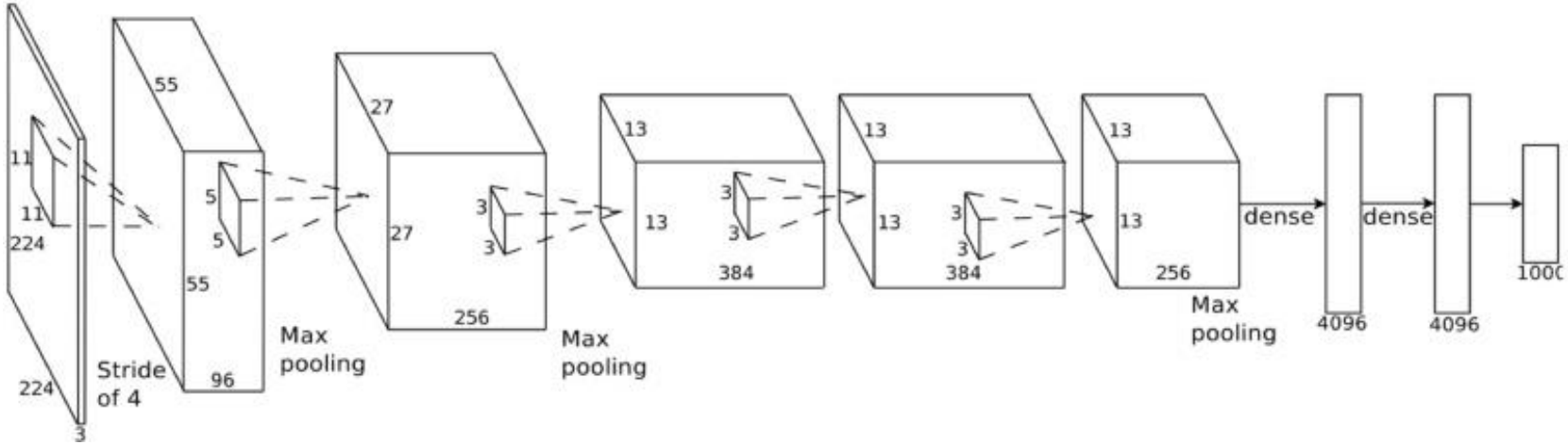
Глубинное обучение
Лекция 7: Свёрточные сети в задачах компьютерного зрения

Лектор: Антон Осокин

ФКН ВШЭ, 2020

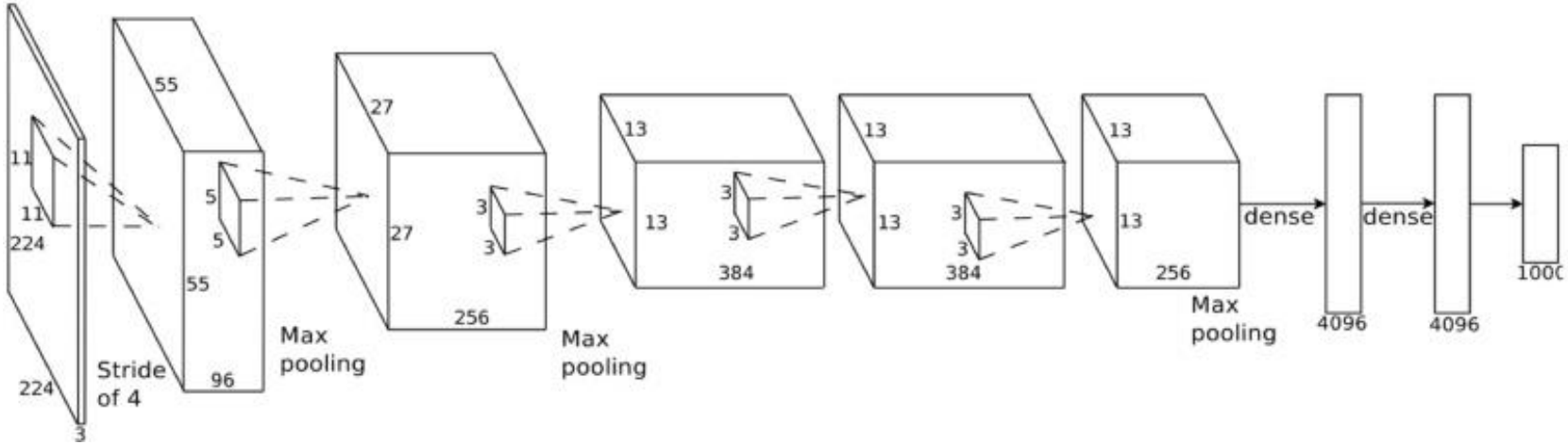
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Recap: классификация изображений



- Задача классификации изображений решена! (в нормальных условиях)
- Вход сети – изображение
- Выходы сети соответствуют классам
- Обучение с разметкой, сделанной людьми.
- Кросс-энтропийная функция потерь (log loss)
- Много архитектур сетей (например, ResNet [He et al.; 2015])
- Основные слои: свёртки, нелинейность, пулинг, нормировка (batchnorm)

Recap: классификация изображений



- Задача классификации изображений решена! (в нормальных условиях)
- Вход сети – изображение
- Выходы сети соответствуют классам
- Обучение с разметкой, сделанной людьми.
- Кросс-энтропийная функция потерь (log loss)
- Много архитектур сетей (например, ResNet [He et al.; 2015])
- Основные слои: свёртки, нелинейность, пулинг, нормировка (batchnorm)
- Перенос успеха в другие задачи:
  - переиспользовать выученные представления

План лекции

- Обнаружения объектов (object detection)
  - R-CNN, Fast R-CNN, Region Proposal Networks
  - Быстрые детекторы: SSD and YOLO
- Сегментация изображений (image segmentation)
  - Fully convolutional networks
  - Masked R-CNN
- Поиск похожих изображений (image retrieval)
  - Siamese architecture
  - Отслеживание объектов на видео
- Обучение без разметки (self-supervised pretraining)
- Распознавание действий на видео (action recognition)

Обнаружение объектов (detection)

- Задача найти объекты на изображении
- Найти = поставить прямоугольник (bounding box)

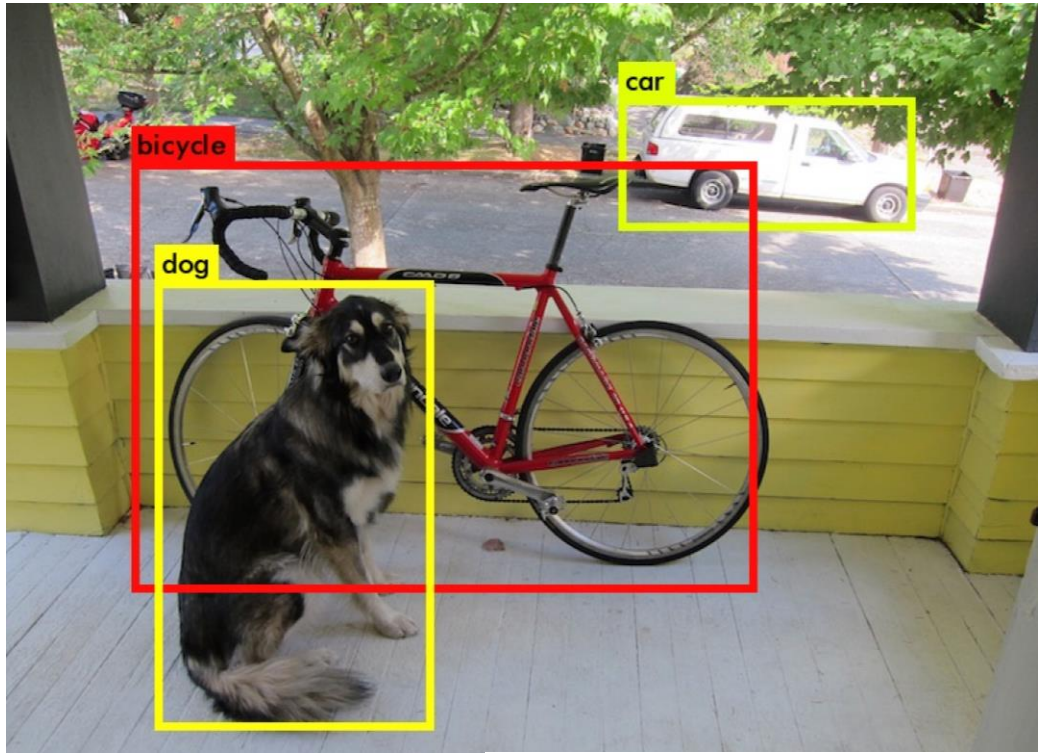
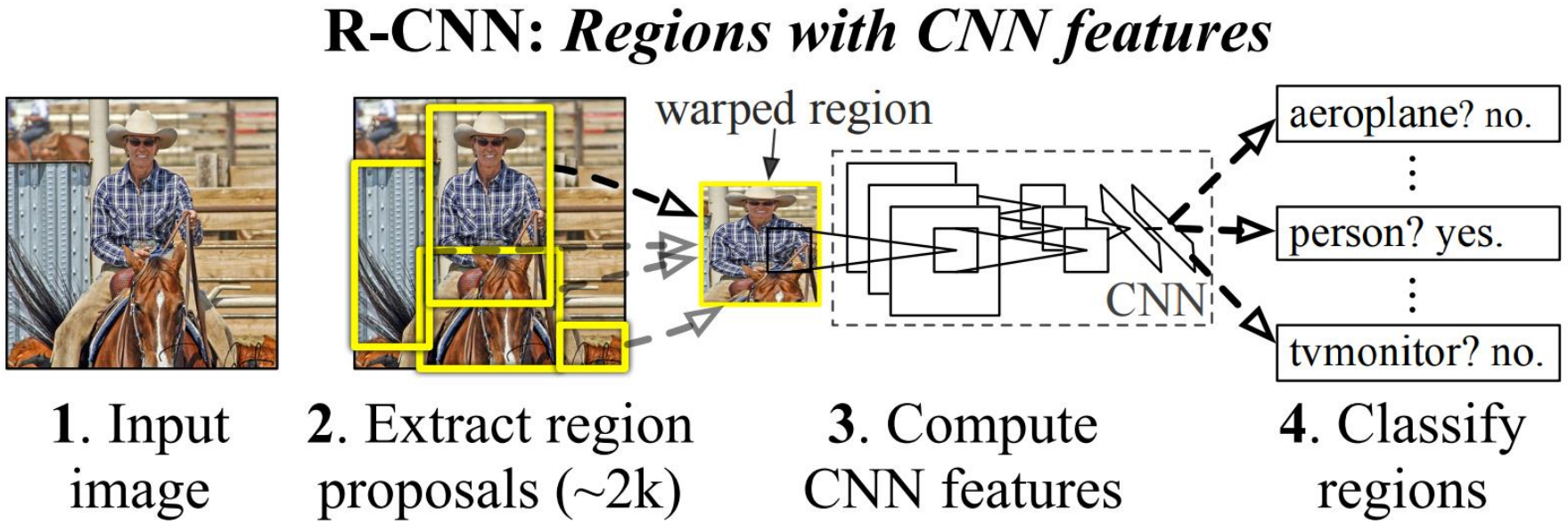


image credit: Joseph Redmon

Ранние методы: R-CNN

[Girshick et al., 2013]

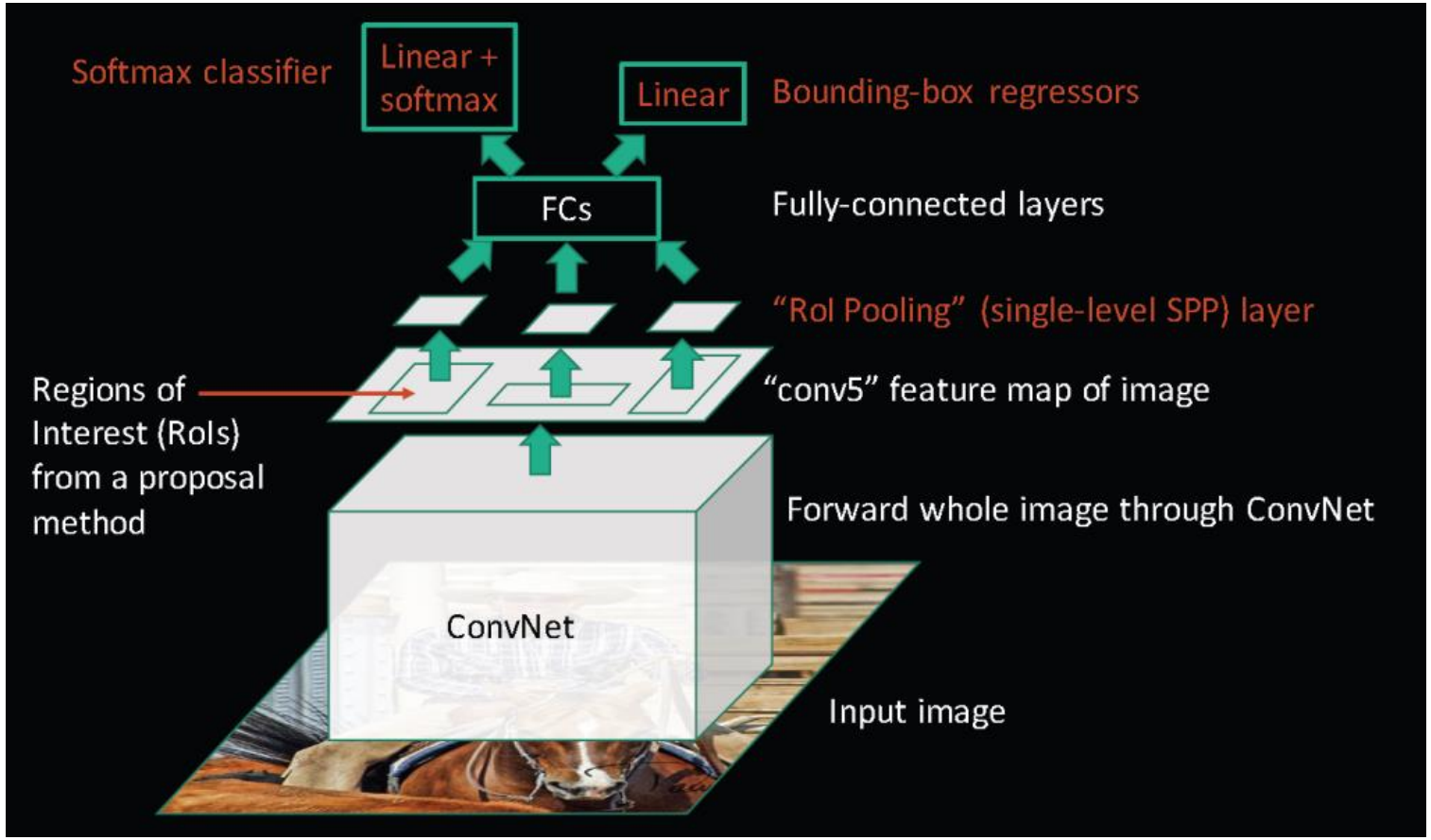


- Основная идея – классифицировать гипотезы (object proposals)
- Используем CNN для каждой гипотезы
- На выходе: метка класса и уточнение позиции объекта
- Проблема: сильный дисбаланс объектов и фона
  - Контроль баланса в батче, специальные функции потерь (focal loss)

Fast R-CNN

[Girshick 2015]

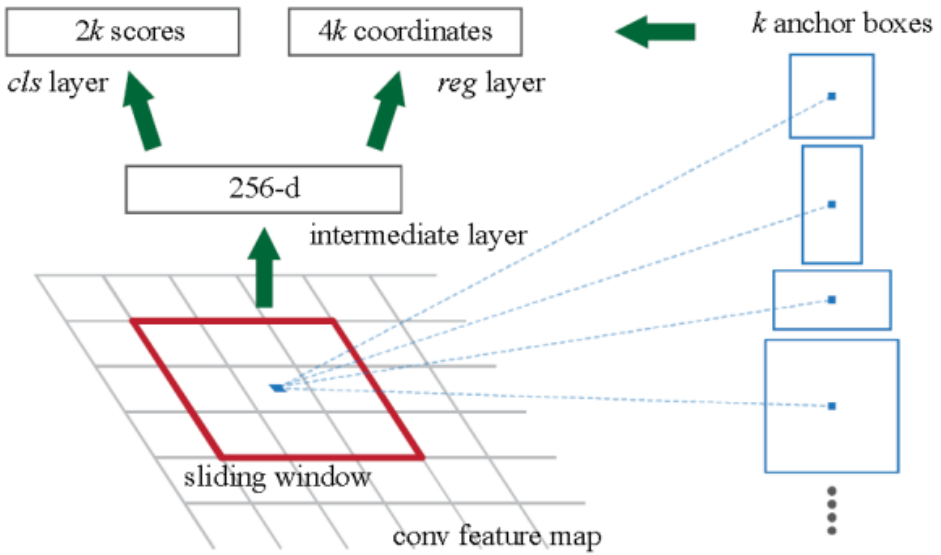
- Недостаток R-CNN – медленная скорость работы
- Много пересекающихся гипотез – неэффективно
- Идея: разделить вычисления свёрток между гипотезами

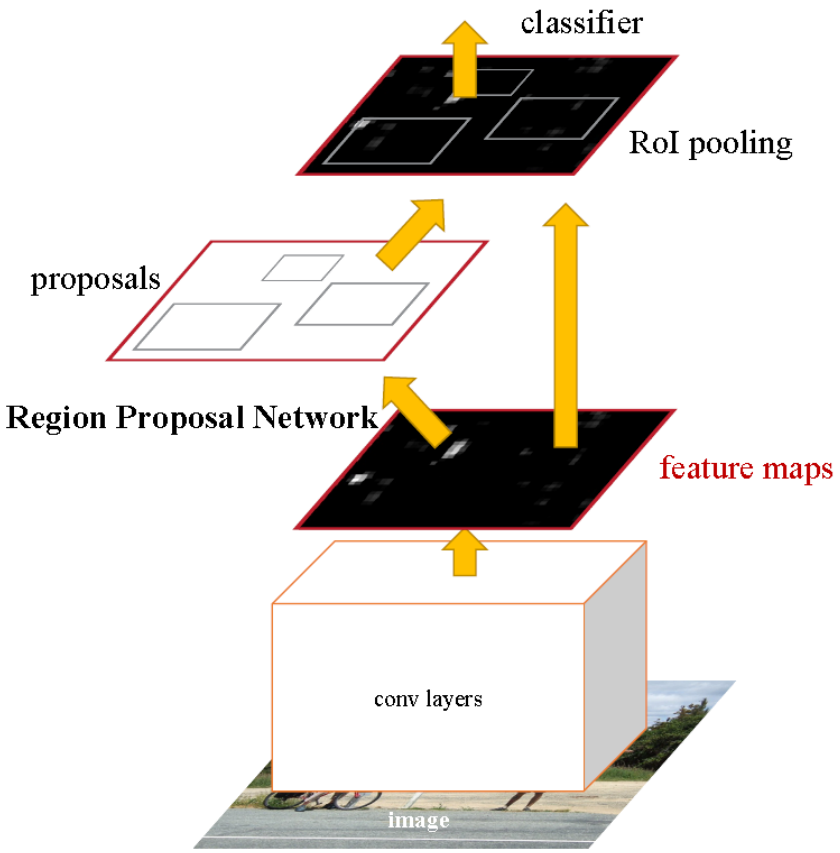


Region proposal network: Faster R-CNN

[Ren et al., 2015]

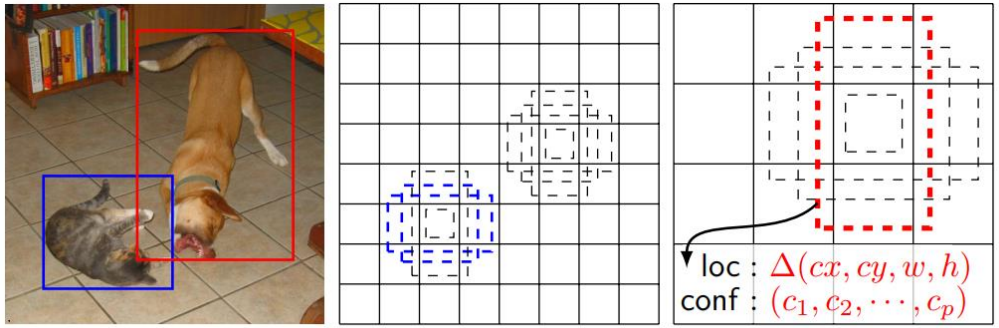
- Fast R-CNN нужны гипотезы
- Гипотезы считать медленно
- Идея: гипотезы из сети
- 5-17 FPS



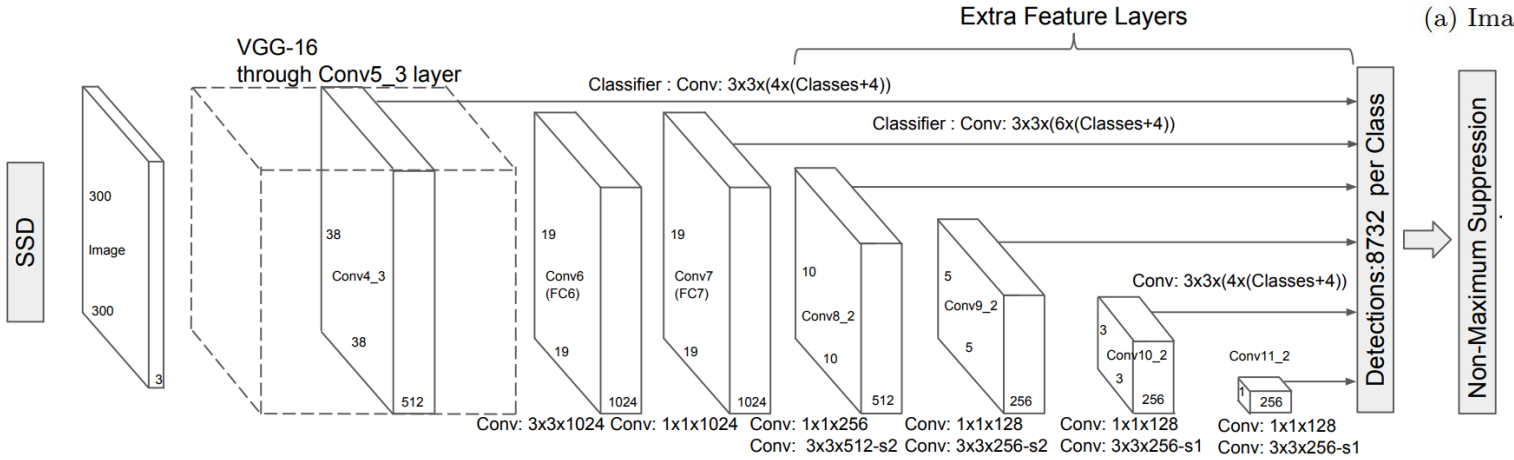


Fast detectors: YOLO, SSD, RetinaNet

- Идея: отказ от двух стадий модели, ответ за 1 проход
- Только RPN
- SSD: 59 FPS



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map



[Liu et al., 2016]

- Конкуренция моделей из 2-х и 1-й стадии (скорость и качество)
- Более новые модели: FCOS [Tian et al.; 2019], DETR [Carion et al.; 2020]

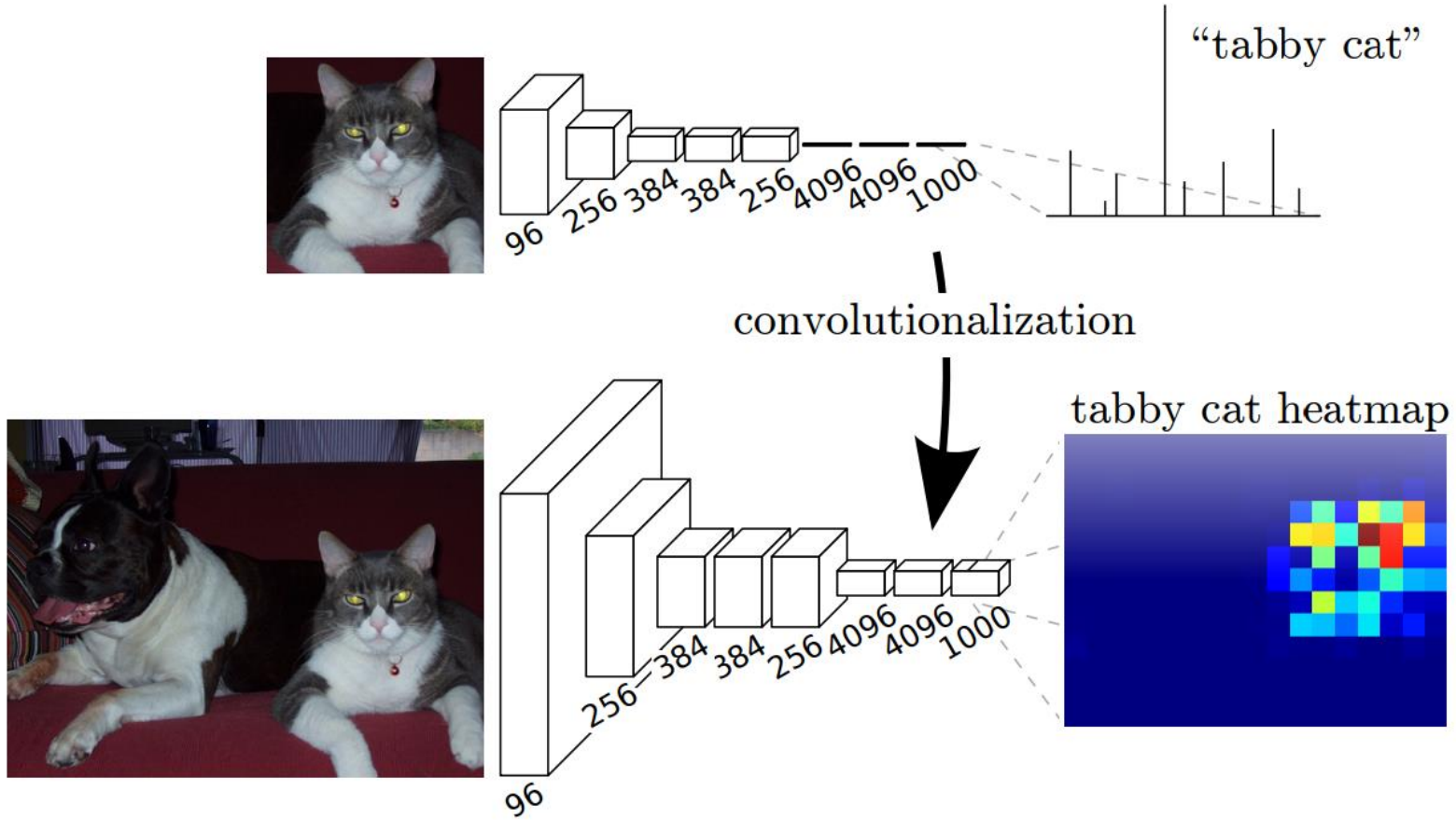
Сегментация изображений

- Задача найти объекты на изображении
- Найти = метки класса для пикселей

Fully-convolutional CNN

Идея из 90-х, [Long et al., 2015]

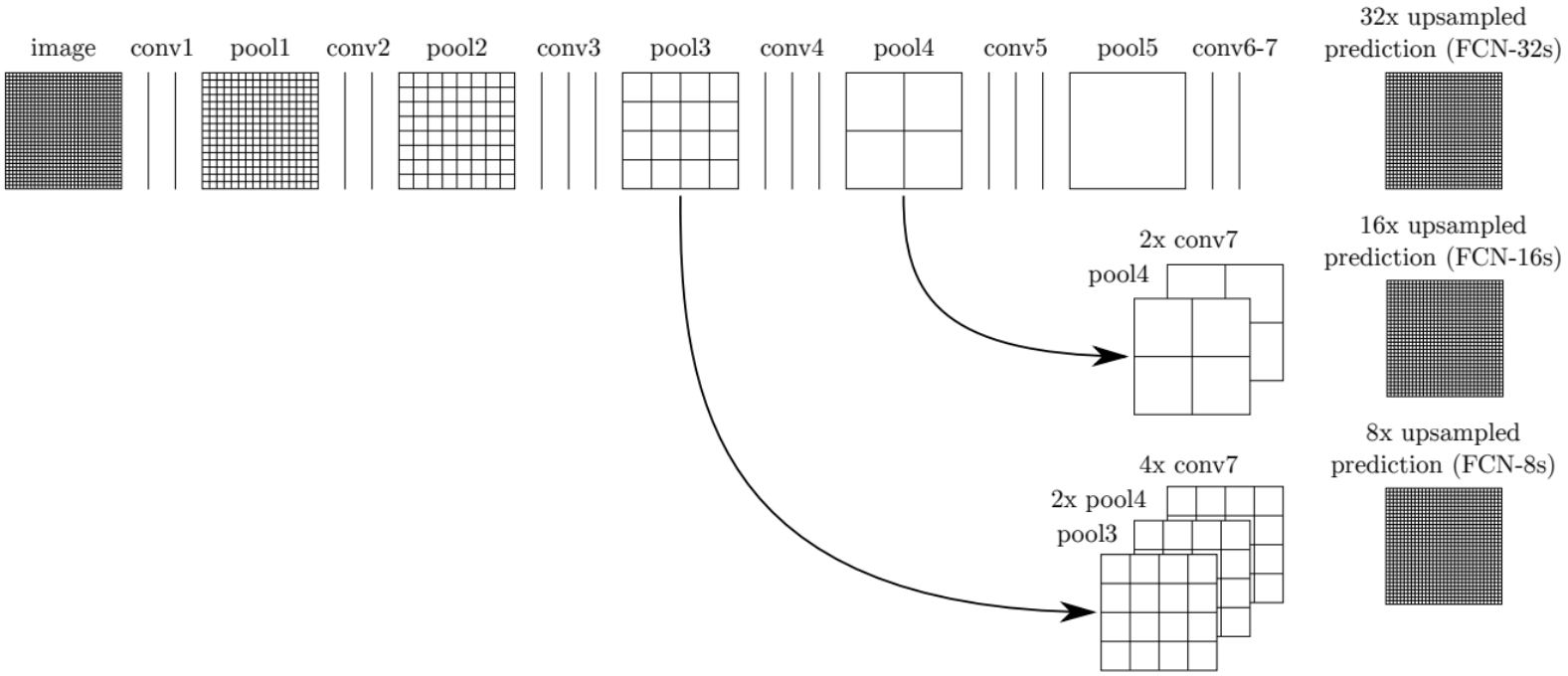
- Идея: применить CNN скользящим окном
- Недостаток – очень низкое разрешение выхода



Fully-convolutional CNN

[Long et al., 2015]

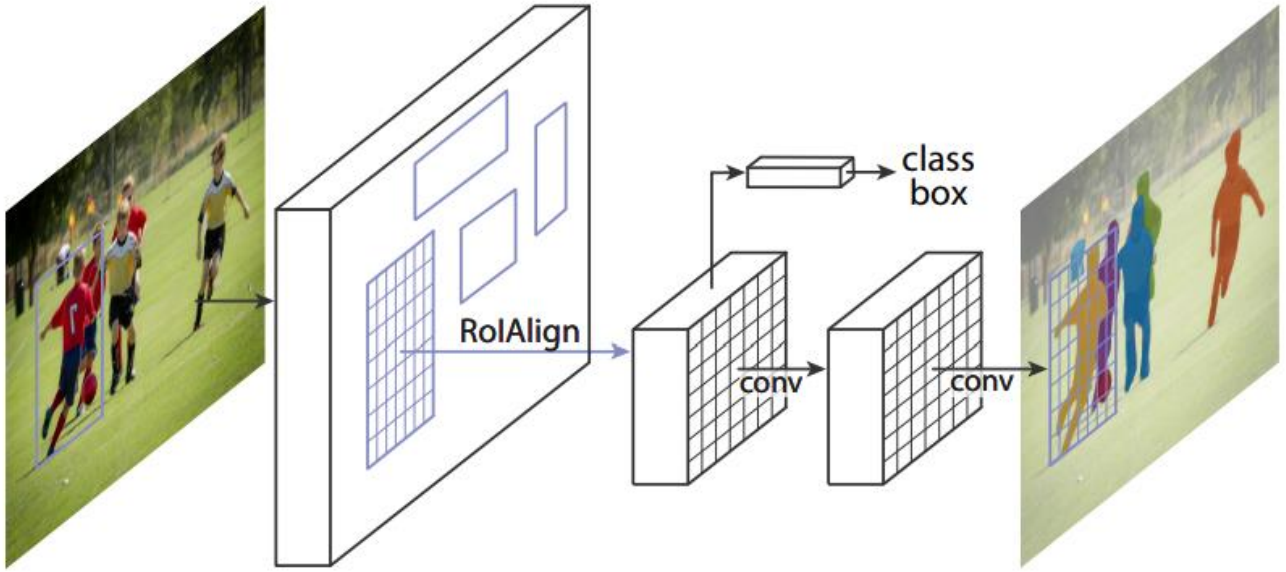
- Идея: применить CNN скользящим окном
- Недостаток – очень низкое разрешение выхода
- Идея: разрешение с помощью более глубоких слоев
- Используются upconv, dilated conv, etc.
- Модели такого типа известны как U-net (детали важны!)



Сегментация объектов: Mask R-CNN

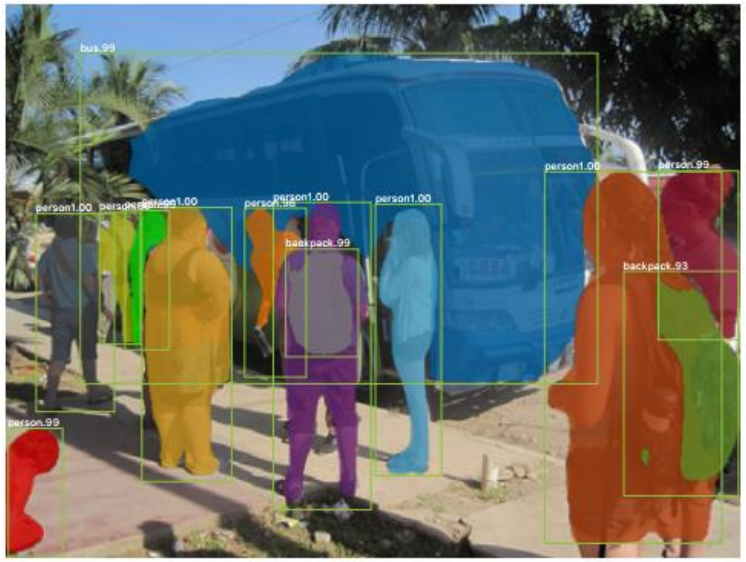
[He et al., 2017]

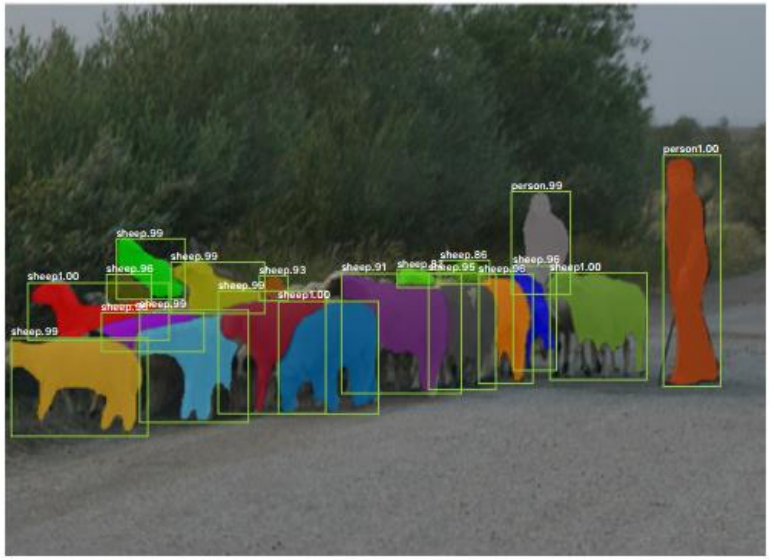
- Идея: использовать детекцию (Faster R-CNN) для сегментации
- Недостаток – из-за maxpool теряется точная позиция
- Идея: использовать «гладкий pooling»
- Билинейная интерполяция границ пикселей

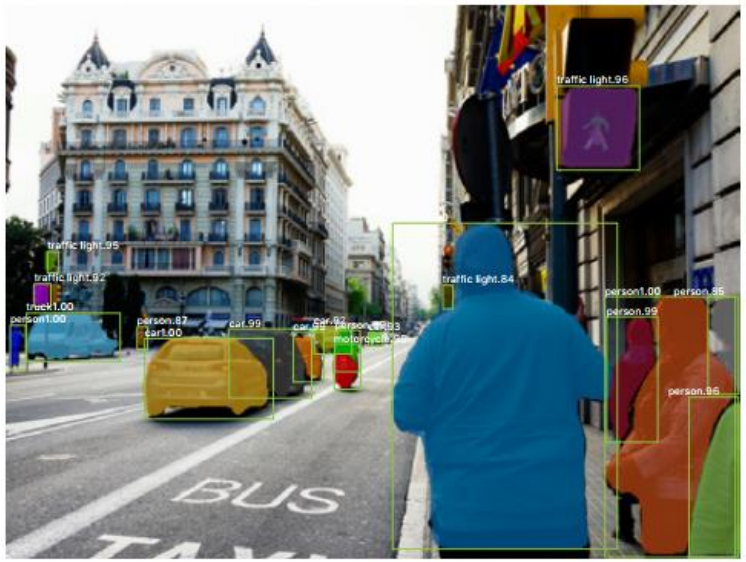


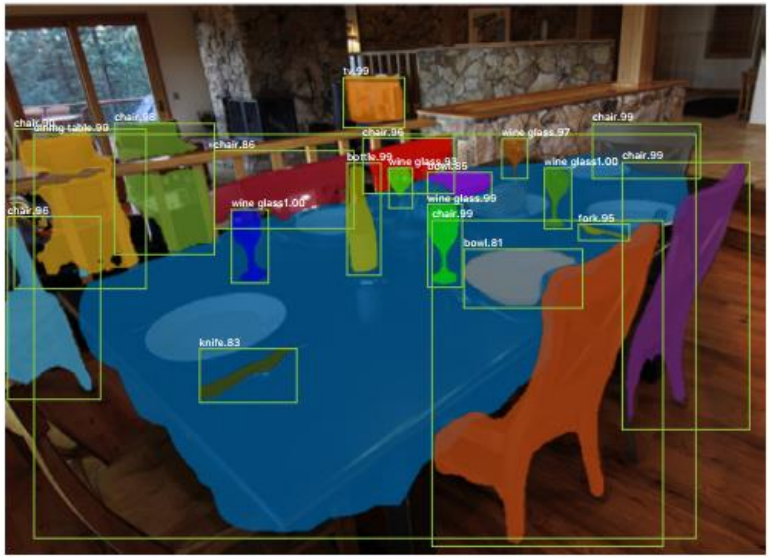
Сегментация объектов: Mask R-CNN

[He et al., 2017]









Поиск изображений (retrieval)

- Задача найти похожие изображения
- Задача идентификации (например, лица)
- Подход: описать изображение небольшим вектором (128, 256) и делать поиск ближайших соседей по L2 метрике
- Быстрые приближенный алгоритмы поиска
- Можно использовать предобученные сети
- Обучение специальных признаков!

Сиамские сети (Siamese nets)

- Идея: использовать одну и ту же сеть на двух изображениях, и считать расстояние между признаками
- Вопрос – как обучать?

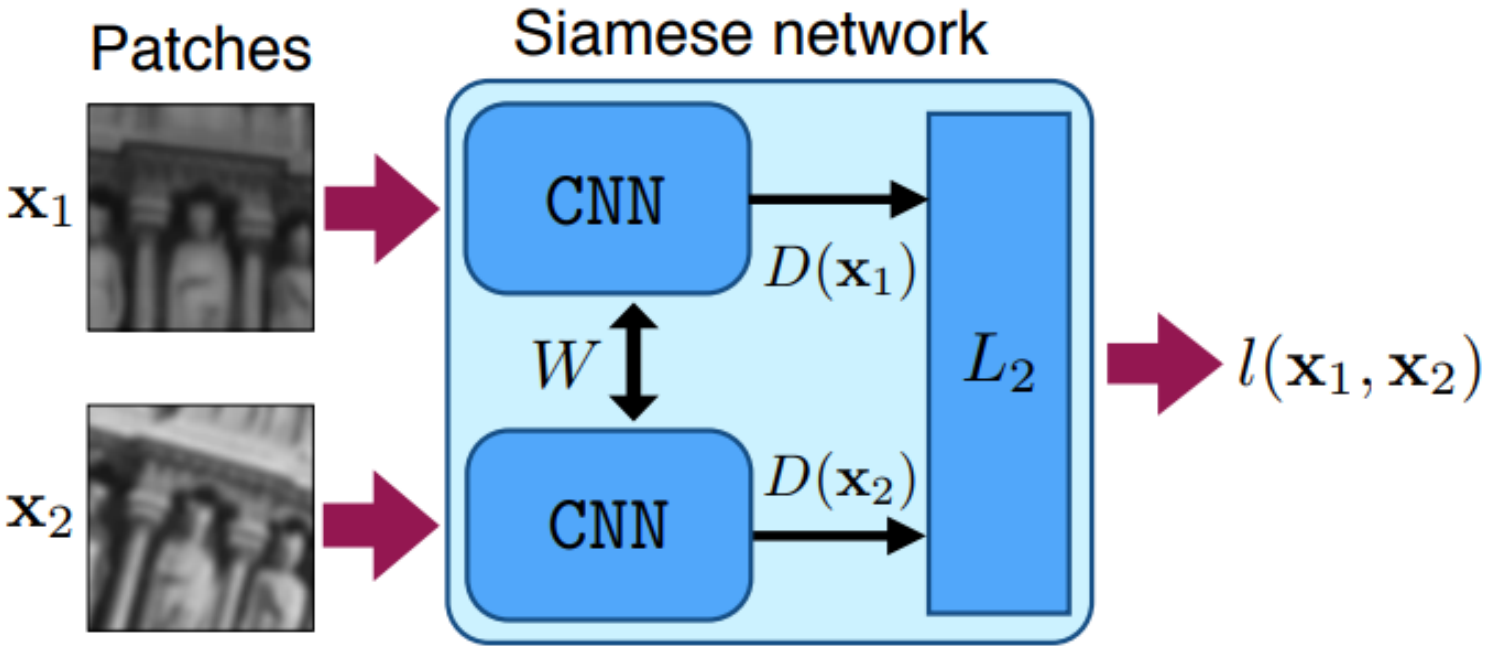


Image from [Simo-Serra et al., 2015]

Сиамские сети (siamese)

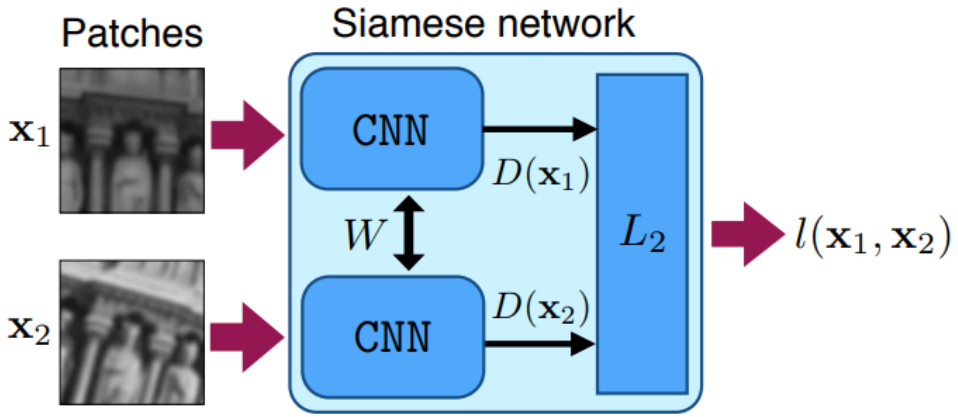
- Идея: использовать одну и ту же сеть на двух изображениях, и считать расстояние между признаками
- Вопрос – как обучать?

- Вариант 1 – Contrastive loss

y = 1 - положительная пара

y = 0 - отрицательная пара

m – margin, чтобы не отталкивать непохожие



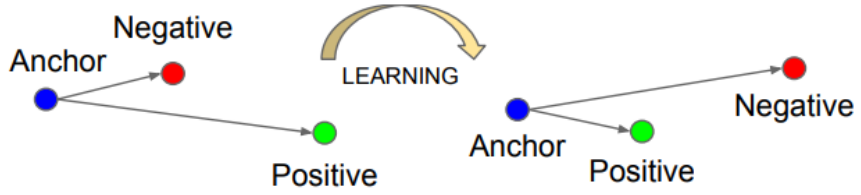
$$\ell(x_1, x_2) = y\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2 + (1-y)\max(0, m - \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2)$$

Векторы ***x*** нормированные!

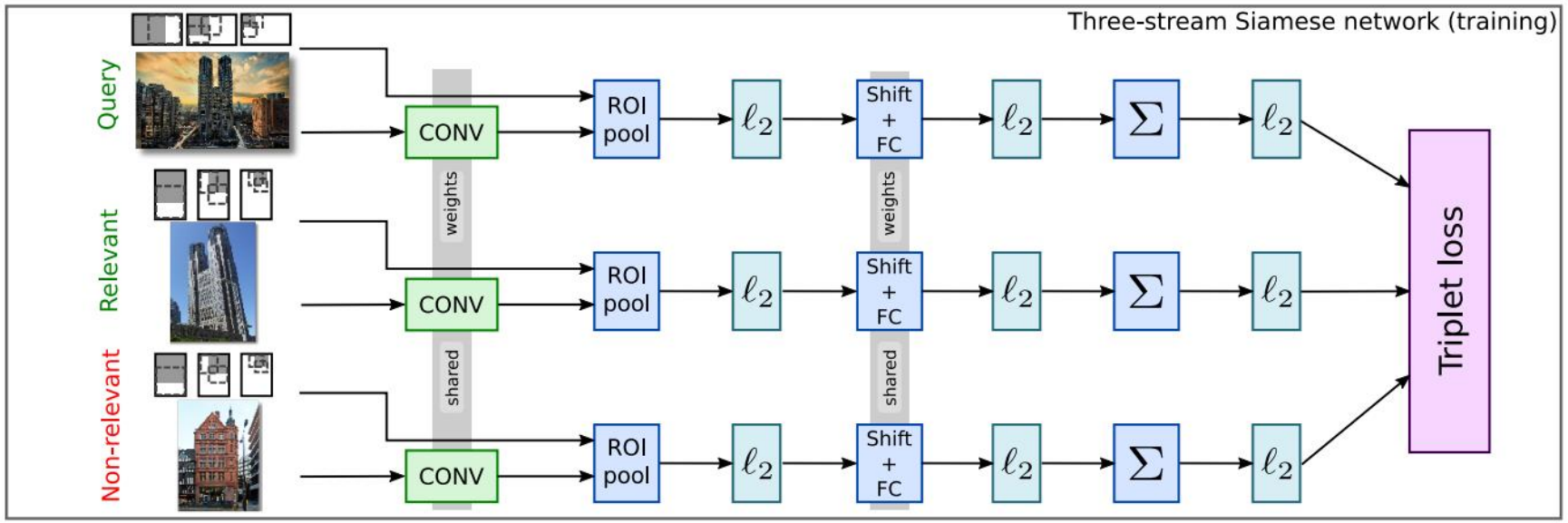
Image from [Simo-Serra et al., 2015]

Сиамские сети (siamese)

- Идея: использовать одну и ту же сеть на трёх изображениях, и считать расстояние между признаками
- Вариант 2 – Triplet loss



$$\max(0, m + \|q - d^+\|^2 - \|q - d^-\|^2)$$

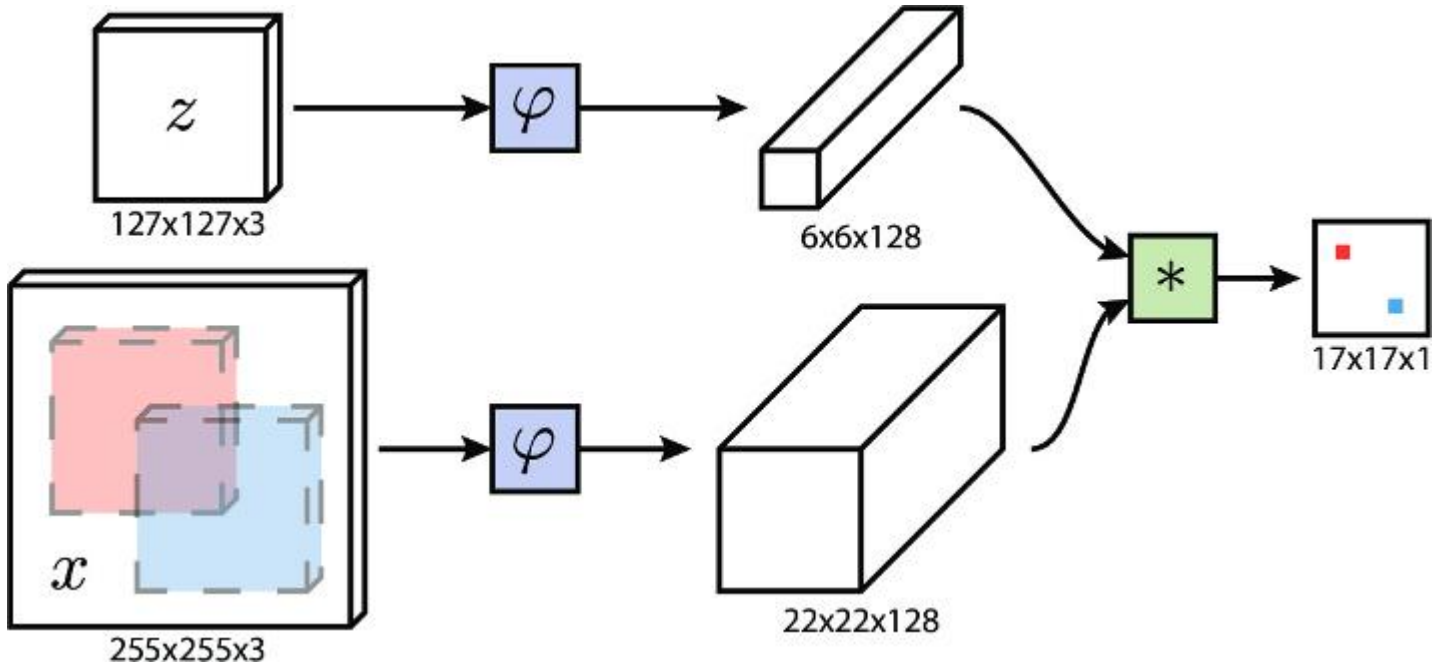


[Simo-Serra et al., 2015; Gordo et al., 2016]

Отслеживание объектов на видео

[Bertinetto et al., 2016]

- Идея: одну из веток сиамский сетей применять свёрточно



Отслеживание объектов на видео

[Bertinetto et al., 2016]

- Идея: одну из веток сиамский сетей применять свёрточно
- Real-time, online

Обучение без разметки (self-supervised pretraining)

- С 2015 года есть много работ по пред-обучению без меток от людей
  - Обучение на прокси задаче – без дополнительной разметки
    - Полулярный вариант: патчи из одной картинки или нет (+ аугментации)
  - Contrastive loss (x_i – представления нормированные или нет, τ – температура)

$$L_{ij} = -\log\left(\frac{\exp(\vec{x}_i^T \vec{x}_j/\tau)}{\sum_{k\in\text{negatives}}\exp(\vec{x}_i^T \vec{x}_k/\tau)}\right)$$

  - Где брать отрицательные примеры? – случайно или поиск
  - MoCo [He et al.; 2019], SimCLR [Chen et al.; 2020],
  - SOTA: MoCov2 [Chen et al.; 2020], BYOL [Grill et al.; 2020], SimCLRv2 [Chen et al.; 2020]
  - https://lilianweng.github.io/lil-log/2019/11/10/self-supervised-learning.html

- Использование огромных размеченных датасетов тоже развивается
  - Big transfer [Kolesnikov et al.; 2020] - большие ResNet
  - Vision transformer [Dosovitskiy et al.; 2020] - трансформер на патчах

Классификация видео

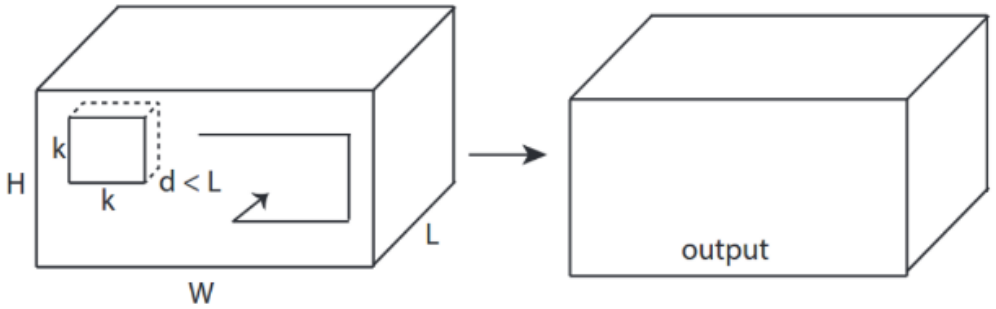
- Задача: распознавание действий на видео

Подходы к видео

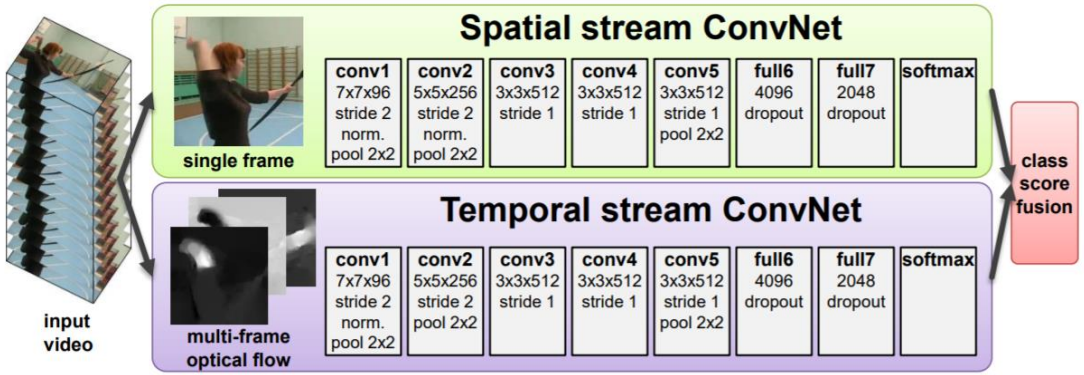
- Задача: распознавание действий на видео
- Подходы:
  - Извлечь CNN признаки и каждого кадра и усреднить
  - Рекуррентная сеть над признаками с кадров [Karpathy et al., 2014]

(часто работает плохо!)

  - 3D свёртки [Tran et al., 2015]



  - Двупоточные сети [Simonyan&Zisserman, 2014]:



Заключение

- Компьютерное зрение активно использует нейросети
  - Область фрагментирована по задачам
  - Есть задачи зрения, где нейросети работают хуже
- Одна из самых вычислительно тяжелых областей
  - Можно и нужно использовать готовые веса!
- Много специализированных курсов и ресурсов