# Свёрточные сети в задачах компьютерного зрения

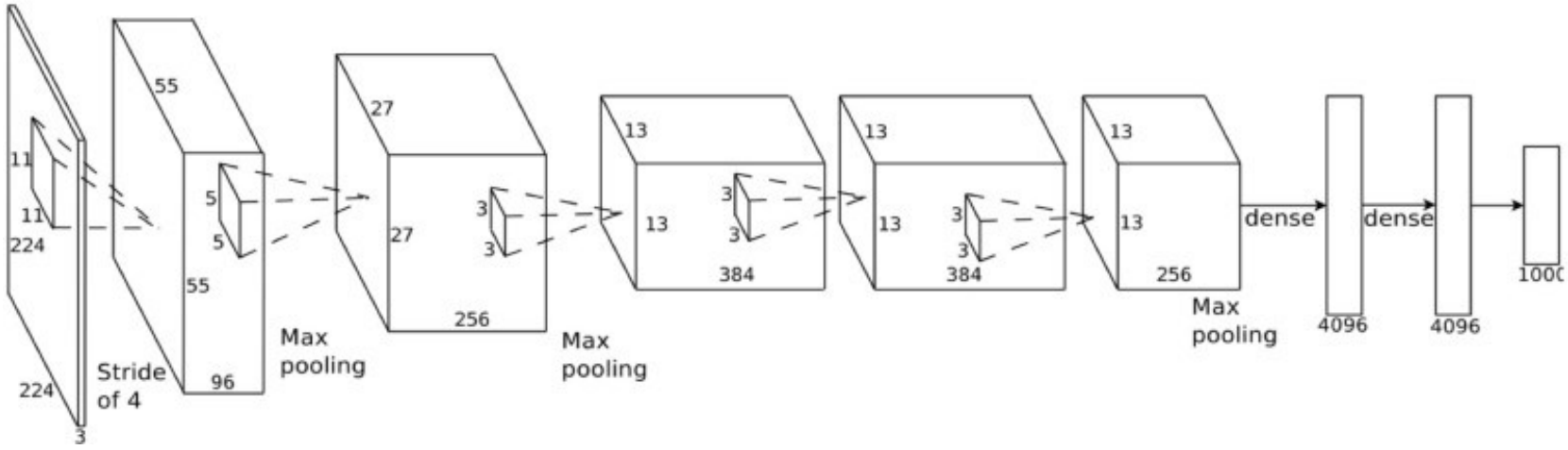
Антон Осокин

ФКН ВШЭ

29.09.2017


НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

# Recap: классификация




- Задача классификации изображение решена! (почти)
- Вход сети – изображение
- Выходы сети соответствуют классам
- Кросс-энтропия для обучения
- Много архитектур сетей (например, ResNet)
- Блок свёрточных слоев в начале сети
- Идея – переиспользовать выученные представления

# План лекции

- Детекция объектов
  - R-CNN, Fast R-CNN, Region Proposal Networks
  - Fast detectors: SSD and YOLO
- Сегментация изображений
  - Fully convolutional networks
  - CRF as RNN
  - Masked R-CNN
- Поиск похожих изображений
  - Siamese architecture
  - Отслеживание объектов на видео
- Распознавание действий на видео

# Часть 1: детекция объектов

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

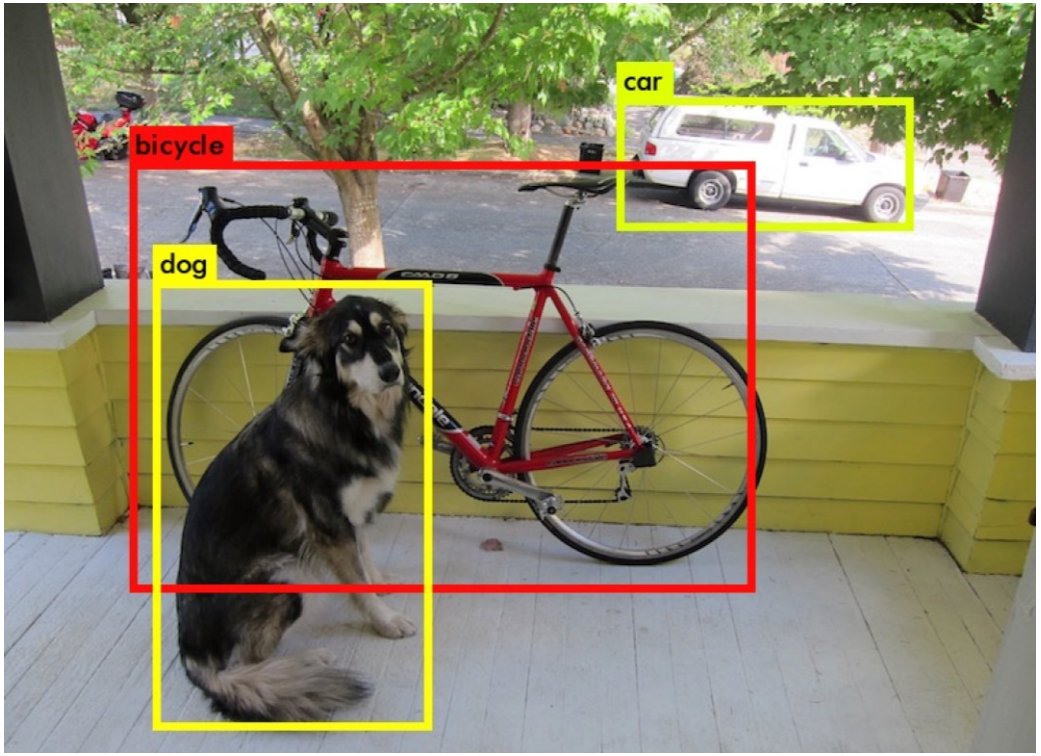
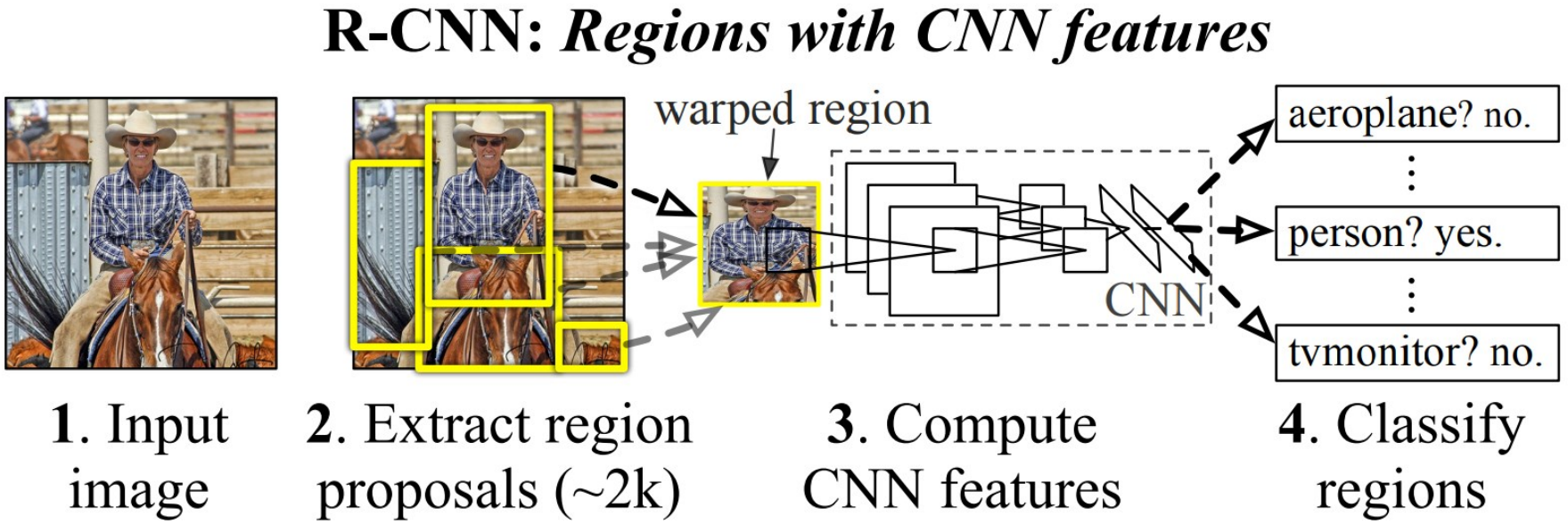



image credit: Joseph Redmon

# Ранние методы: R-CNN

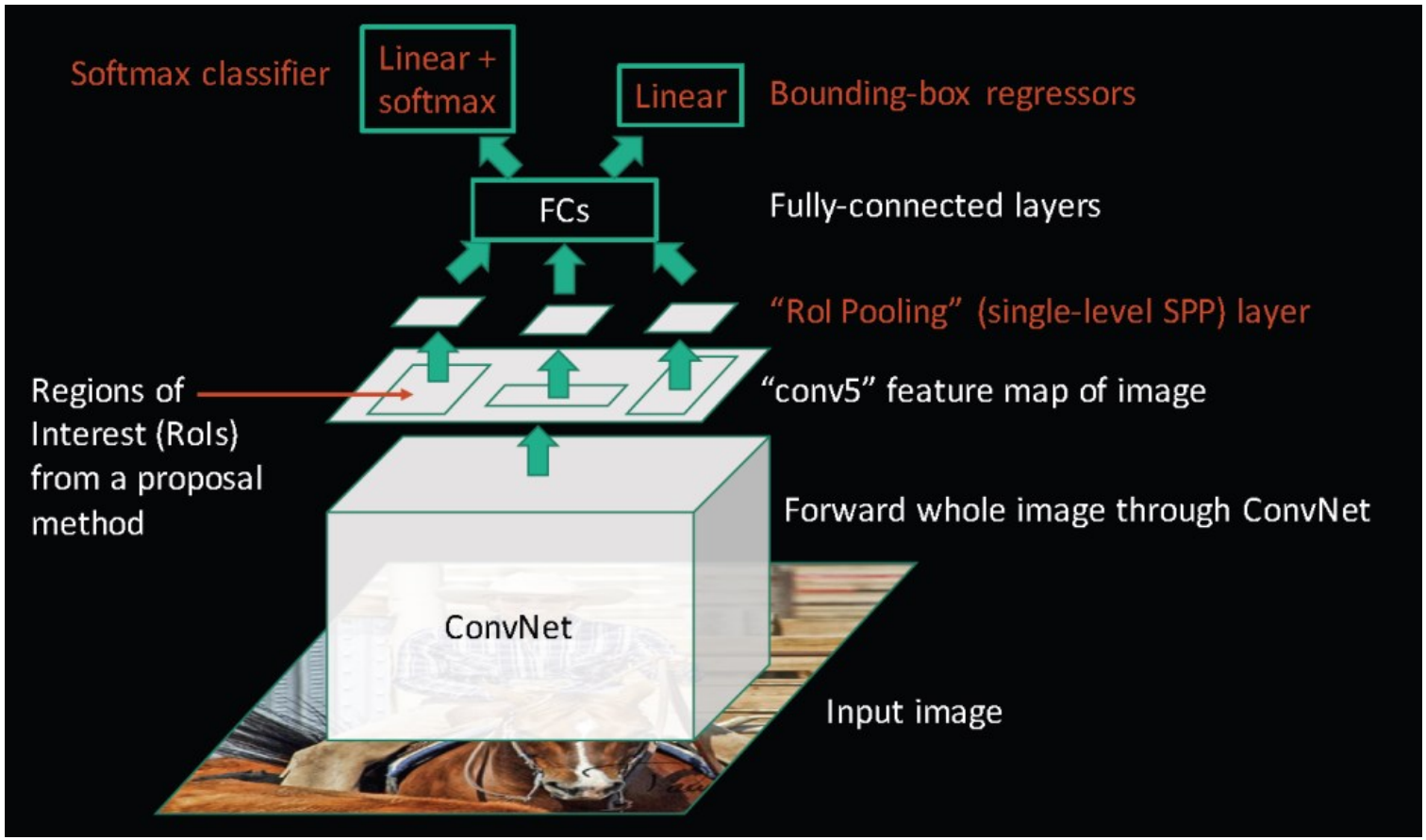



- Основная идея – классифицировать гипотезы (object proposals)
- Используем CNN для каждой гипотезы
- На выходе: метка класса и уточнение позиции объекта

[Girshick et al., 2013]

# Fast R-CNN

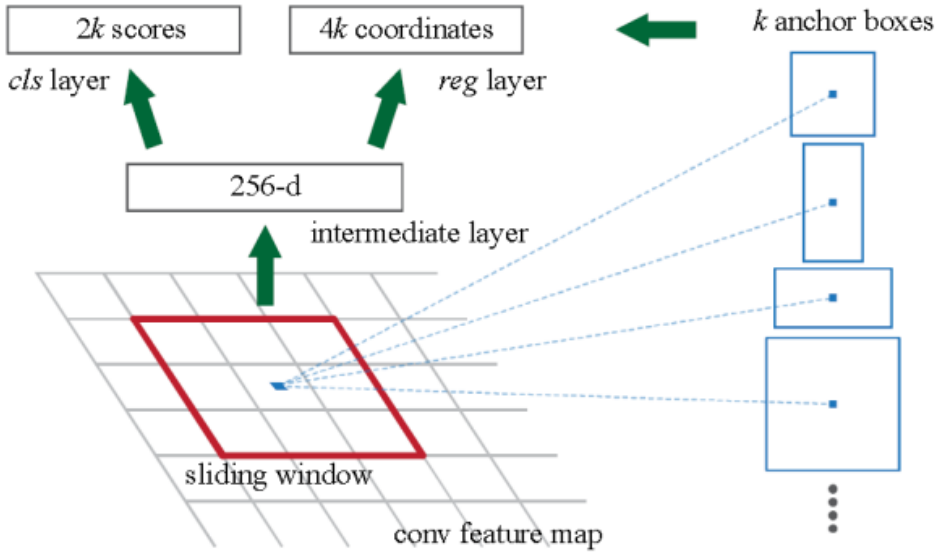
- Недостаток R-CNN – медленная скорость работы
- Много пересекающихся гипотез – неэффективно
- Идея: разделить вычисления свёрток между гипотезами

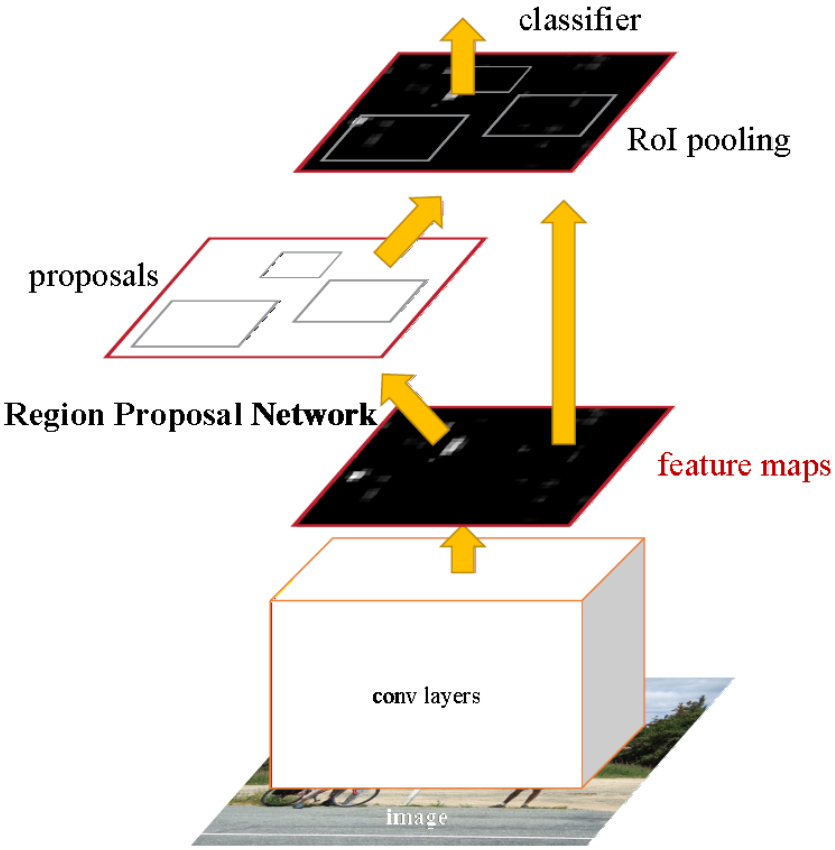



[Girshick 2015]

# Region proposal network

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

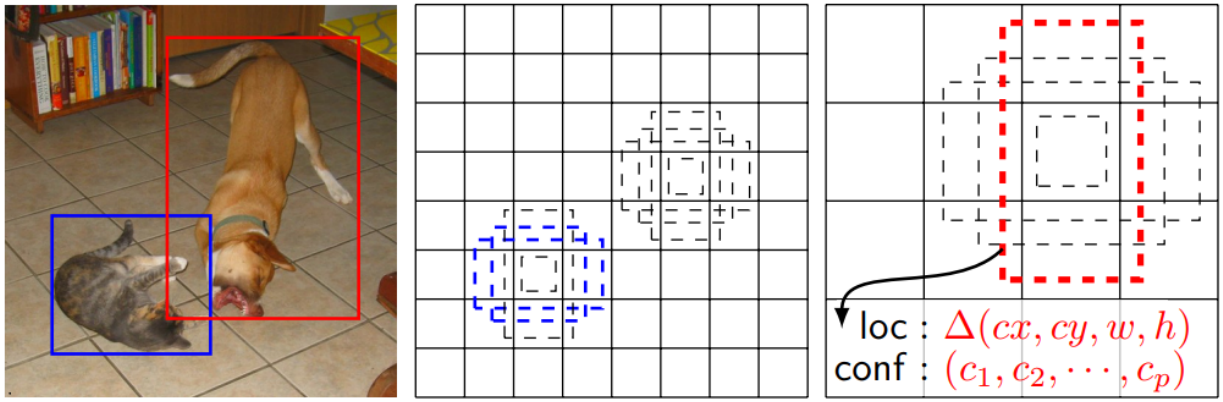


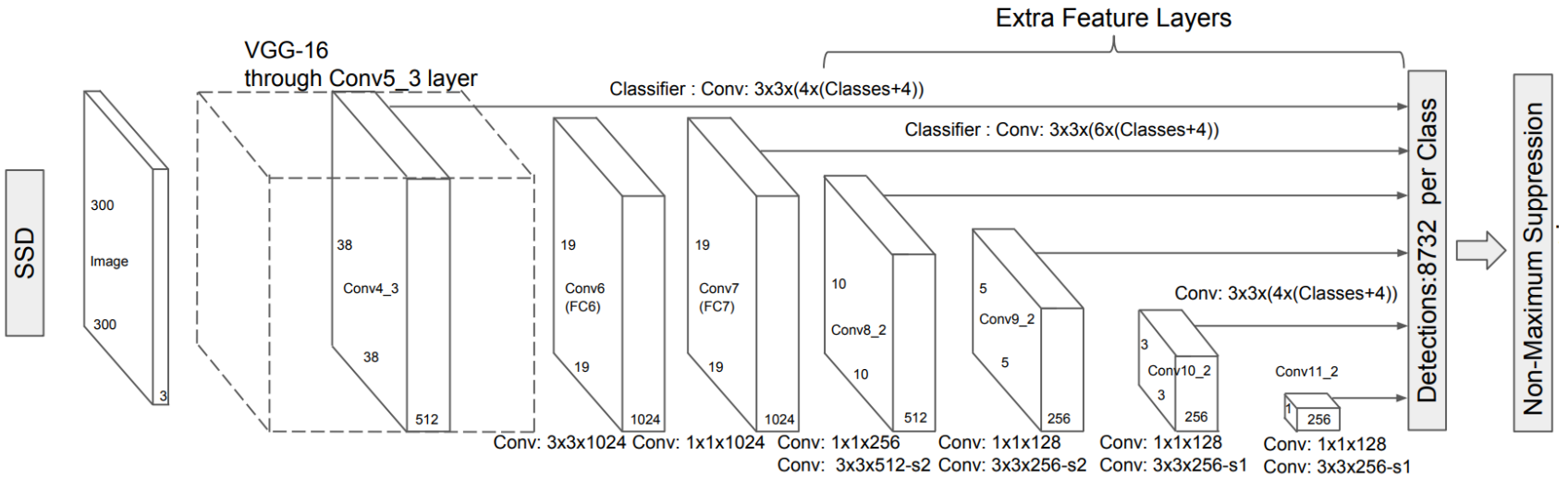

[Ren et al., 2015]

# Fast detectors: YOLO, SSD, YOLOv2

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




(a) Image with GT boxes (b) $8 \times 8$ feature map (c) $4 \times 4$ feature map



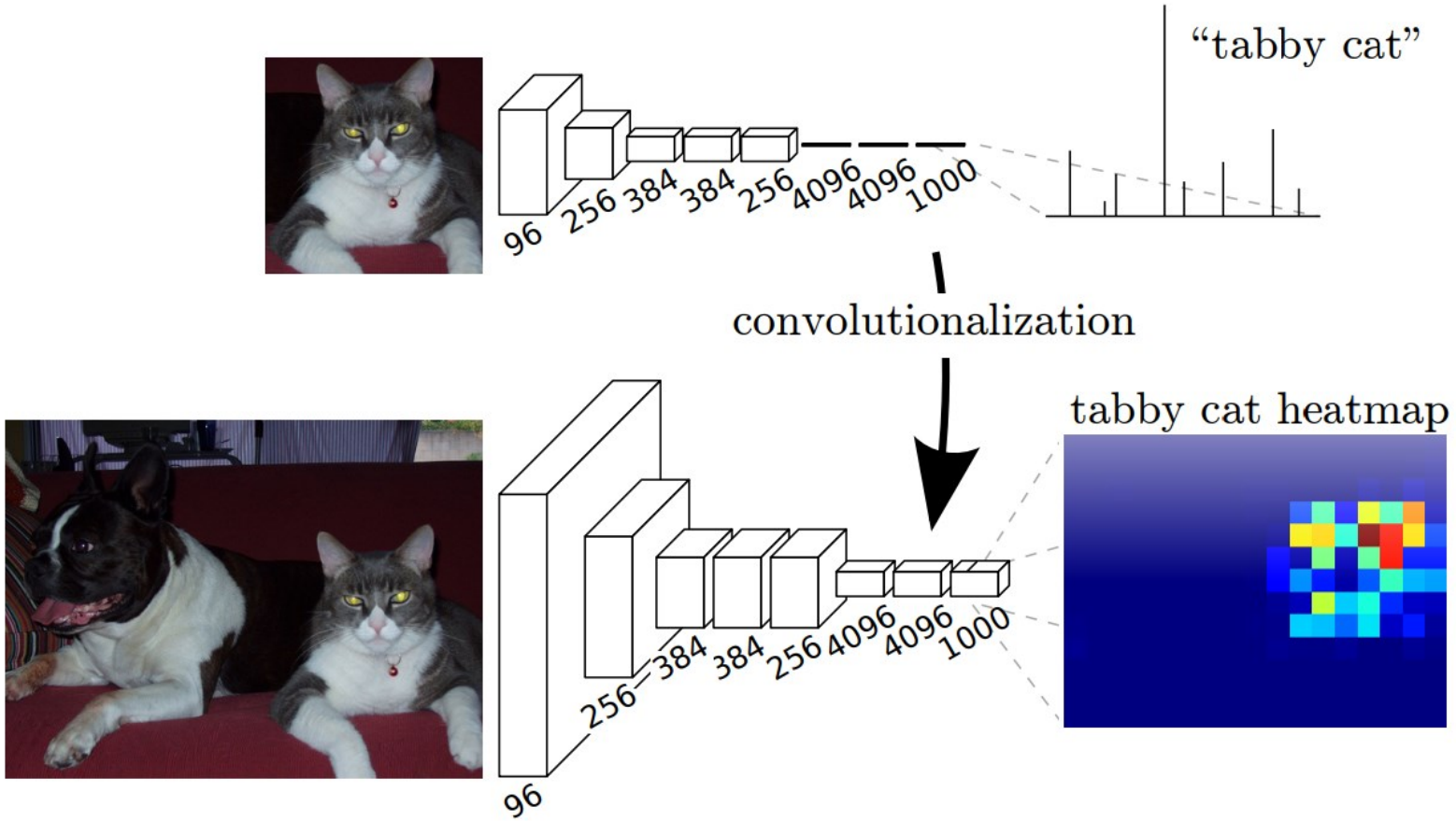

[Liu et al., 2016]

# Часть 2: сегментация

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

# Fully-convolutional CNN

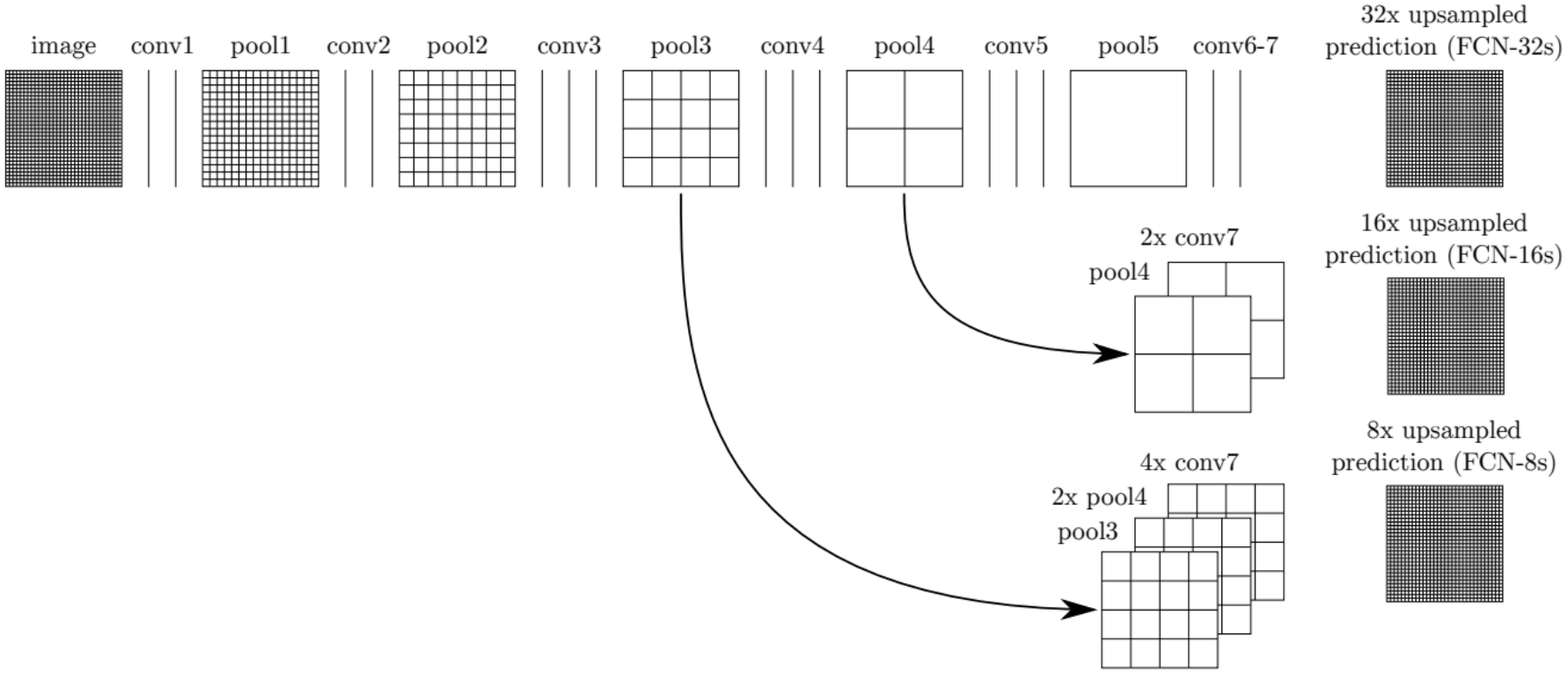
- Идея: применить CNN скользящим окном
- Недостаток – очень низкое разрешение выхода




Идея из 90-х, [Long et al., 2015]

# Fully-convolutional CNN

- Идея: применить CNN скользящим окном
- Недостаток – очень низкое разрешение выхода
- Идея: разрешение с помощью более глубоких слоев
- Используются upconv, dilated conv, etc.




[Long et al., 2015]

# CRF поверх CNN

- Идея: добавить локальный обмен информацией из CRF
- Представить передачу сообщений как вычислительный граф
- Обучать совместно

- Модель Dense CRF [Krahenbuhl&Koltun, NIPS 2011, ICML 2013]

$$E(\mathbf{x}) = \sum_i \psi_u(x_i) + \sum_{i<j} \psi_p(x_i, x_j), \;\; \psi_p(x_i, x_j) = \mu(x_i, x_j) \sum_{m=1}^{M} w^{(m)} k_G^{(m)}(\mathbf{f}_i, \mathbf{f}_j),$$

- Вид парных потенциалов ограничен (RBF ядра)
- Возможен эффективный вар. вывод (параллельный!)

$$Q_i(x_i = l) = \frac{1}{Z_i} \exp\left\{ -\psi_u(x_i) - \sum_{l' \in \mathcal{L}} \mu(l, l') \sum_{m=1}^{K} w^{(m)} \sum_{j \neq i} k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) Q_j(l') \right\}$$

[Zheng et al., 2015]

# CRF поверх CNN

- Возможен эффективный вар. вывод (параллельный!)

$$Q_i(x_i = l) = \frac{1}{Z_i} \exp\left\{ -\psi_u(x_i) - \sum_{l' \in \mathcal{L}} \mu(l, l') \sum_{m=1}^{K} w^{(m)} \sum_{j \neq i} k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) Q_j(l') \right\}$$

**Algorithm 1** Mean-field in dense CRFs [29], broken down to common CNN operations.

$Q_i(l) \leftarrow \frac{1}{Z_i} \exp\left(U_i(l)\right)$ for all $i$ ▷ Initialization

**while** not converged **do**

$\tilde{Q}_i^{(m)}(l) \leftarrow \sum_{j \neq i} k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) Q_j(l)$ for all $m$ ▷ Message Passing

$\check{Q}_i(l) \leftarrow \sum_m w^{(m)} \tilde{Q}_i^{(m)}(l)$ ▷ Weighting Filter Outputs

$\hat{Q}_i(l) \leftarrow \sum_{l' \in \mathcal{L}} \mu(l, l') \check{Q}_i(l')$ ▷ Compatibility Transform

$\breve{Q}_i(l) \leftarrow U_i(l) - \hat{Q}_i(l)$ ▷ Adding Unary Potentials

$Q_i \leftarrow \frac{1}{Z_i} \exp\left(\breve{Q}_i(l)\right)$ ▷ Normalizing

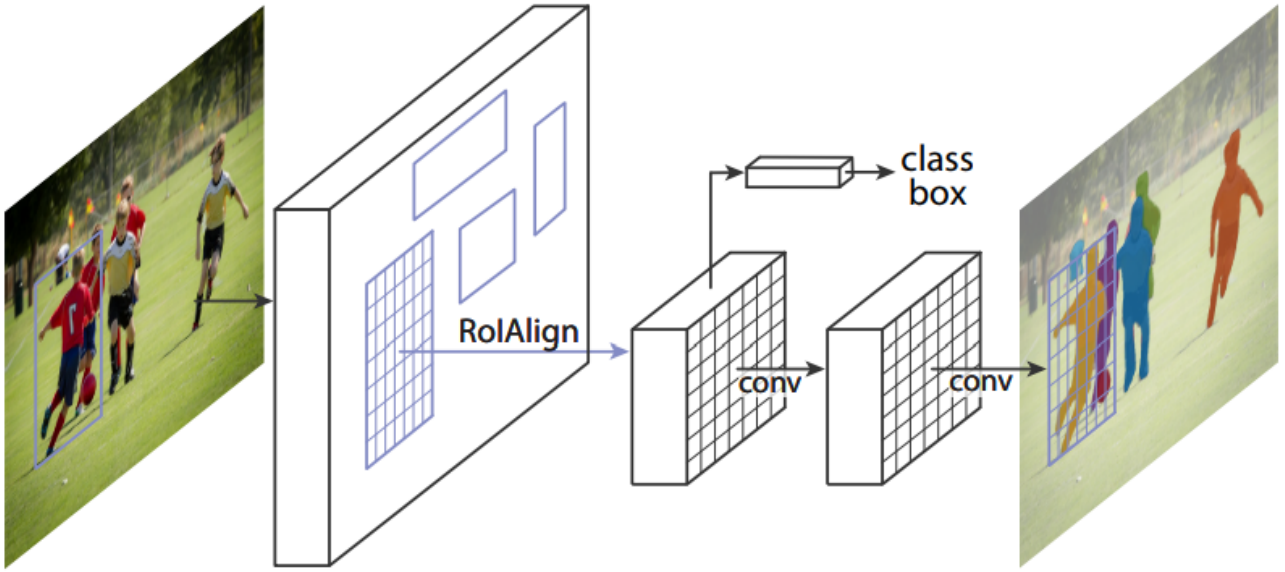
**end while**

[Zheng et al., 2015]

# CRF поверх CNN

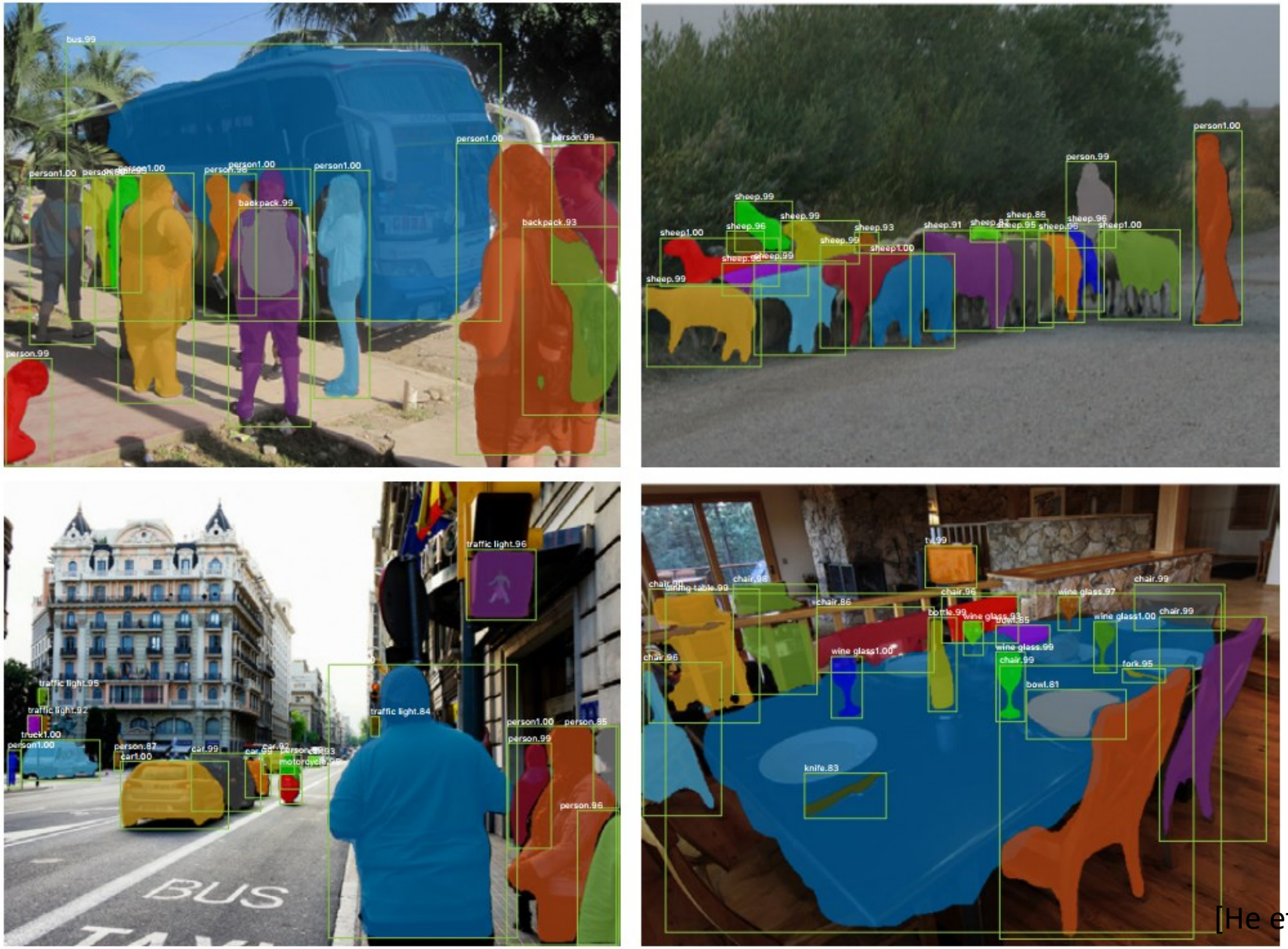
# Сегментация объектов: Mask R-CNN

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




[He et al., 2017]

# Сегментация объектов: Mask R-CNN




[He et al., 2017]

# Часть 3: 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!
- Не всегда SOTA

# Сиамские сети (siamese)

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

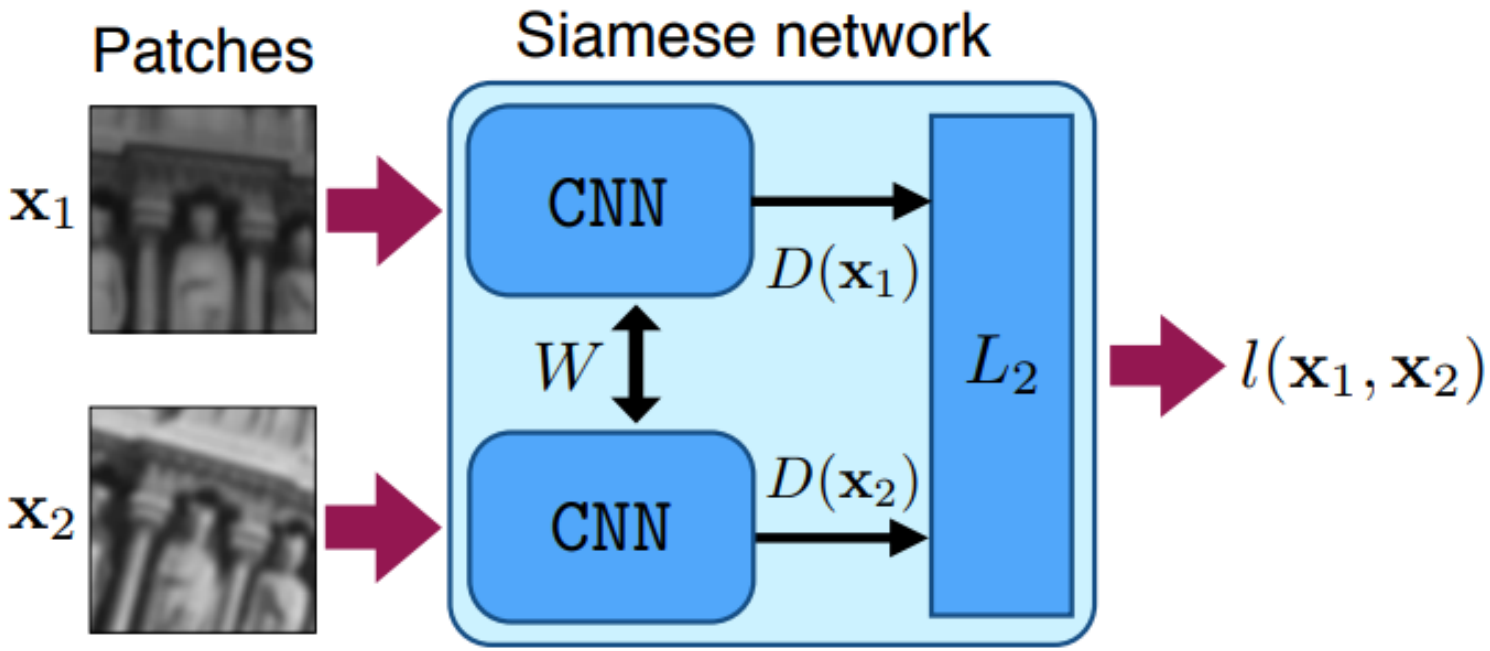
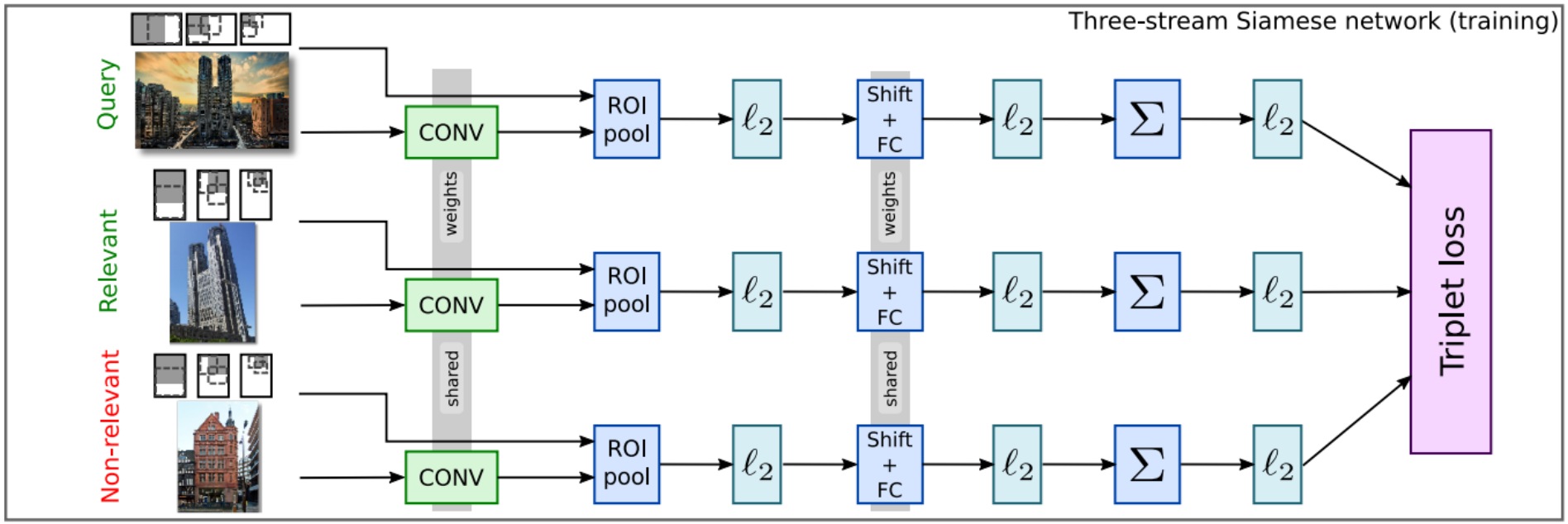
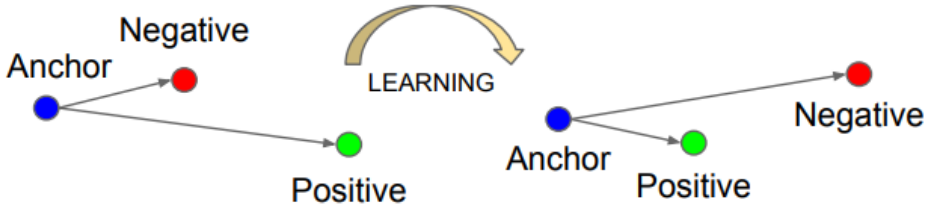



Image from [Simo-Serra et al., 2015]

# Сиамские сети (siamese)

- Идея: использовать одну и ту же сеть на двух изображениях, и считать расстояние между признаками



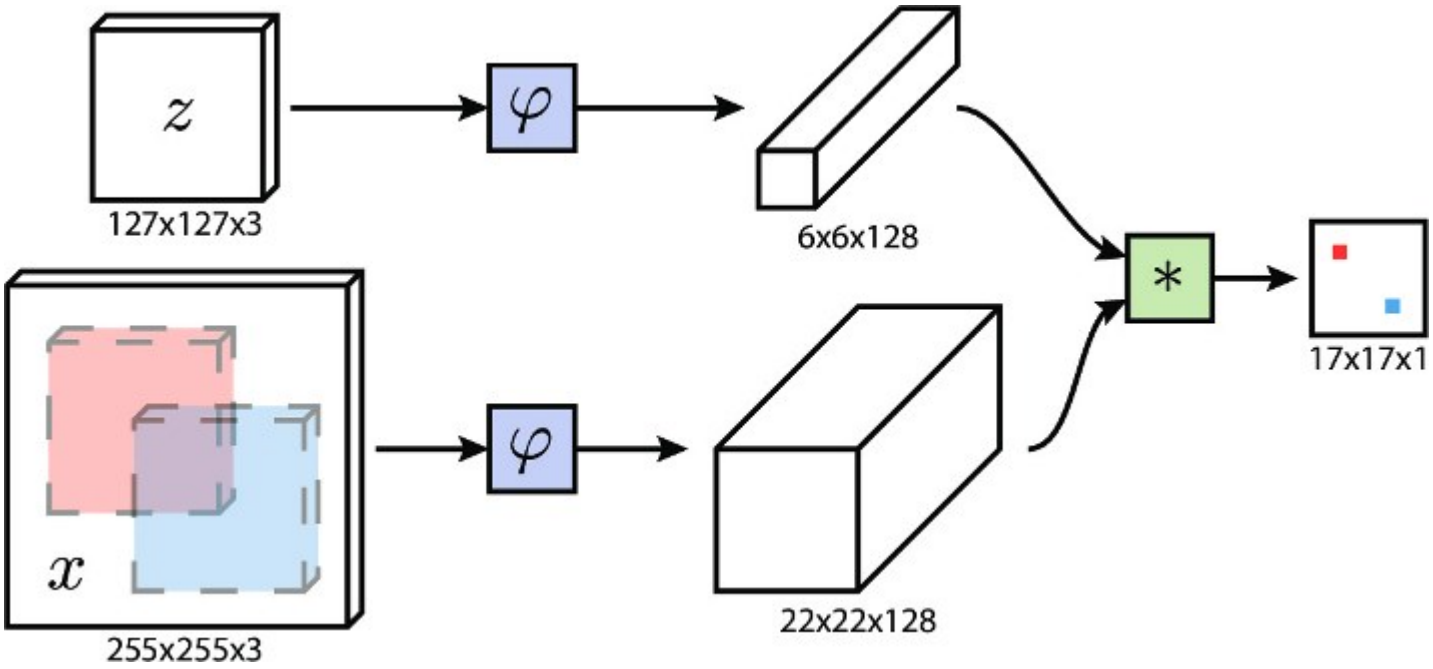




$$\max(0, m + \|q - d^+\|^2 - \|q - d^-\|^2)$$

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

# Отслеживание объектов на видео

- Идея: одну из веток сиамский сетей применять свёрточно




[Bertinetto et al., 2016]

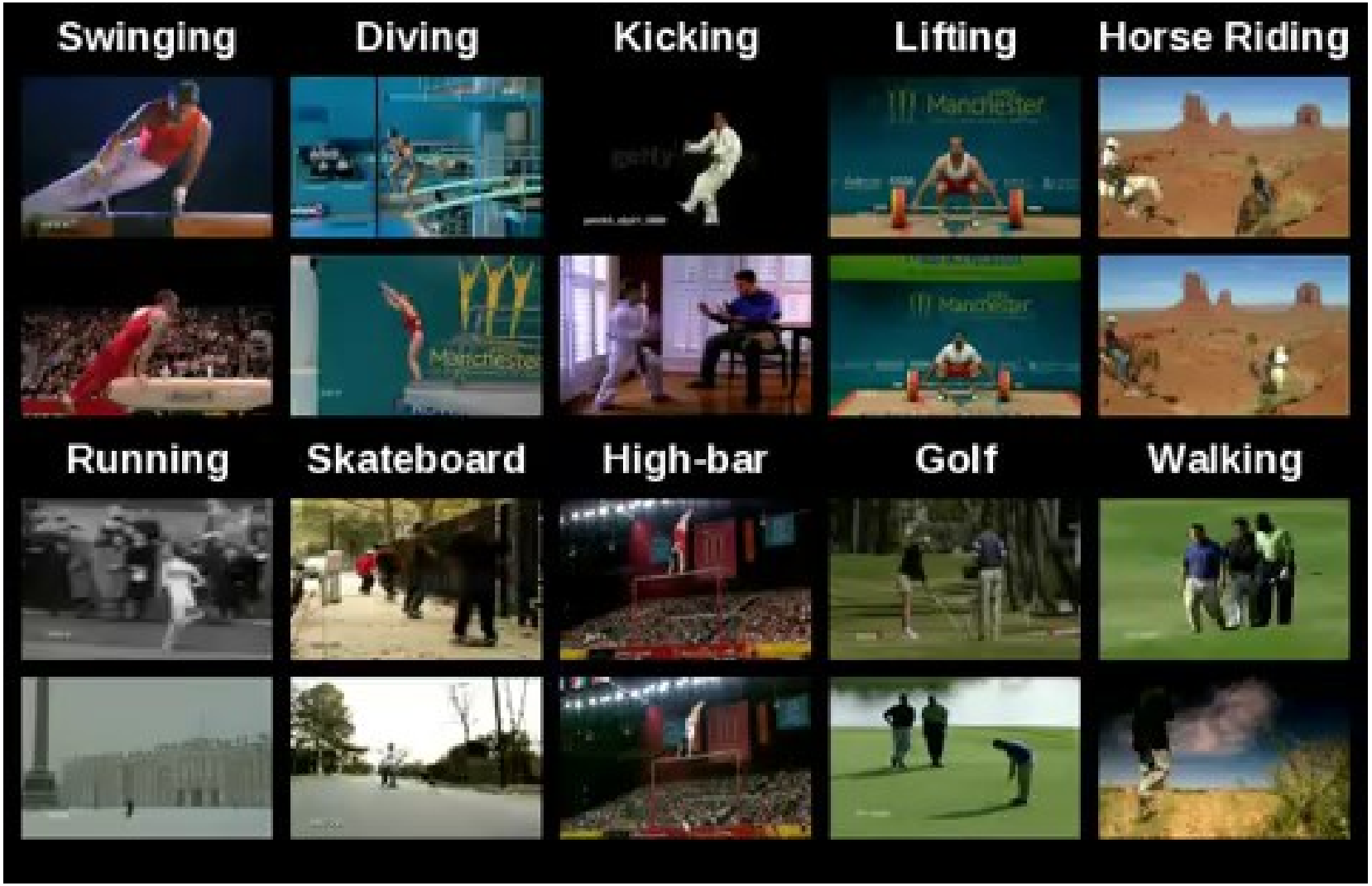
# Отслеживание объектов на видео

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


[Bertinetto et al., 2016]

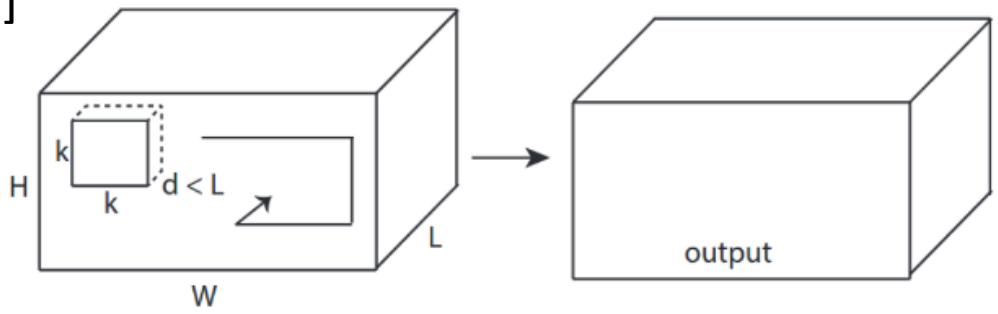
# Часть 4: классификация видео

- Задача: распознавание действий на видео

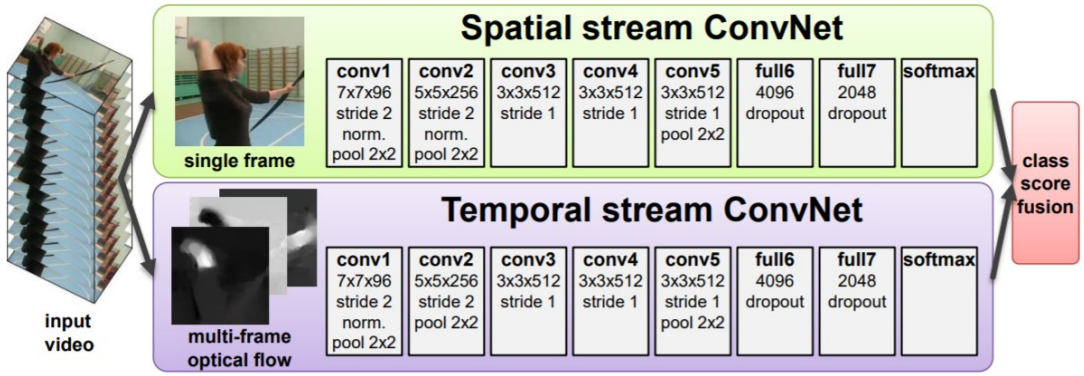


# Подходы к видео

- Задача: распознавание действий на видео
- Подходы:
  - Извлечь CNN признаки и каждого кадра и усреднить
  - Рекуррентная сеть над признаками с кадров [Karpathy et al., 2014]
  - 3D свёртки [Tran et al., 2015]




  - Двупоточные сети [Simonyan&Zisserman, 2014]:



# Заключение

- Компьютерное зрение активно использует нейросети
- Есть задачи зрения, где нейросети не работают
- Очень большая область
- Одна из самых вычислительно тяжелых областей
- Много специализированных курсов