

## Лекция 5. Computer Vision: детекция и семантическая сегментация

Как решать задачу "Где изображено?"

- Bounding box - прямоугольник  $(x_1, y_1, x_2, y_2)$ , в котором вложены объекты. Или же вектор другого параметризации:  $(x_c, y_c, w, h)$
- Выделить те, и только те пиксели, которые относятся к объекту

Постановки задач.

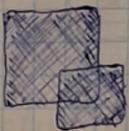
- Локализация: предсказание метки + bounding box.
- Детекция: выделение ограничивающих рамок для нескольких объектов + классификация объектов
- Рекурсивная сегментация: сопоставление каждого пикселя метки принадлежности к определ. категории.

Задача семантической сегментации изображения с дифференцированными объектами называется задачей сегментации экземпляров. (instance segmentation)

## Метрики качества

### Intersection over Union (IoU)

$$IoU = \frac{\text{Area of intersection}}{\text{Area of union}}$$



$IoU \in [0, 1]$ , где 0 соответствует отсутствию пересечения между предиктом и меткой, а 1 - полностью пересечено.

Проводя аналогии с метриками бинарной классификации, можно вычислить precision и recall, где

- True Positive (TP) -  $IoU > \text{threshold}$
  - False Positive (FP) -  $IoU < \text{threshold}$  или дублированной bounding box для одного и того же объекта
  - False Negative (FN) - объект не найден или уверенность  $(IoU < \text{threshold})$  слишком низкая
- средний AP (average precision)

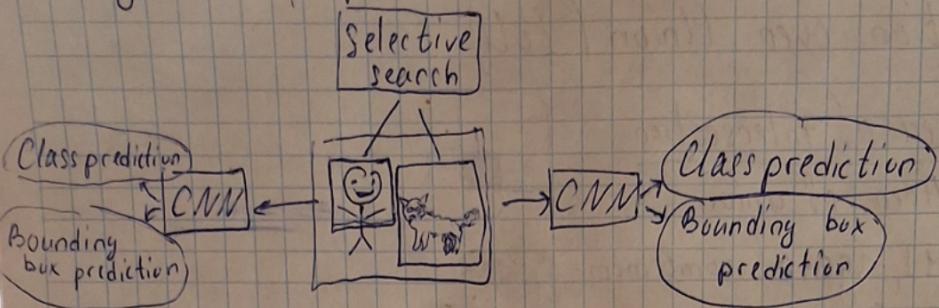
$$mAP = \frac{1}{K} \sum_{i=1}^K AP_i;$$

## R-CNN

Идея: двухэтапный подход

- Первый этап: выделение подозрительных на наличие объектов областей (RoI - Regions of Interest)
- Второй этап: подать выделенные на пред. этапе области в сеть для классификации и уточнение границ bounding box'а

1. Input image
2. Extract region proposals ( $\sim 2k$ )
3. Compute own features
4. Classify regions



Метод Выбора ROI - конструктивно-эвристический Selective Search

### Замечания

- Архитектура. CNN в этом подходе используется исключительно как feature extractor (2013 год нейросети еще не доверяли полное решение задачи).

На полученных ее признаках отдельно обучалась SVM и отдельно линейная регрессионная модель для уточнения ограничивающей рамки

- Допренировка (Fine tune) Выбранную претренированную модель допренимали на конкретном датасете след. образом.

- Для каждого изображения генерировали множество с помощью selective search.

- Считали, что в прямоугольнике изображены объекты, если их соотношение  $IoU > 0,3$  с некоторым объектом в таргете разметке

- Брали за окно с различными объектами чье окно без объекта, т.о. решалась задача классификации с K+1 классом (+ фон)

### Уточнение границ

На bounding box'ах с  $IoU > 0,3$  для каждого блока обучалась линейная модель на CNN признаках, удаляя значения:

$$\left( \frac{x - \hat{x}}{w}, \frac{y - \hat{y}}{h}, \log \frac{\hat{w}}{w}, \log \frac{\hat{h}}{h} \right)$$

не абсолютные коорд, а относит. поправки

где величины с краинкой соответствуют предложенной ограничивающей рамке из selective search

→ Почему  $\log \frac{\cdot}{\cdot}$ ? Ошибка в один пиксель на изначальном изображении не совпадала с этой же ошибкой в случае большого изображения

$\hat{x}, \hat{y} \dots$  — параметры региона от Selective Search (центр, размер)  
 $x, y \dots$  — исходный bounding box

Проблема: Selective search порождает много регионов, из-за чего один и тот же объект может быть покрыт несколькими ROI, для каждого из которых модель даст прав. ответ

Решение: Non Maximum Suppression (NMS).

Берём все bounding box. Упорядочиваем по IoU, по степени уверенности классификатора. Если на последующем шаге имеется значительное пересечение с прошлым, то он удаляется

о-

и  
и  
и

и  
и  
и

и  
и  
и

и  
и  
и

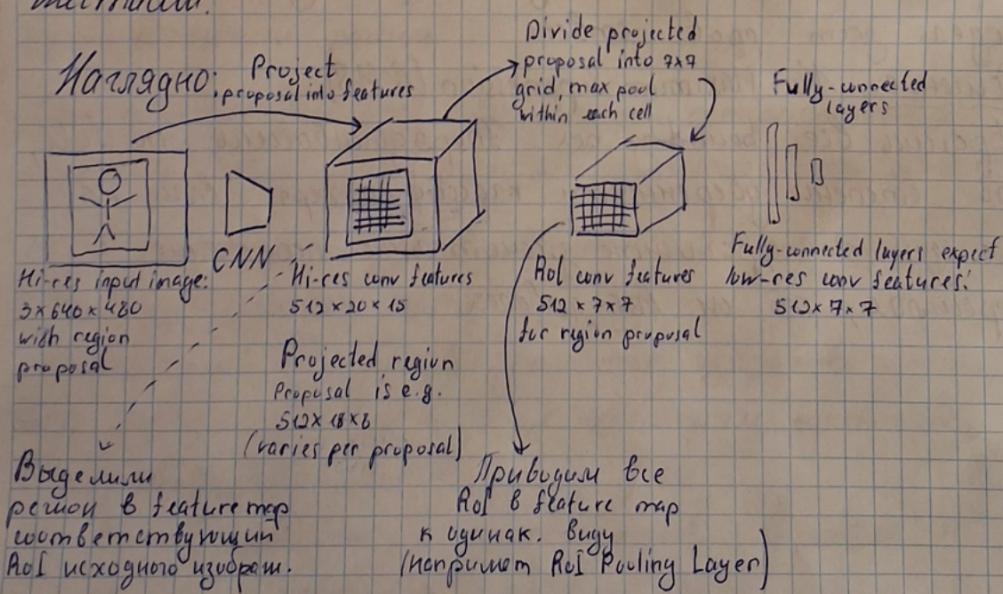
## Недостатки R-CNN

- Большое время работы (каждый ROI подается на вход в сеть)
- Много подозрительных и при этом бесполезных ROI
- Не адаптирующий под данные алгоритм выбора ROI
- Решение

## Fast R-CNN

Ключевой недостаток R-CNN заключается в спорности работы. Бумажными горячками являются запуск полного алг. классиф. для всех кандидатов.

Метод. Ворезать подозрительные на объект региона не из исходного изображения, а из карто признаков, которая строится только один раз для всего входного изображения всеми. Извлеченные слова или перевести в "crop" и "feature extractor" местами.



## Faster R-CNN

Проблема: Selective search не учитывает специфику данных. Хотелось бы решать задачи регионов интересов также с помощью нейронки.

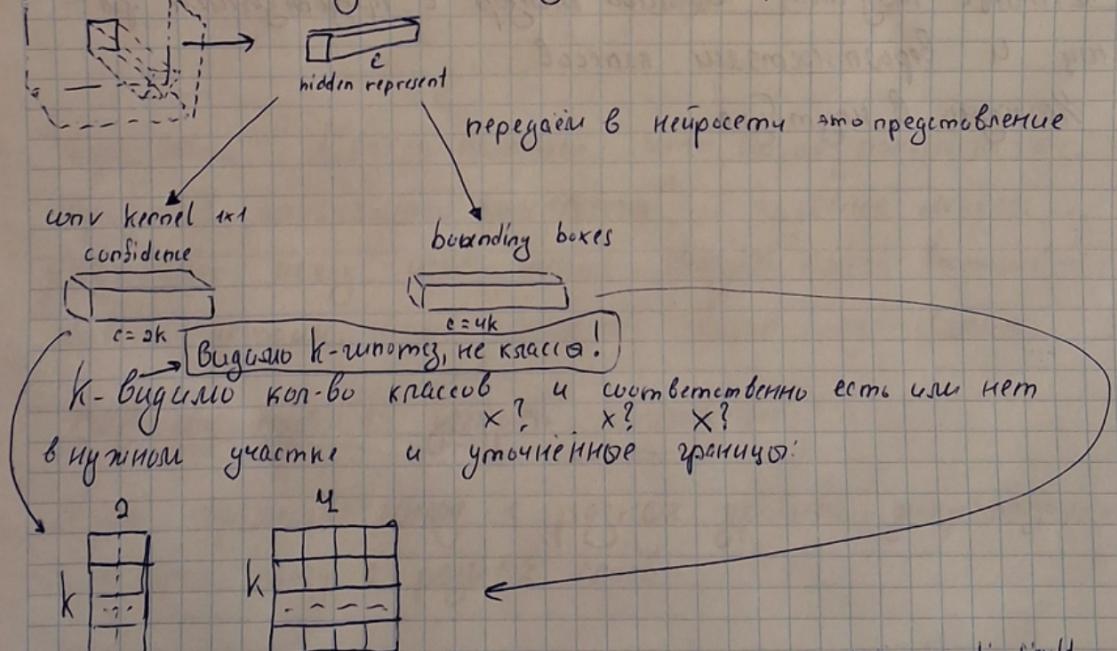
Идея: RPN (Region Proposal Network)

Также как раньше получаем feature map. Дальше хотим предложить регионы интереса.

Уз  $3 \times H \times W$  получаем  $C \times \frac{H}{16} \times \frac{W}{16}$ , где каждый пиксель имеет большое receptive field ~ примерно всё изображение.

Каждый пиксель отвечает за регион, который он предполагает.

Если зафиксировать пиксель и взять все классы, совместно. если, то получим вектор.

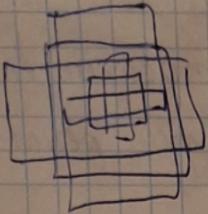


То есть пиксель <sup>после нейронки MLP</sup> отвечает на вопрос - есть ли во всей ~~фотографии~~ <sup>объект</sup>, его центр и определенные размеры например, и какой <sup>есть</sup> его границы

Пиксель предполагает к чипомез

максимальное receptive field

Такие применяются через API



И нужно было бы просто дополнить эти области, а не предсказывать с нуля?

YOLO

Идея: использовать one-stage (или proposal-free) метод.  
Изучи словари, обучи сеть, которая ведёт сразу  
(за один просмотр изображения) и огранич. рамки, ч  
классифицирует на них. Без всяких ROI

В конце получали общий тендер с предсказанием для  
ищ и вероятностями классов.

Нападки в подтубке

## Рекурсивная сегментация

Задача в том, чтобы дать штамп блеска пикселя

Ключевое — сворачиваем исходное изображение feature map. После этого мы обратно разворачиваем. То есть если в приложении для "мы сдали", хотели понять где оно на изображении. Для этого применяется Transposed convolution

Удобно понять через произв. матрицы на вектор

В квадратике

$$\begin{matrix} \square \\ \square \end{matrix} \rightarrow \square \cdot \boxed{\square} + \square \cdot \boxed{\square} +$$

kernel

$$+ \square \cdot \boxed{\square} + \square \cdot \boxed{\square}$$

Проблема: Модель помнит что, но не помнит где.  
Проблема "избыточного контекста", в глубинах модель забывает исход. изображение. Модель плохо воспринимает детали, лишь какие-то общие представления

Как бороться с утерей контекста.

Нужно хранить информацию с разных уровней декомпозиции  
Объединять информацию с разных слоёв

U-Net

~~Decoder~~

LinkNet

Собираем сущность и уточнения геометрического профилей  
Грамотно реализовать прошлые циклы

