как предсказать их ограничивающие прямоугольники?
- 2) Объекты могут быть ближе или ближе к камере => ограничивающие прямоугольники имеют совершенно разные размеры.
- 3) Размер изображения играет важную роль => простое изменение размера рамки не будет работать во многих случаях.
- 4) Перекрывающиеся объекты
- 5) Какие метрики точности использовать?



Метрики

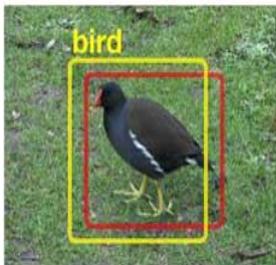
1) Как проверить правильность локализации объекта? - Пересечение областей (IoU).



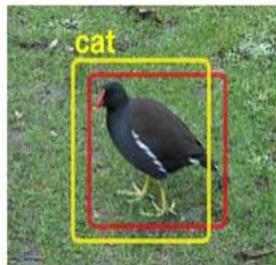
Если $\text{IoU} >$ порог (обычно 0,5), то это True Positive, иначе это False Positive.
Если для какого-либо присутствующего на изображении объекта нет прогноза, это False Negative.

Positives & Negatives

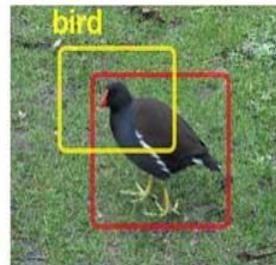
TP



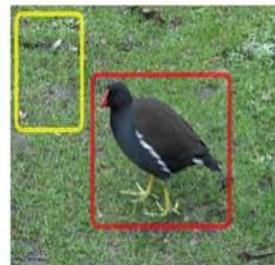
FP



wrong class

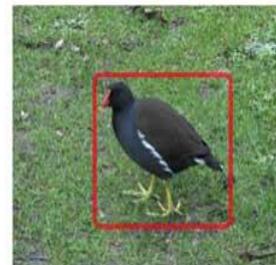


IOU < 0.5



no overlap

FN



no prediction

Метрики

2) Основная метрика - mean Average Precision (mAP).

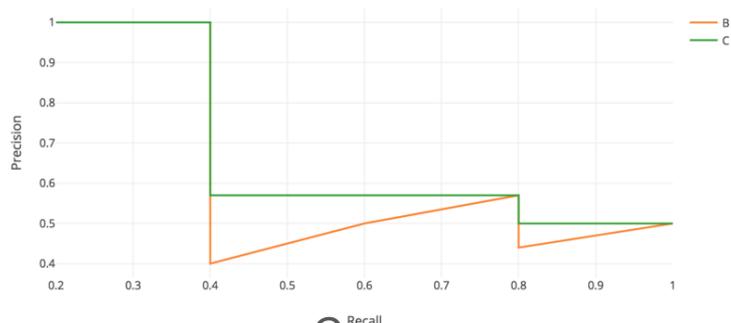
$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$AP = \frac{1}{11} \times (AP_r(0) + AP_r(0.1) + \dots + AP_r(1.0))$$

mAP metric это среднее значение APs для всех классов в наборе данных.

Мы можем вычислить несколько mAP для любого порога IoU. Если порог равен IoU 0.75, то метрика mAP для этого порога обозначается как mAP @ .75



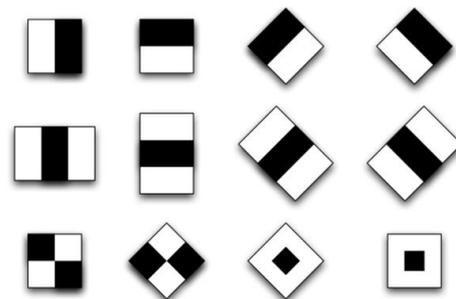
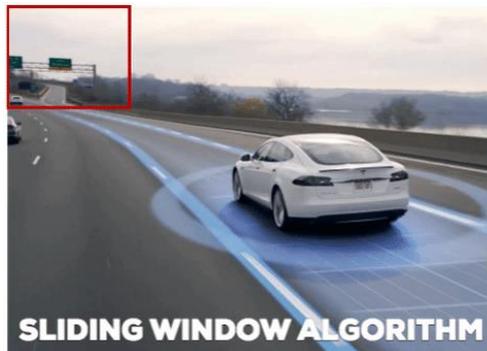
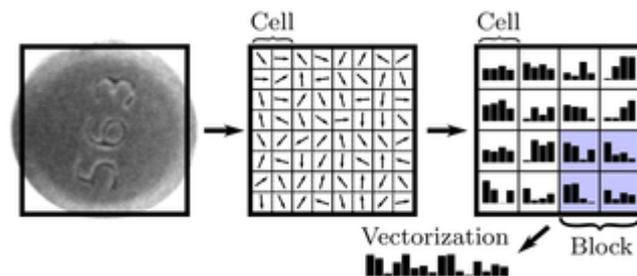
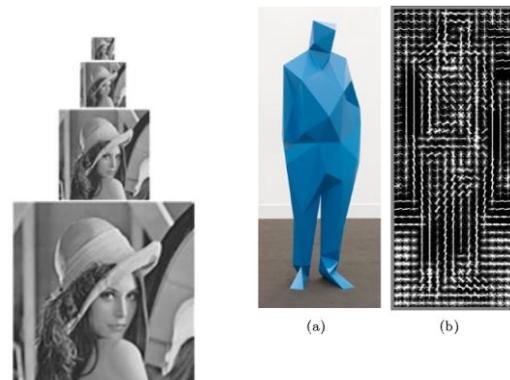
Изменяя порог, мы можем получить разные значения Precision and Recall. Оранжевая линия - это реальная кривая PR, а зеленая линия - максимальная.

Средняя точность для конкретного класса вычисляется так: усредняем максимальные Precision полученные для значений Recall [0.0, 0.1, ..., 0.9, 1.0]

Классические подходы

- 1) Гистограммы ориентированных градиентов (HOG)
- 2) Каскады Хаара
- 3) SIFT

Можно запустить классификатор в области изображения в режиме раздвижного окна.



Нейросетевой подход

1) Region-based

- a) RCNN
- b) Fast RCNN
- c) Faster RCNN

2) One-shot:

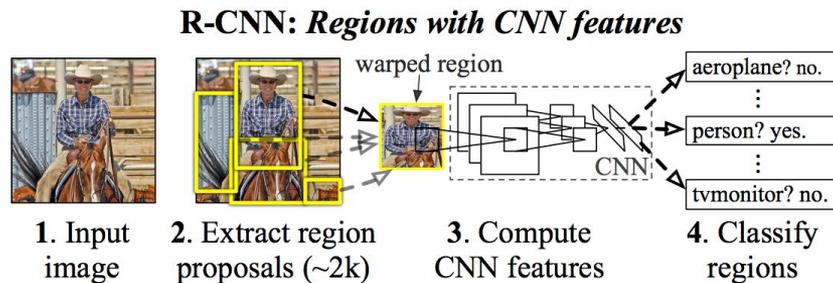
- a) SSD
- b) YOLO

3) Points-based

- a) CenterNet
- b) CornerNet
- c) ExtremeNet

RCNN (Regions with CNN features)

- 1) Формирование областей с использованием любого классического метода
- 2) Каждая область с исходного изображения подается на вход в CNN, подготовленную для классификации изображений.
- 3) Оценка полученного вектора характеристик с использованием класса, обученного для каждой категории изображений.
- 4) Порог обнаружения



Непрерывное подавление (NMS)

В большинстве случаев мы получаем несколько сильно перекрывающихся областей для одного объекта, тогда как нам нужен только один. NMS отбрасывает ограничивающие прямоугольники, которые имеют высокое перекрытие с ограничивающими прямоугольниками с более высоким коэффициентом обнаружения.

Алгоритм NMS для одного класса:

Сортируйте ограничивающие блоки по показателям обнаружения в порядке убывания. Храните их в списке L.

Перейдем к списку L:

Выберите первый ограничивающий прямоугольник в L, назовите его B

Вычислите его IoU со всеми другими ограничивающими блоками в списке L.

Удалите ограничивающие прямоугольники из L, у которых IoU с B больше некоторого порога.



Преимущества и недостатки RCNN

Плюсы:

Первый «действительно хороший» детектор объектов

Минусы:

Слишком медленное. Дублирование вычислений в случае перекрывающихся областей (без совместного использования вычислений).

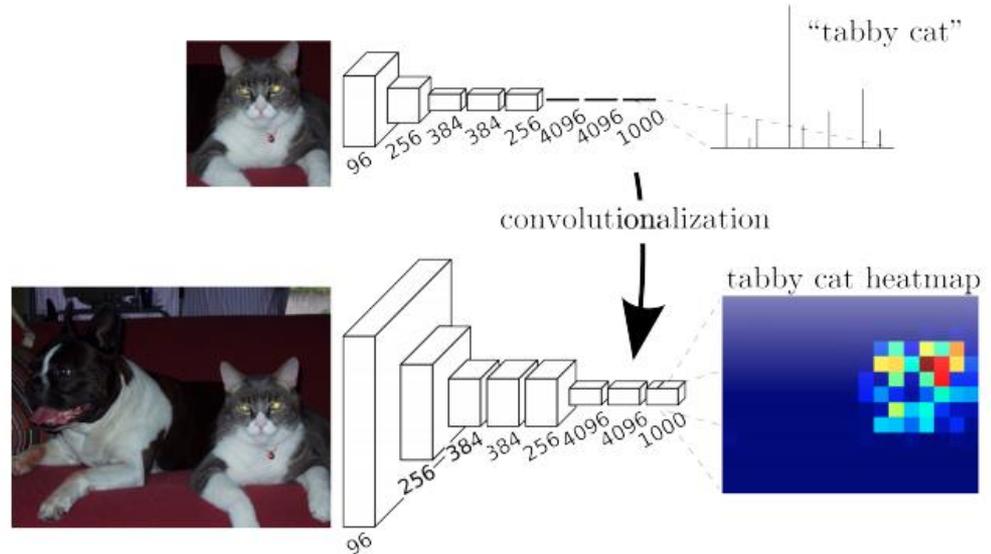
Изображения не заботятся о масштабе объекта / соотношении сторон.

Обучение является как вычислительно сложным, поэтому дорогостоящим.

Несколько этапов обнаружения, которые не очень удобны.

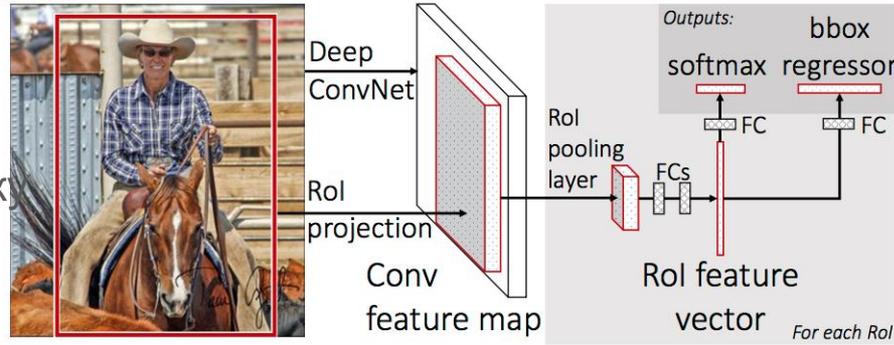
Концепция полной сверточной сети (FCN)

2) FC можно рассматривать как сверточный слой с фильтром, который покрывает весь входной тензор.



Fast RCNN

- 1) Используйте ту же технику выделения областей (выборочный поиск)



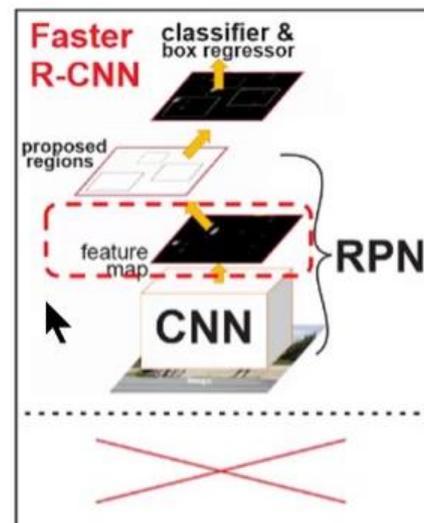
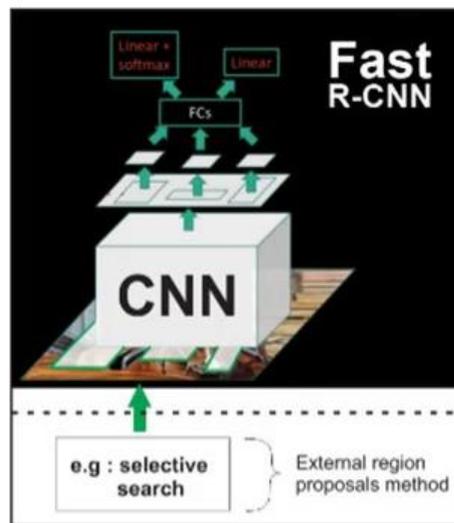
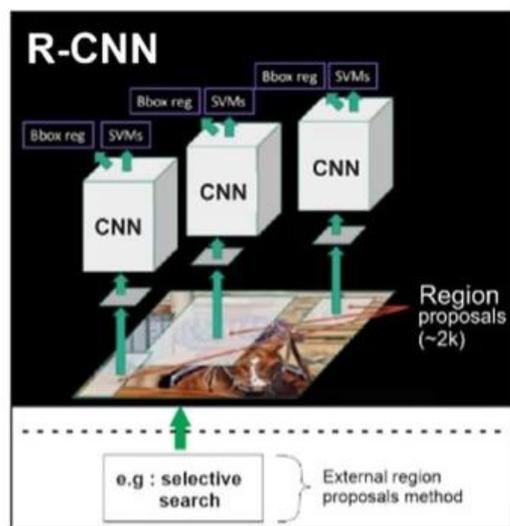
input

0.88	0.44	0.14	0.16	0.37	0.77	0.96	0.27
0.19	0.45	0.57	0.16	0.63	0.29	0.71	0.70
0.66	0.26	0.82	0.64	0.54	0.73	0.59	0.26
0.85	0.34	0.76	0.84	0.29	0.75	0.62	0.25
0.32	0.74	0.21	0.39	0.34	0.03	0.33	0.48
0.20	0.14	0.16	0.13	0.73	0.65	0.96	0.32
0.19	0.69	0.09	0.86	0.88	0.07	0.01	0.48
0.83	0.24	0.97	0.04	0.24	0.35	0.50	0.91

- 2) Загрузите полное изображение в полностью сверточную сеть, получите карту признаков.
- 3) Вырежьте области с этой карты признаков, соответствующие выделенным областям.
- 4) Подайте эту карту признаков на полностью полносвязную сеть и предскажите класс области и размер ограничивающей рамки. Размер ограничивающих полей является специфичным для класса.
- 5) примените nms на полученных ограничивающих прямоугольниках.

Fast RCNN

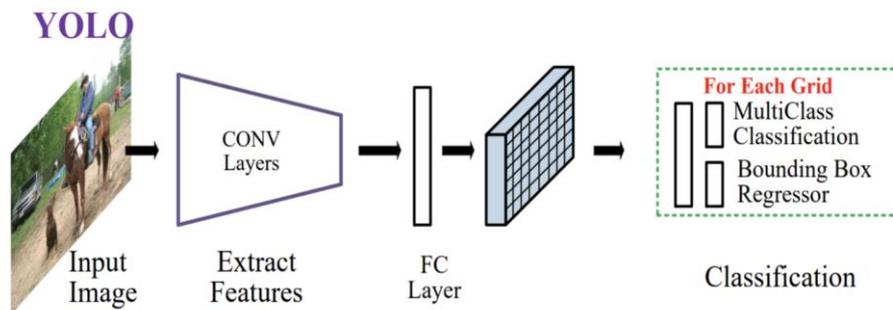
- 1) Обе функции потерь для классификации и регрессии размера прямоугольной рамки оптимизируются совместно.
- 2) Свертки применяются к входному изображению только один раз, но полносвязные слои применяются к каждой выделенной области.



	R-CNN	Fast R-CNN	Faster R-CNN
Test time per image	50 seconds	2 seconds	0.2 seconds
Speed-up	1x	25x	250x
mAP (VOC 2007)	66.0%	66.9%	66.9%

Обнаружение объектов за один проход

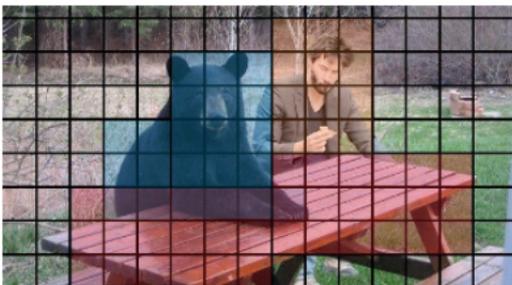
1. Получение признаков из основной сети
2. Прогнозирование карты признаков (“сетки”)
3. Прогнозирование ограничительных прямоугольников из сетки
4. Постобработка прогнозов



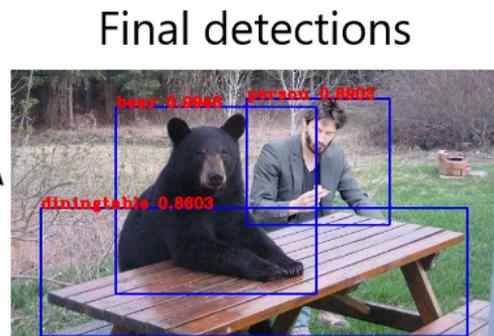
YOLO



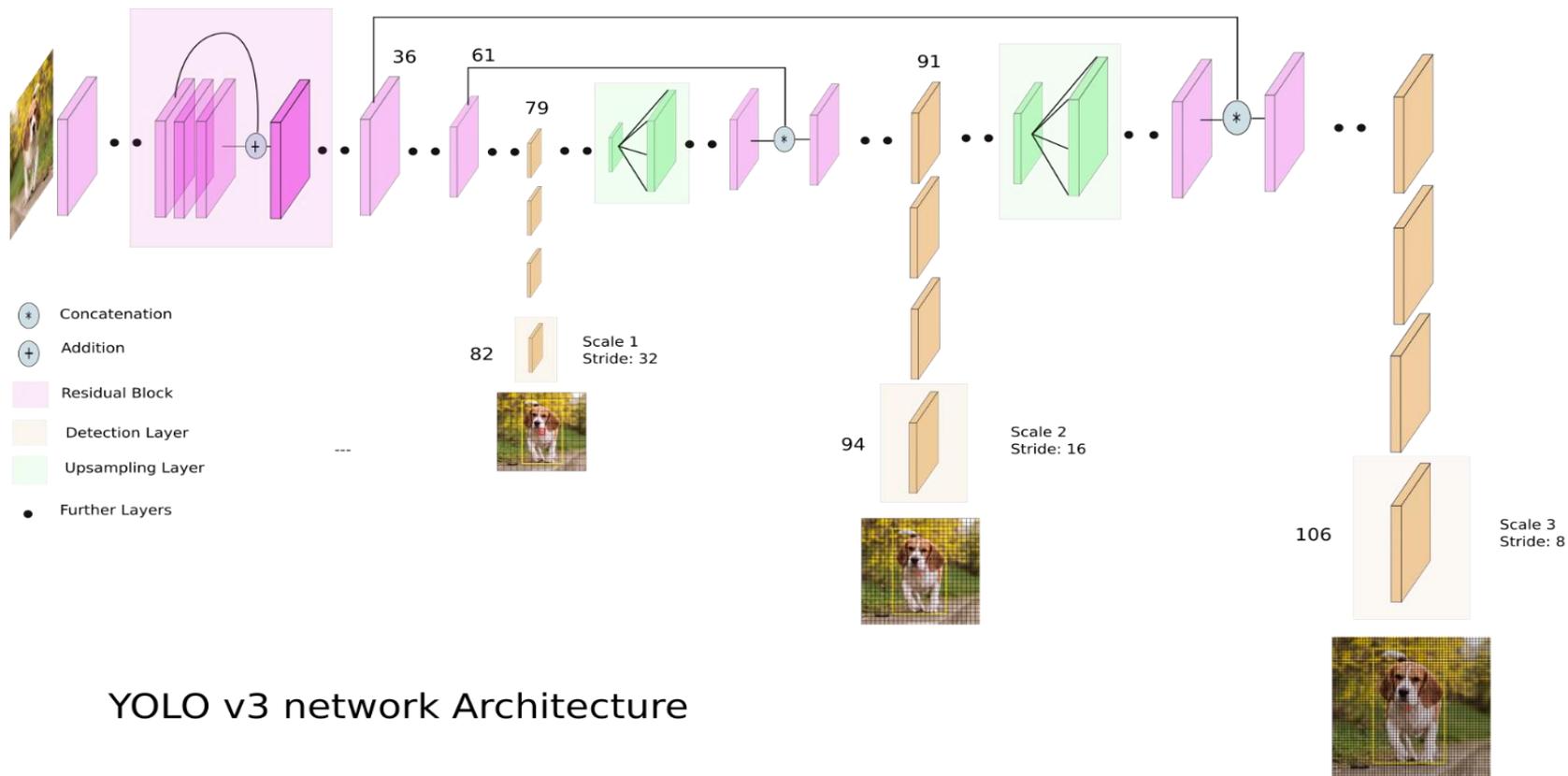
Bounding boxes



Class probability map



YOLO



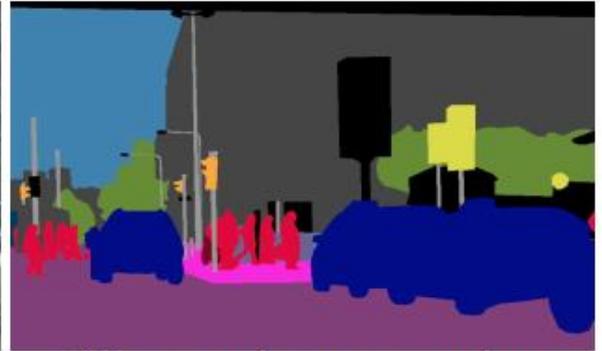
YOLO v3 network Architecture

Сегментация изображений

Задача: дать распределение вероятности по классам для каждого пикселя изображения.

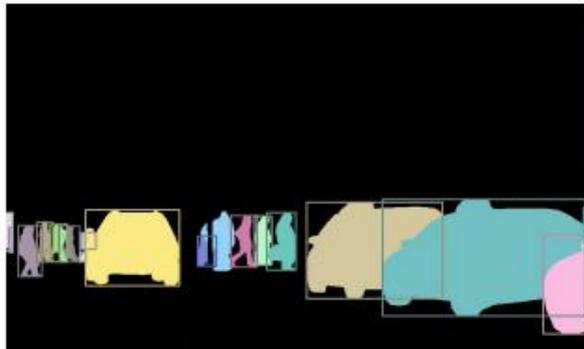


(a) image

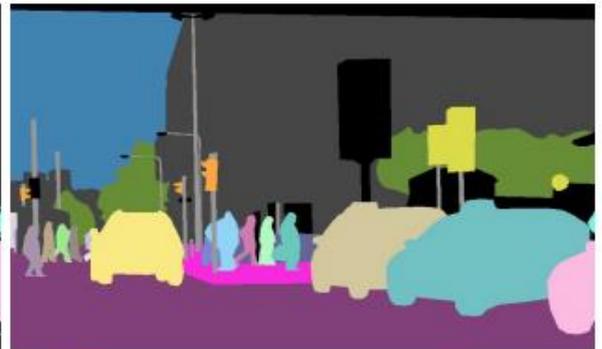


(b) semantic segmentation

Паноптическая сегментация: назначает каждому пикселю метку экземпляра



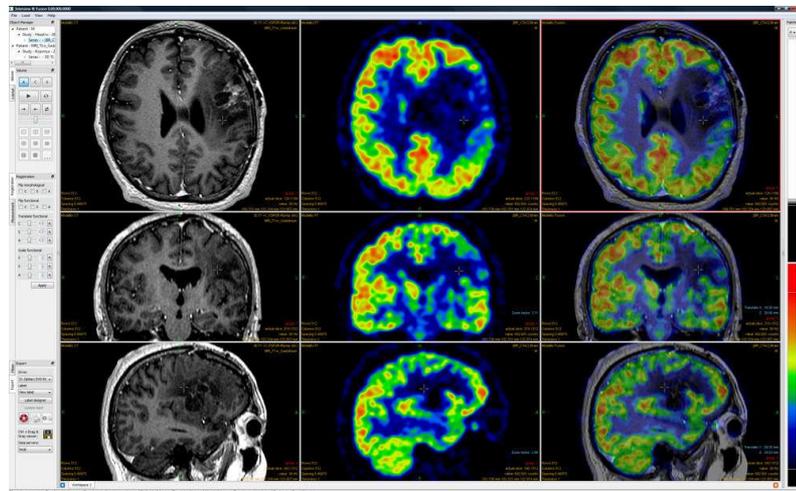
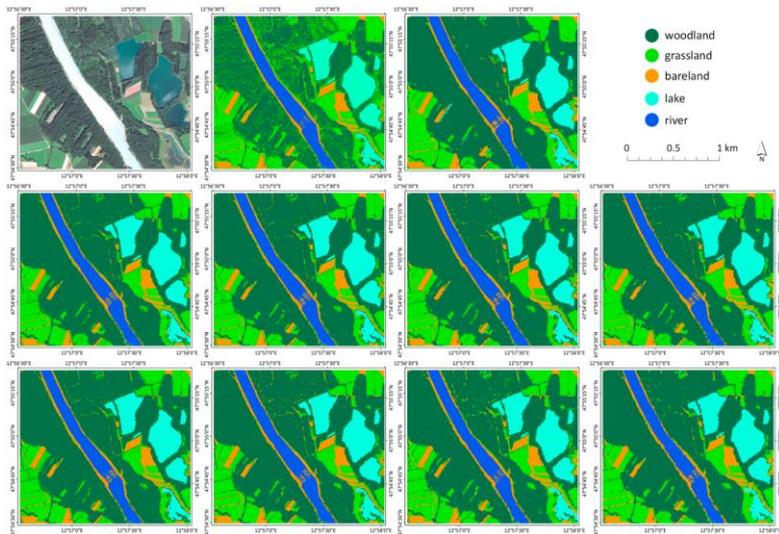
(c) instance segmentation



(d) panoptic segmentation

Приложения

- 1) Медицинские изображения
- 2) Снимки из космоса
- 3) ...



Метрики

- n_{cl} : количество классов, включенных в сегментацию
- n_{ij} : количество пикселей класса i , которое попало в класс j
- t_i : общее количество пикселей класса i в сегментации Pixel Accuracy:

1) Mean Accuracy:
$$\frac{\sum_i n_{ii}}{\sum_i t_i}$$

2) Mean IU:
$$\frac{1}{n_{cl}} \sum_i \frac{n_{ii}}{t_i}$$

3) Panoptic Quality (PQ)
$$\frac{1}{n_{cl}} \sum_i \frac{n_{ii}}{(t_i + \sum_j n_{ji} - n_{ii})}$$

Идея сегментации с помощью CNNs

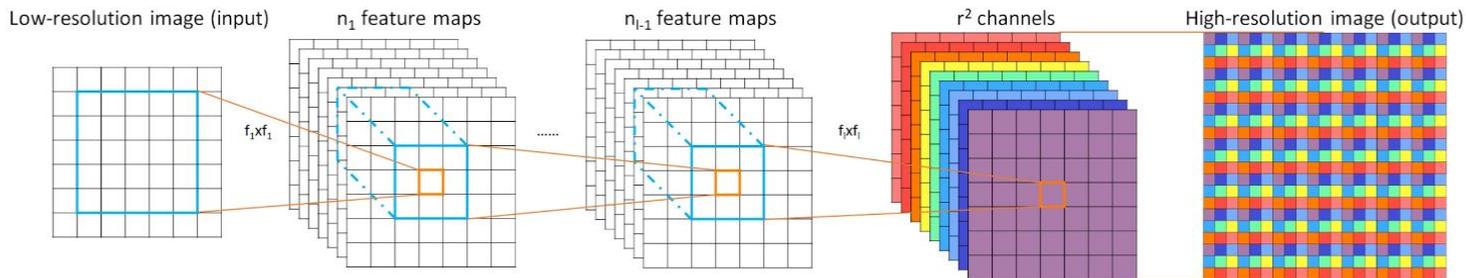
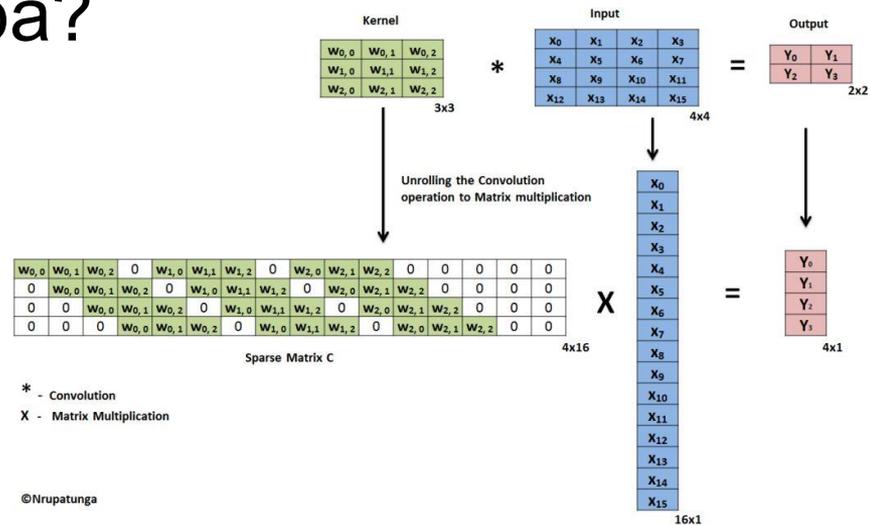
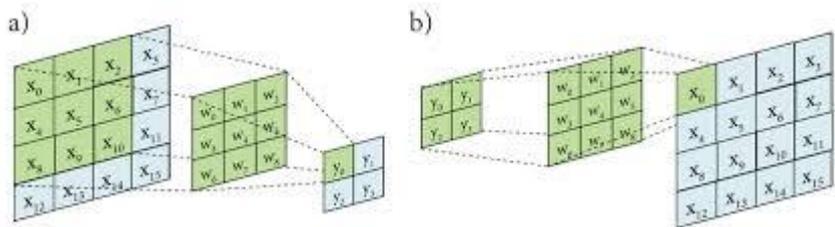
Как обычно, мы получаем вектор сверточных характеристик малого размера, а затем увеличиваем его до размера входного изображения, за исключением количества каналов.

Количество каналов ($N + 1$), где N - число классов сегментации.

Применяется попиксельный softmax, чтобы получить распределение вероятности по классам для каждого пикселя входного изображения.

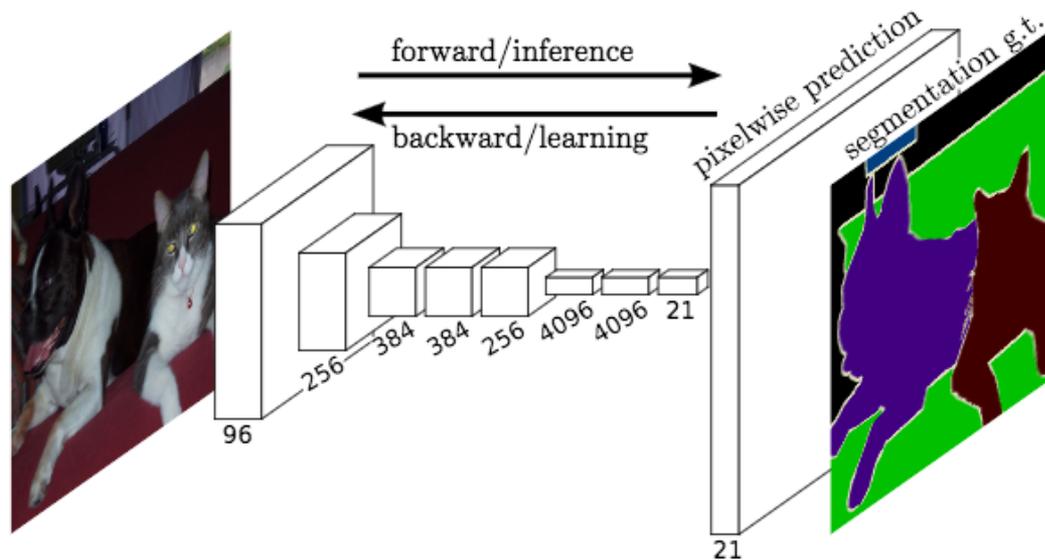
Как повысить размер тензора?

- 1) Билинейная интерполяция
- 2) Транспонированная свертка
- 3) Уровень пиксельной перетасовки



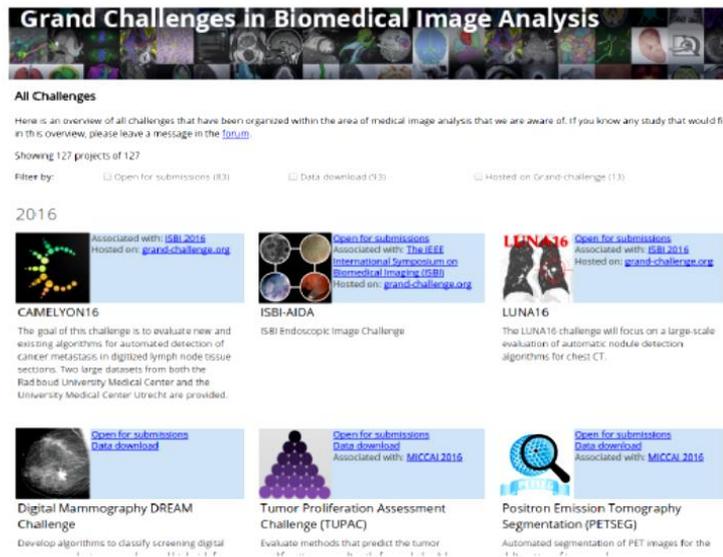
FCN для семантической сегментации

- 1) Полностью сверточная
- 2) AlexNet как основа
- 3) Одношаговый upsampling с транспонированной сверткой с фильтрами размером 64x64 и шагом 32



Задача сегментации для биомедицинских данных

- Многие биомедицинские изображения более просты, чем визуальные сцены
- Одна из основных проблем в медицинской сегментации — малый объём данных (десятки-сотни снимков)



Grand Challenges in Biomedical Image Analysis

All Challenges

Here is an overview of all challenges that have been organized within the area of medical image analysis that we are aware of. If you know any study that would fit in this overview, please leave a message in the [forum](#).

Showing 127 projects of 127

Filter by: Open for submissions (83) Data download (93) Hosted on Grand-challenge (13)

2016

CAMELYON16
Associated with [ISBI 2016](#)
Hosted on: [grand-challenge.org](#)
The goal of this challenge is to evaluate new and existing algorithms for automated detection of cancer metastases in digitized lymph node tissue sections. Two large datasets from both the Radboud University Medical Center and the University Medical Center Utrecht are provided.

ISBI-AIDA
ISBI Endoscopic Image Challenge
Open for submissions
Data download
Associated with [The IEEE International Symposium on Biomedical Imaging \(ISBI\)](#)
Hosted on: [grand-challenge.org](#)

LUNA16
Open for submissions
Data download
Associated with [ISBI 2016](#)
Hosted on: [grand-challenge.org](#)
The LUNA16 challenge will focus on a large-scale evaluation of automatic nodule detection algorithms for chest CT.

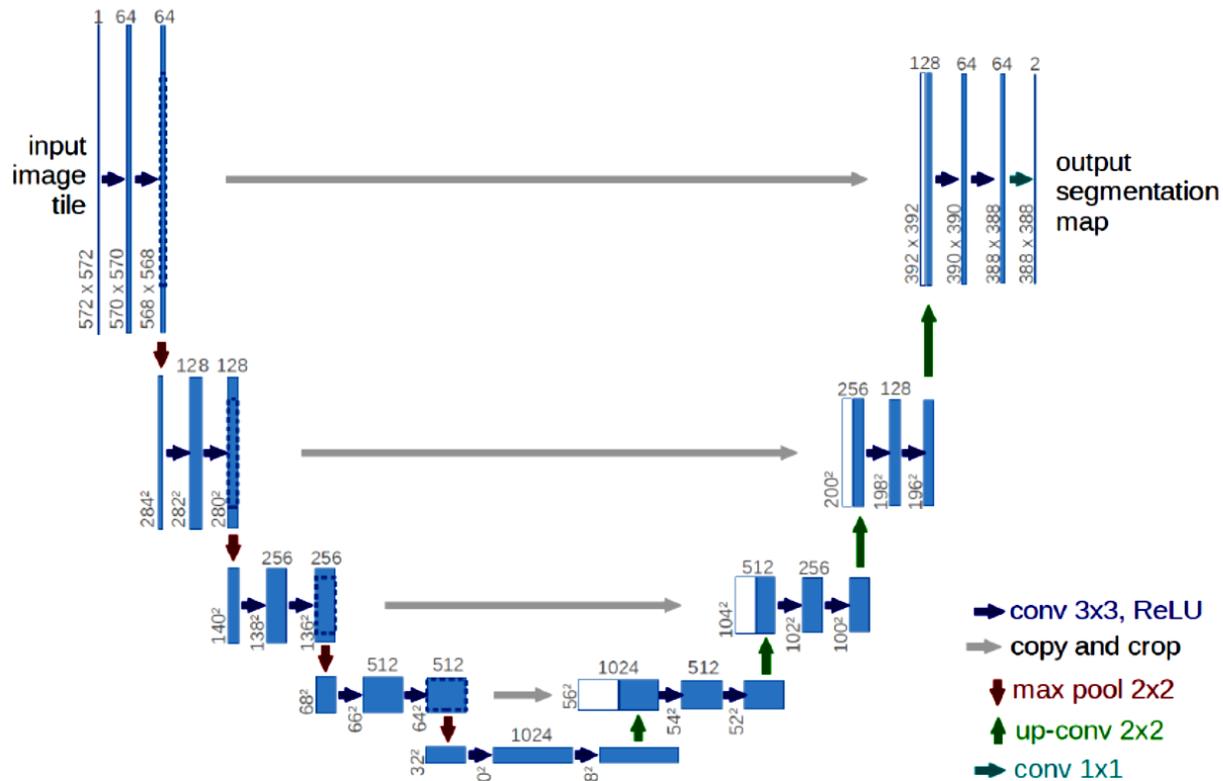
Digital Mammography DREAM Challenge
Open for submissions
Data download
Develop algorithms to classify screening digital

Tumor Proliferation Assessment Challenge (TUPAC)
Open for submissions
Data download
Associated with [MICCAI 2016](#)
Evaluate methods that predict the tumor

Positron Emission Tomography Segmentation (PETSEG)
Open for submissions
Data download
Associated with [MICCAI 2016](#)
Automated segmentation of PET images for the

https://grand-challenge.org/all_challenges/

Архитектура U-Net



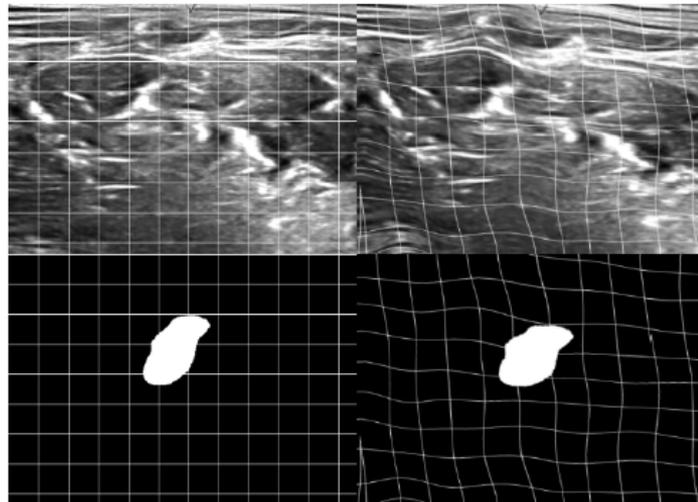
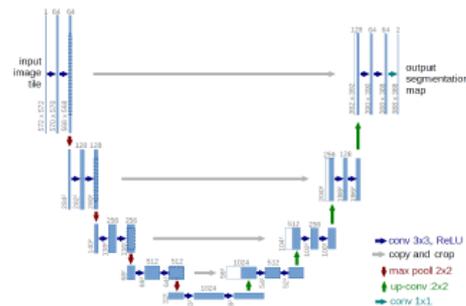
Обучение U-Net

- Mini-batch SGD with momentum:

$$v \leftarrow \gamma v + \alpha \nabla_{\theta} \left(\frac{1}{m} \sum_{i=1}^m L(f(x^{(i)}; \theta), y^{(i)}) \right)$$

$$\theta \leftarrow \theta - v$$

- Авторы предпочитают обучать на больших изображениях, а не с большим batch size.
- batch size = 1 (online learning), momentum = 0.99.
- Data augmentation: elastic transforms



Обучение U-Net

Границы объектов усиливаются по сравнению с внутренностями. Каждый пиксель корректной разметки $w_c(\mathbf{x})$ пересчитывается по формуле

$$w(\mathbf{x}) = w_c(\mathbf{x}) + w_0 \cdot \exp\left(-\frac{(d_1(\mathbf{x}) + d_2(\mathbf{x}))^2}{2\sigma^2}\right),$$

где $d_1(\mathbf{x}), d_2(\mathbf{x})$ — расстояния от \mathbf{x} до двух ближайших клеток.

