

16 – 18.03

2018

Open Data
Science



Whatever

Хакатон по
машинному обучению

Hack

Задача
"2 стула"

Обзор решения
1ого места



Войти

Dbrain

@mail.ru
group

LeaderGPU

Telegram чат

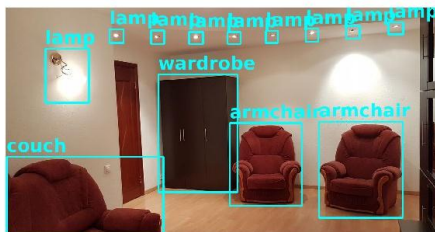
Задача

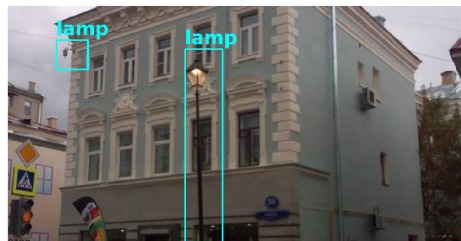
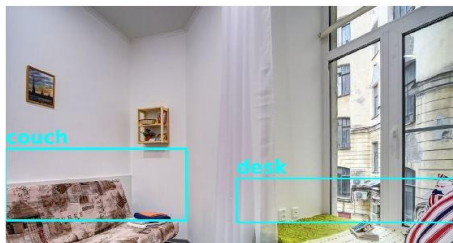
Детектирование мебели на 9 классах:

- 1) dining_table,
- 2) shelf,
- 3) armchair,
- 4) lamp,
- 5) couch,
- 6) desk,
- 7) chair,
- 8) bed,
- 9) wardrobe.

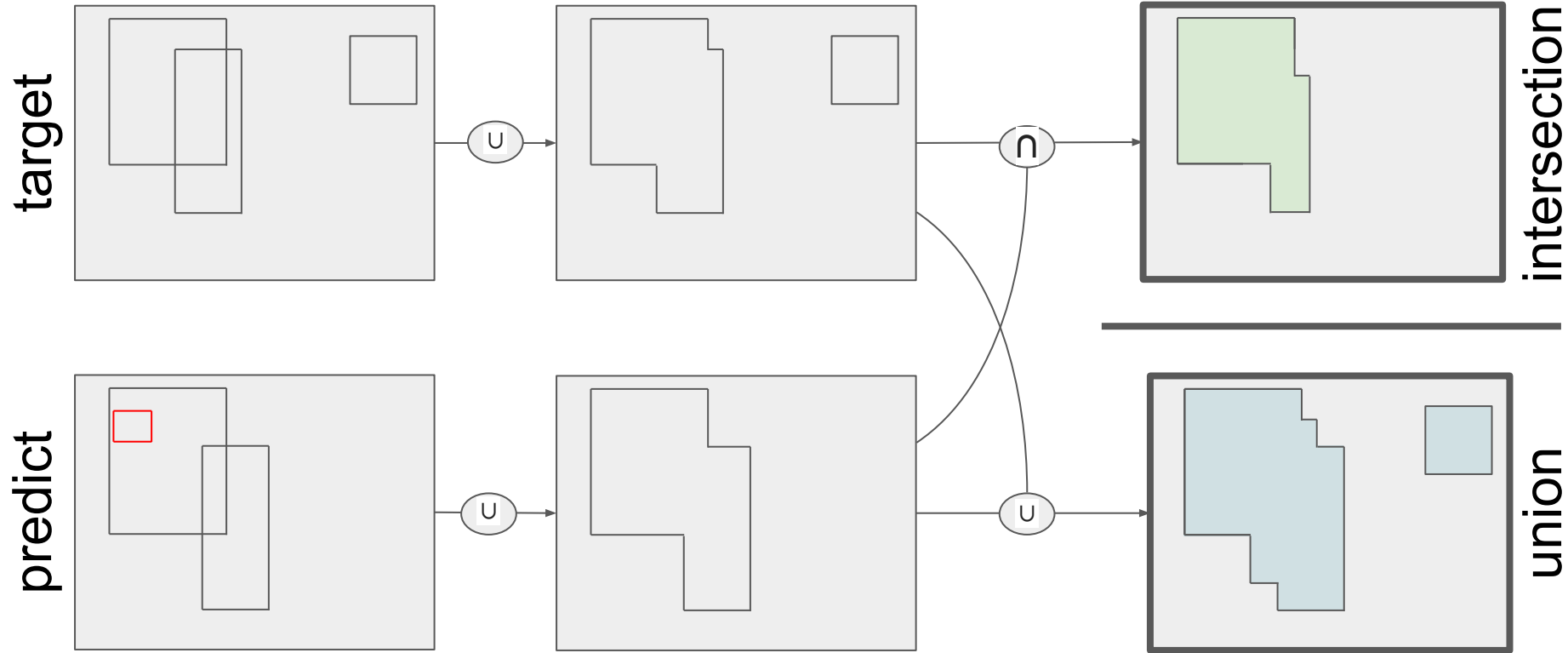
Данные

Размеры train \ test: 7152 \ 3713 картинок





Метрика



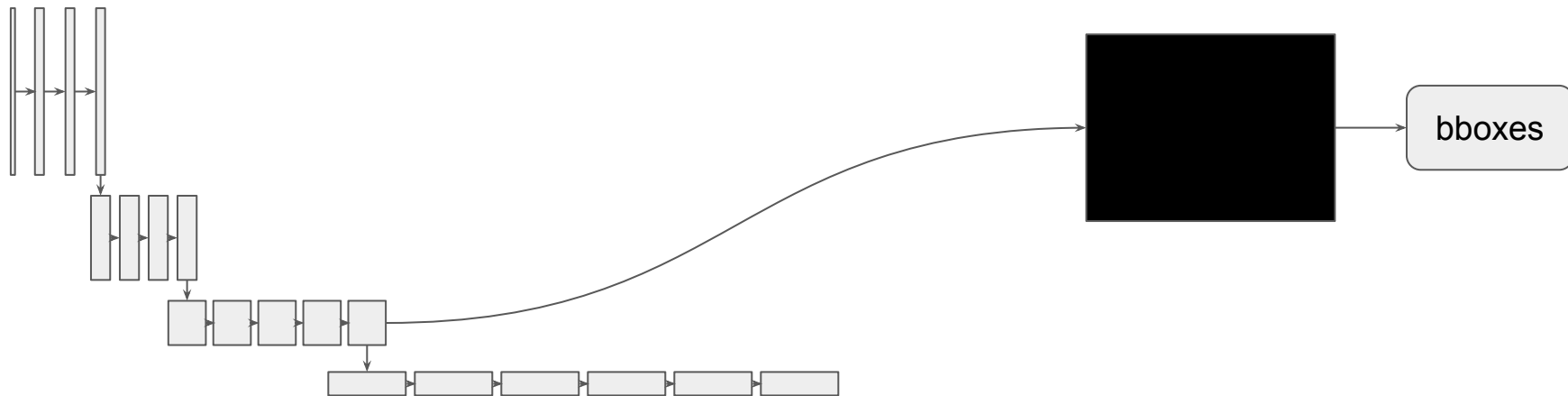
Детекторы

- YOLO
- SSD
- FasterRCNN
- MaskRCNN

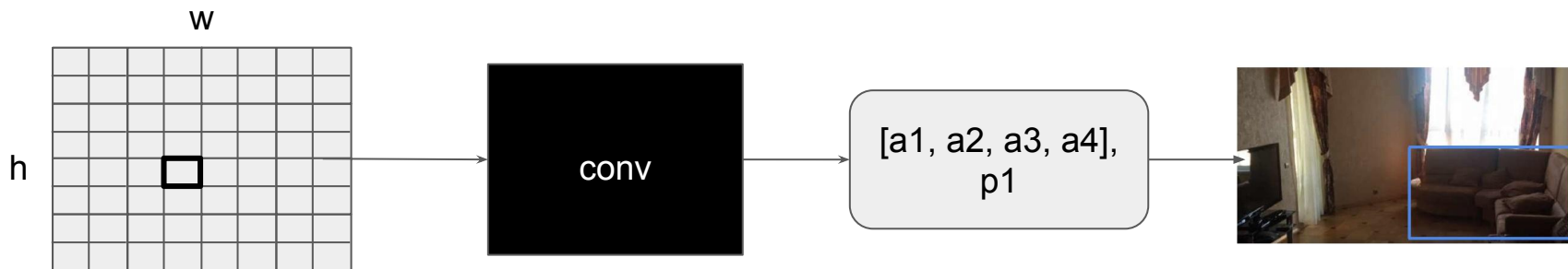
Общее у детекторов

feature extractor

bboxer



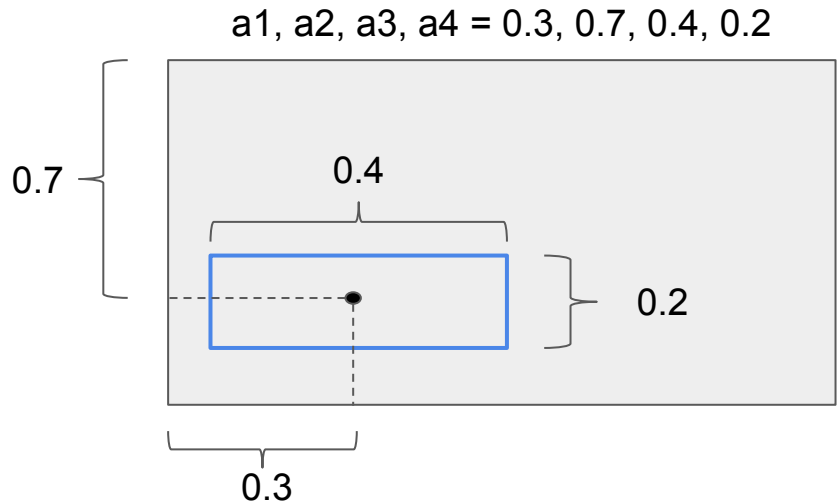
bboxer



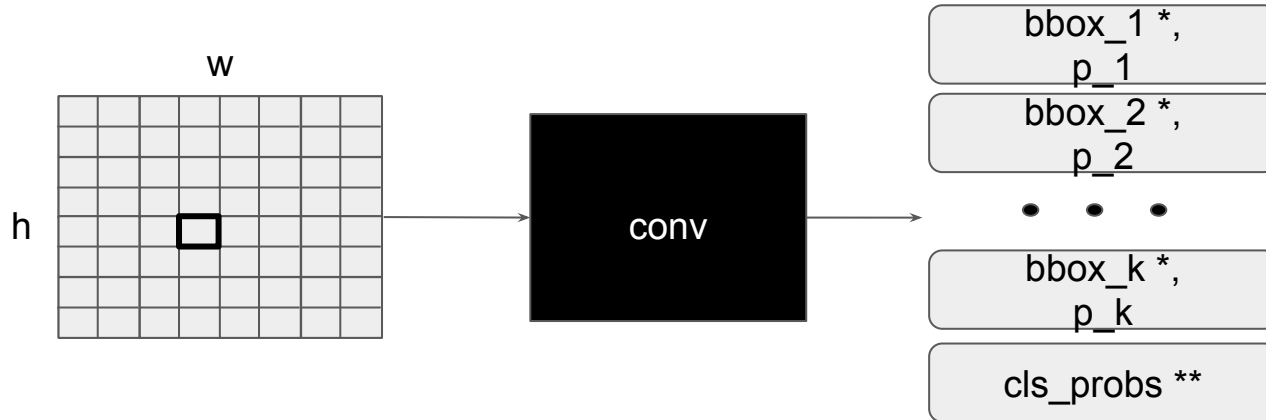
bboxer в YOLO

Кодирование

$a1 = x_center$
 $a2 = y_center$
 $a3 = width$
 $a4 = height$
 $a1, a2, a3, a4 \in [0,1]$

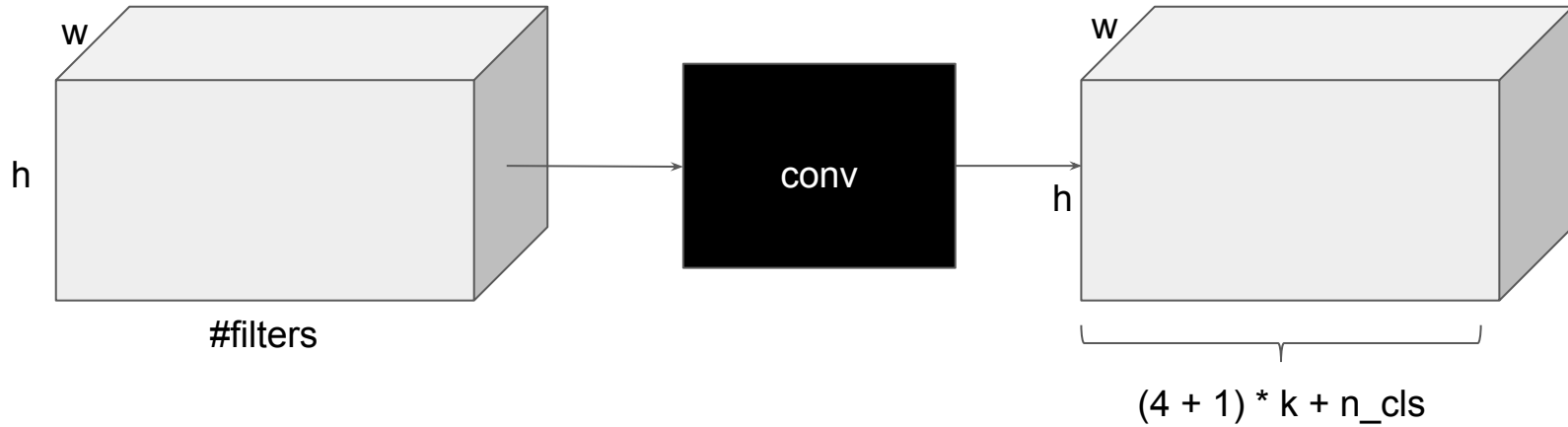


bboxer в YOLO



* $\text{bbox_}^* = [a_1, a_2, a_3, a_4]$,
** $\text{cls_probs} = [\text{cls_1}, \text{cls_2}, \dots, \text{cls_n}]$

bboxer в YOLO



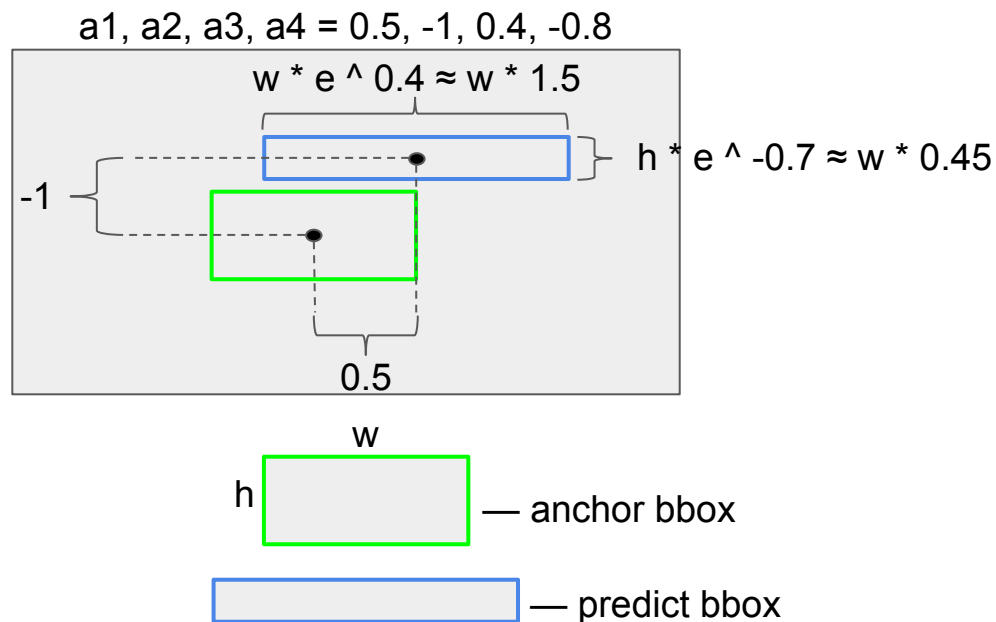
bboxer в SSD

Кодирование

$a1 = \text{shift_x_center}$
 $a2 = \text{shift_y_center}$
 $a3 = \text{scale_width}$
 $a4 = \text{scale_height}$

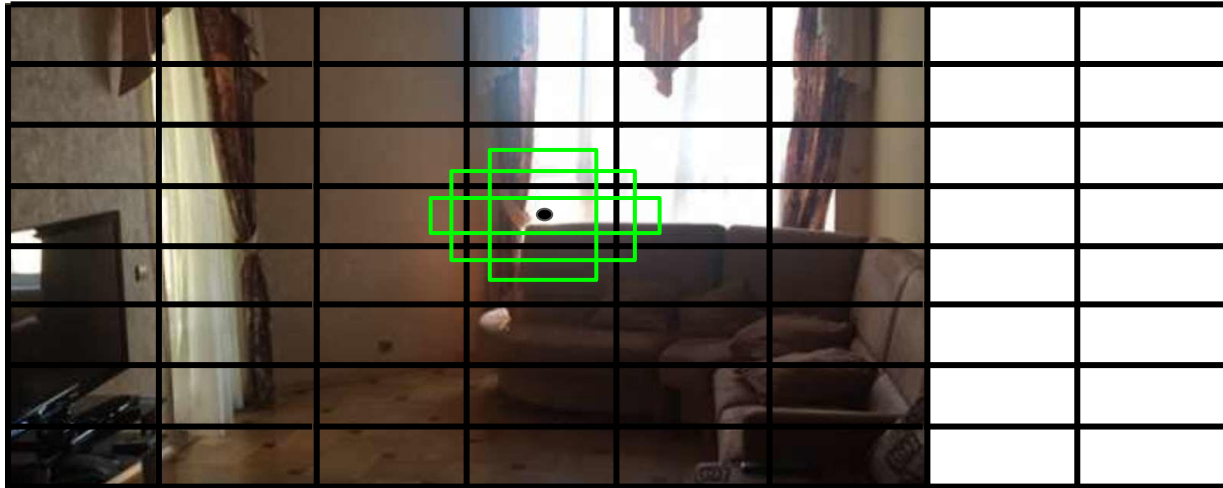
$a1, a2, a3, a4 \in [-\infty, \infty]$

$x_center_predict = x_center_anchor_bbox + a1 * anchor_bbox_width$
 $y_center_predict = y_center_anchor_bbox + a2 * anchor_bbox_height$
 $x_width_predict = anchor_bbox_width * e^{a3}$
 $x_height_predict = anchor_bbox_height * e^{a4}$



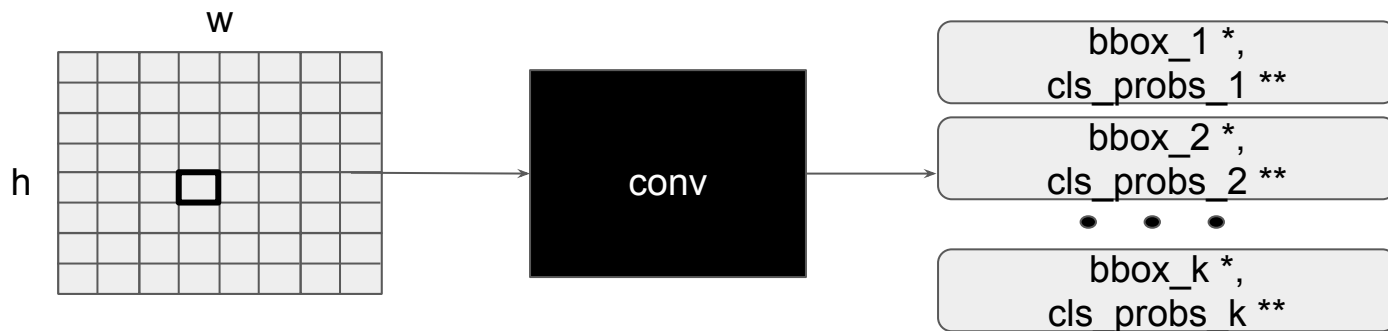
bboxer в SSD

anchors



bboxer в SSD

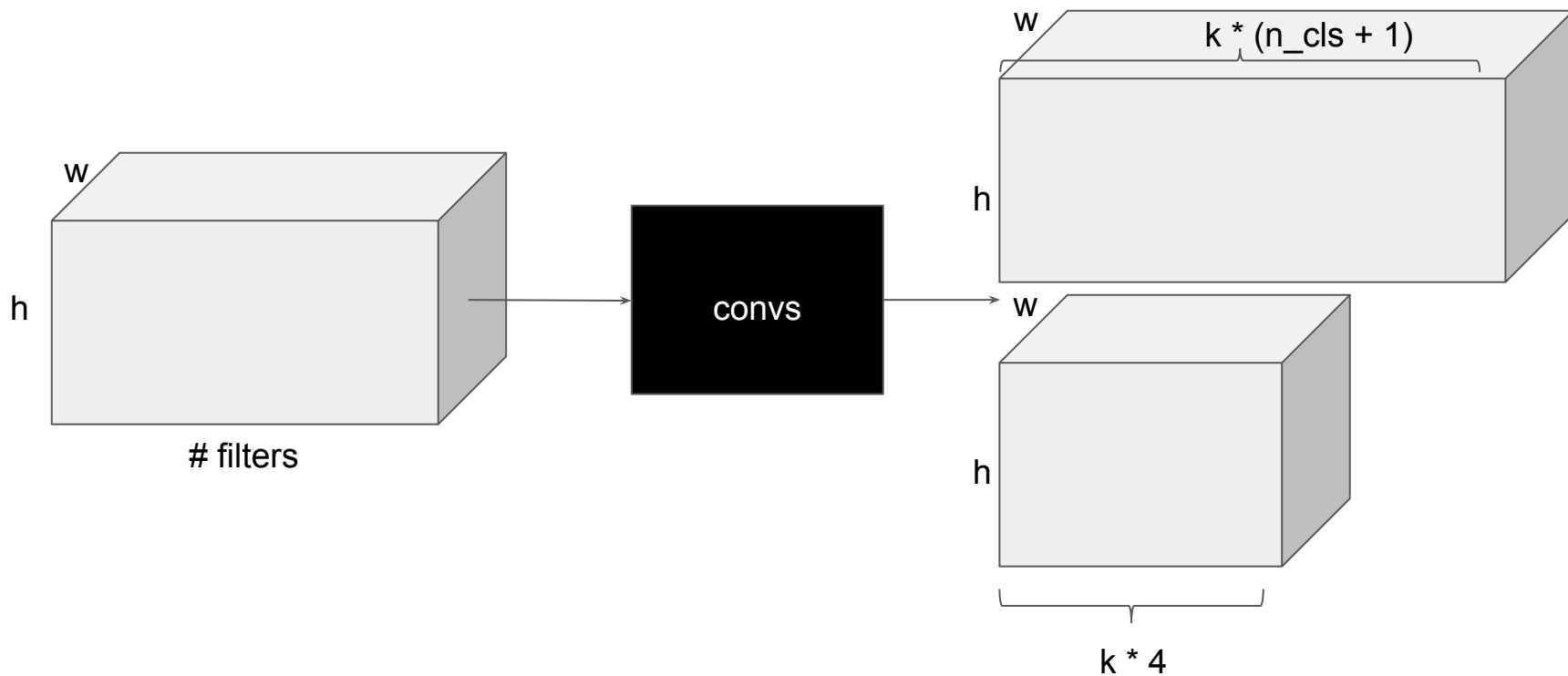
Кодирование



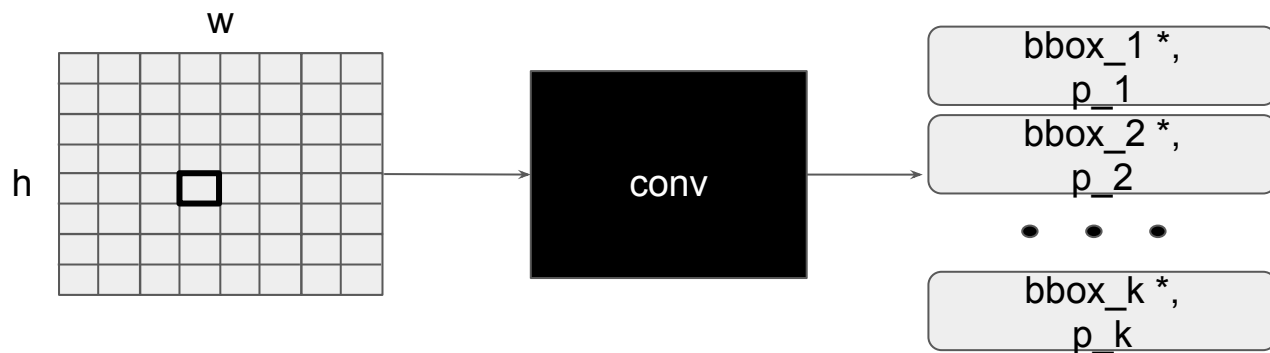
* $\text{bbox}_i^* = [a_1, a_2, a_3, a_4]$,

** $\text{cls_probs}_i^{**} = [\text{cls}_1, \text{cls}_2, \dots, \text{cls}_n, \text{cls_bg}]$

bboxer в SSD

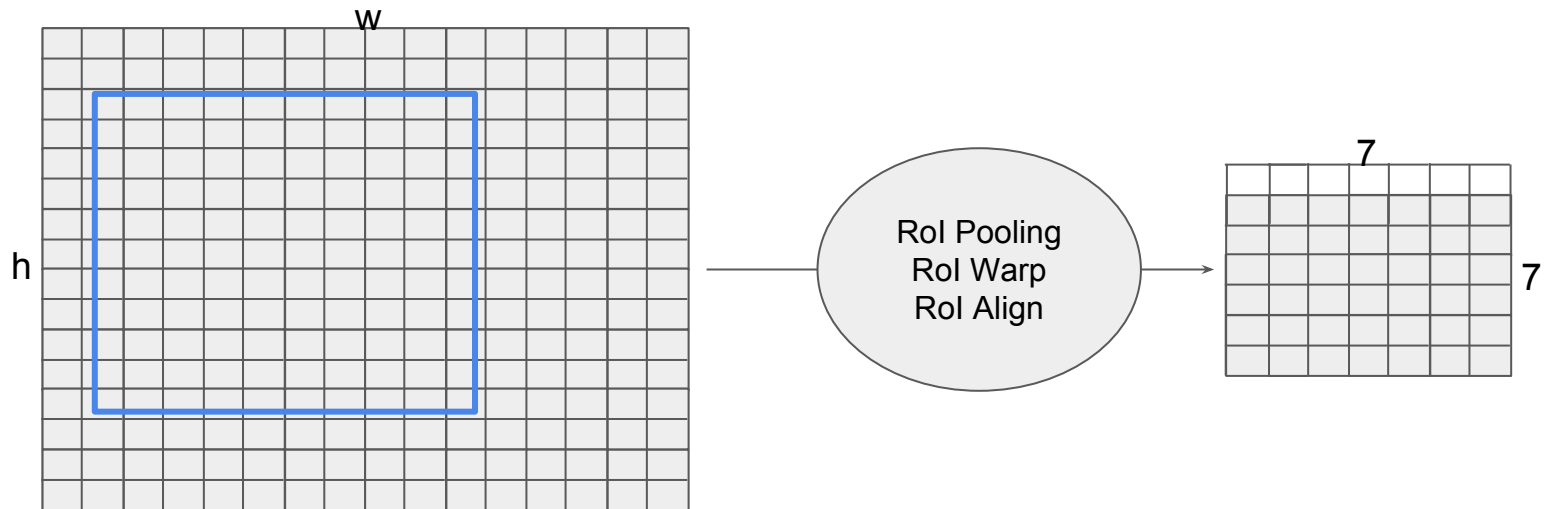


bboxer в FasterRCNN и MaskRCNN

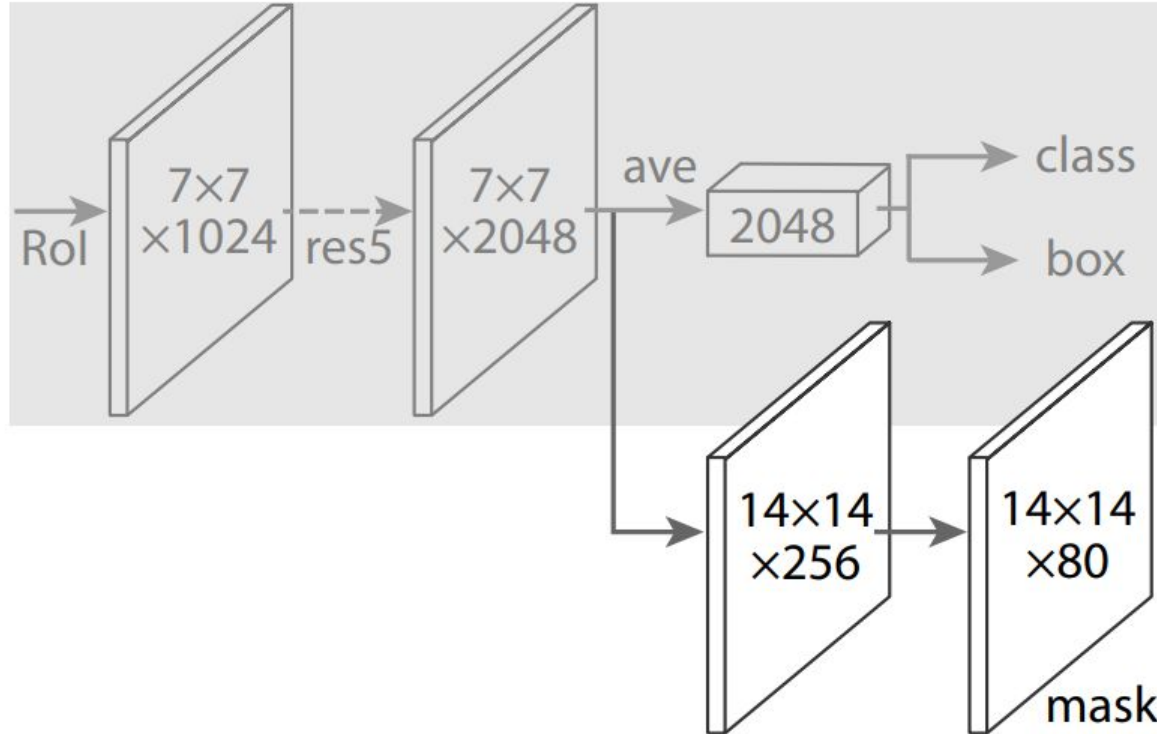


* $\text{bbox_}^* = [a1, a2, a3, a4]$

RoI feature extraction (RoI FE)

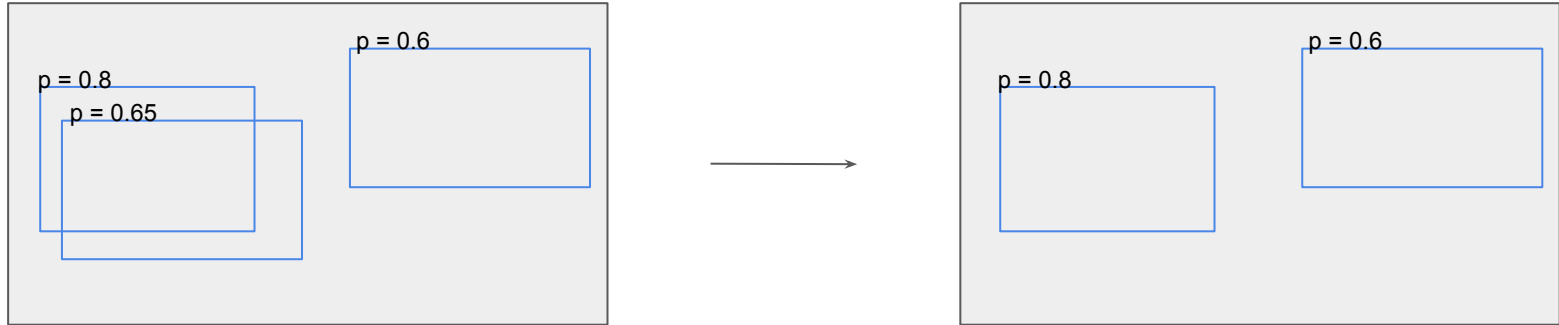


FasterRCNN и MaskRCNN heads



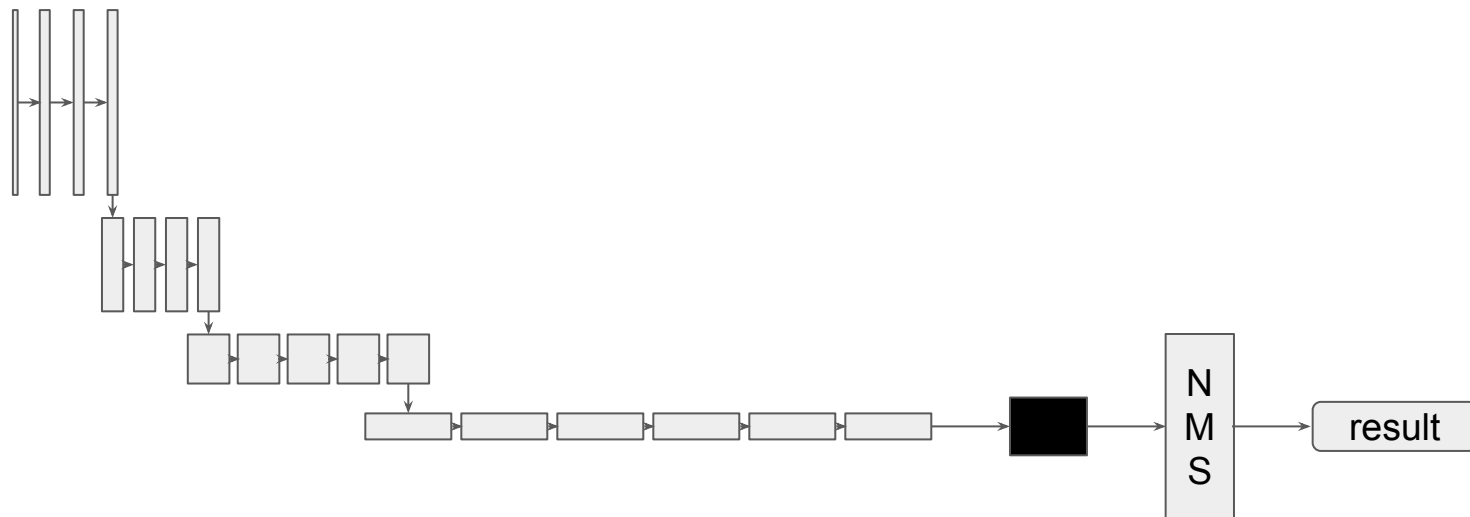
NMS

(non maximum suppression)



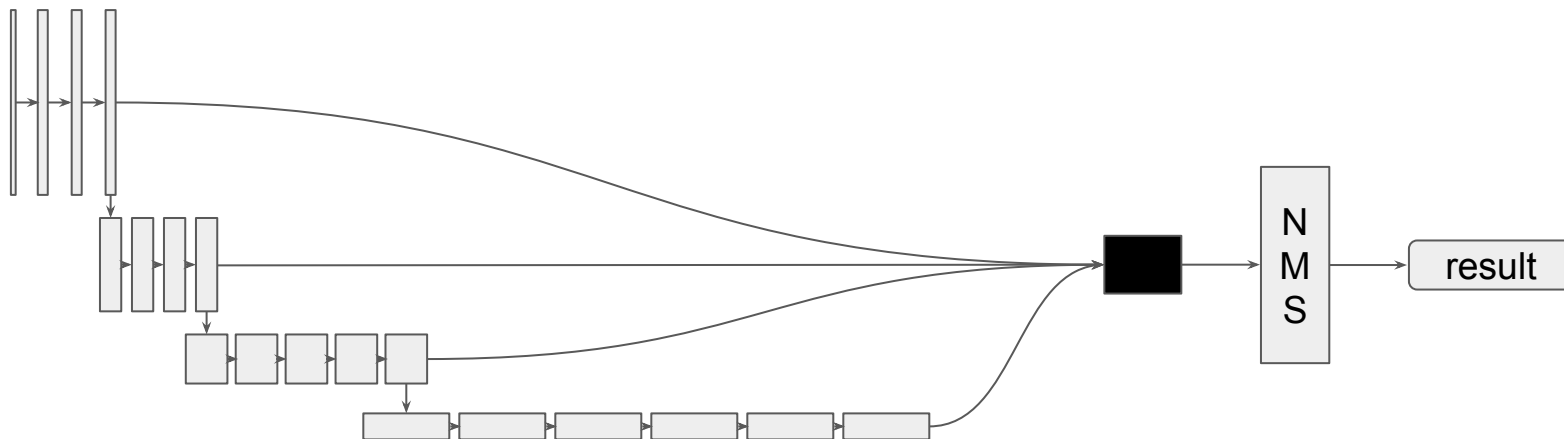
Итоговая архитектура

YOLO



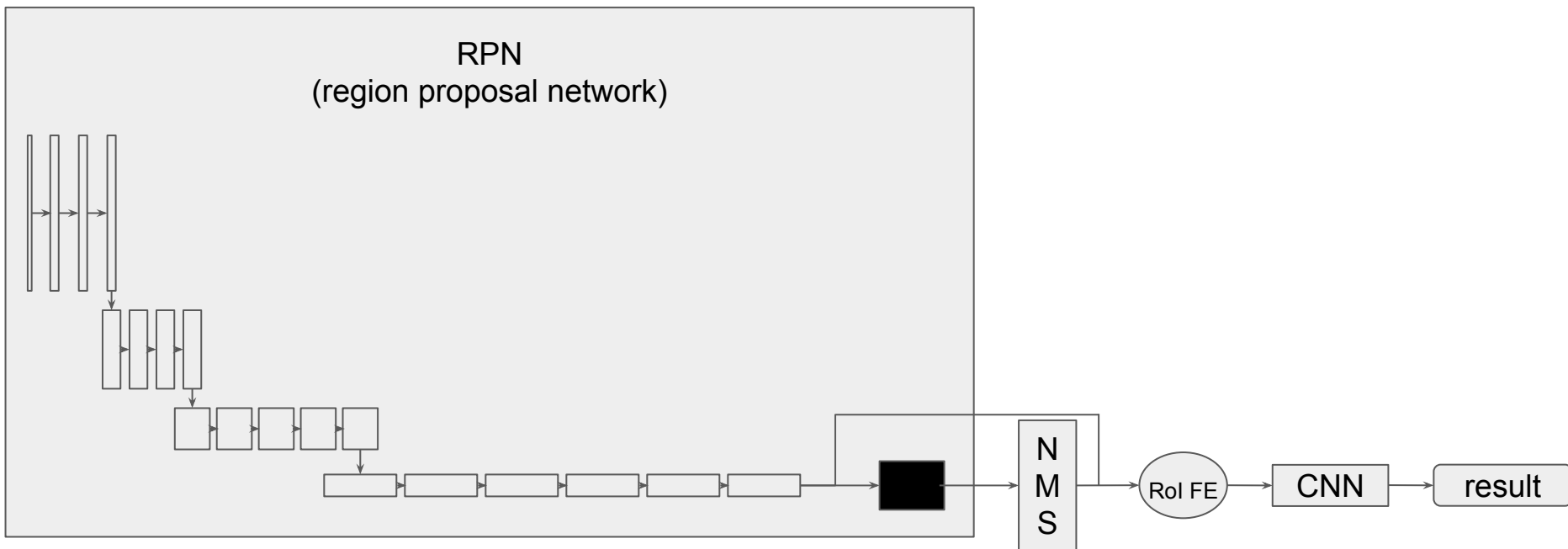
Итоговая архитектура

SSD



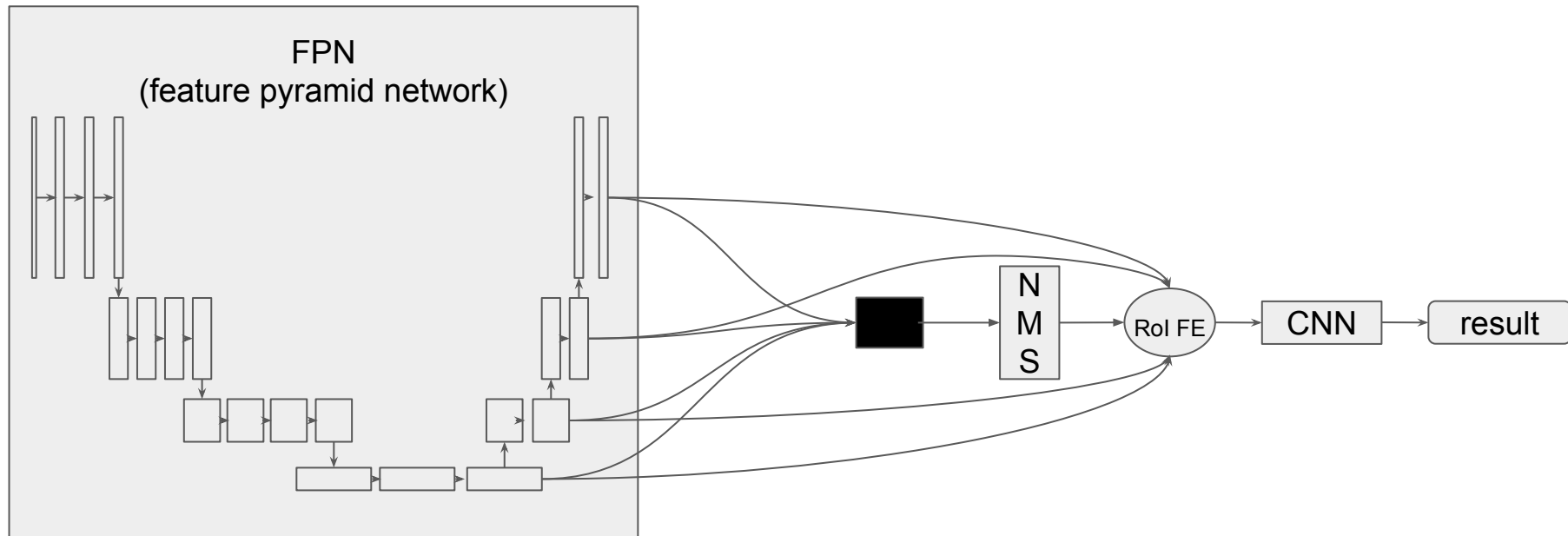
Итоговая архитектура

FasterRCNN



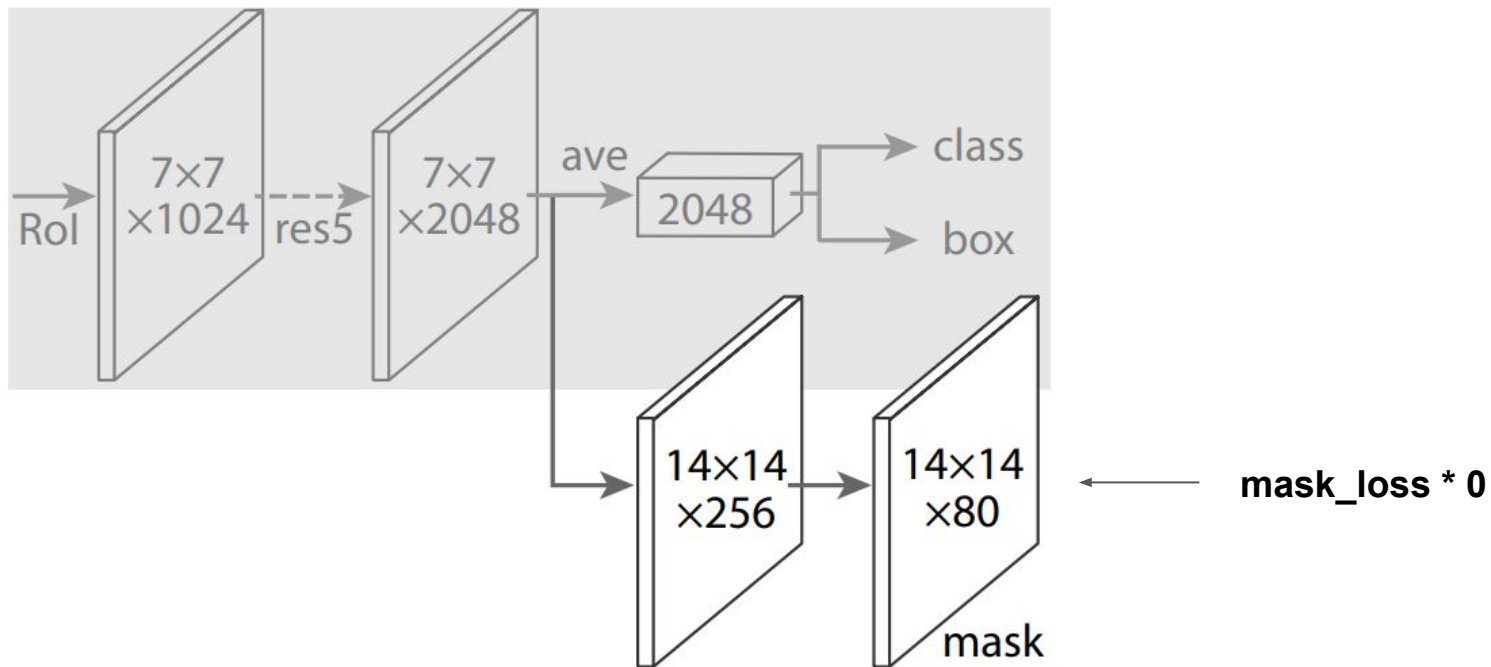
Итоговая архитектура

MaskRCNN

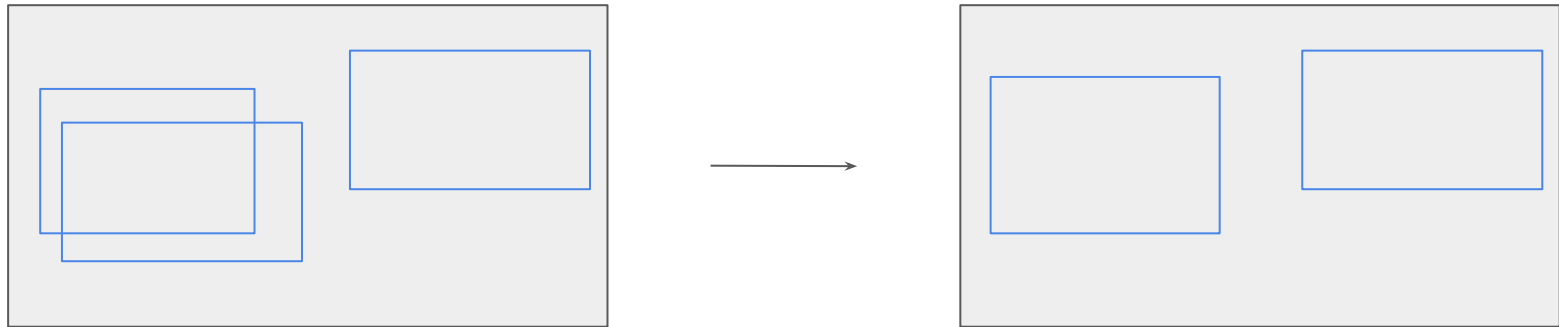


Наше решение

Основано на Keras MaskRCNN от https://github.com/matterport/Mask_RCNN



Ensembling



Валидация

7.5% на val, стратифицированный сплит.

Была большая корреляция с LB.

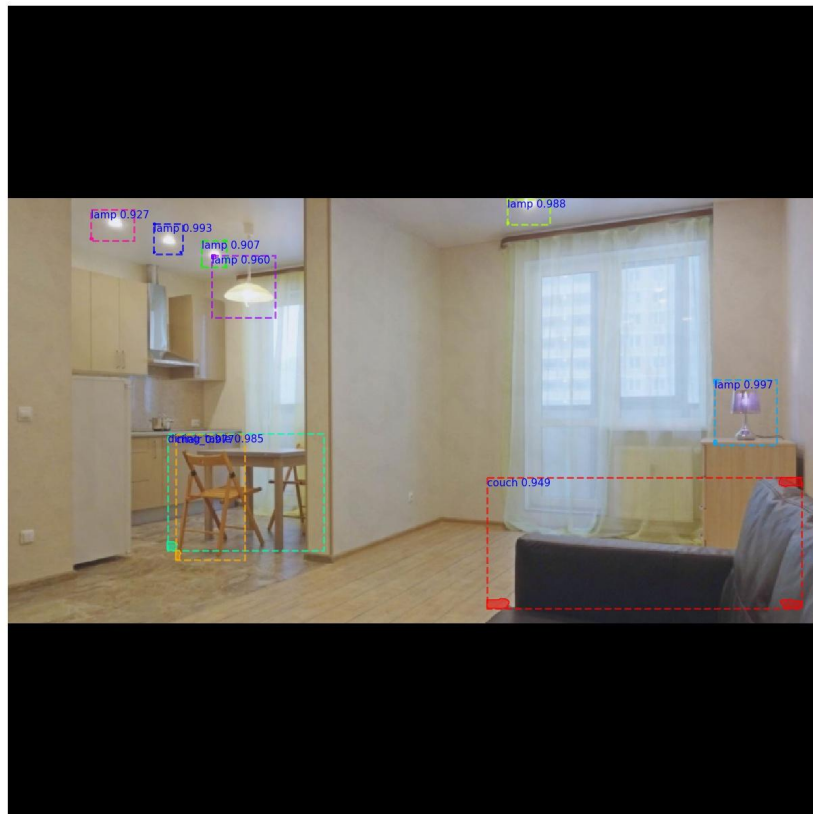
Обучение

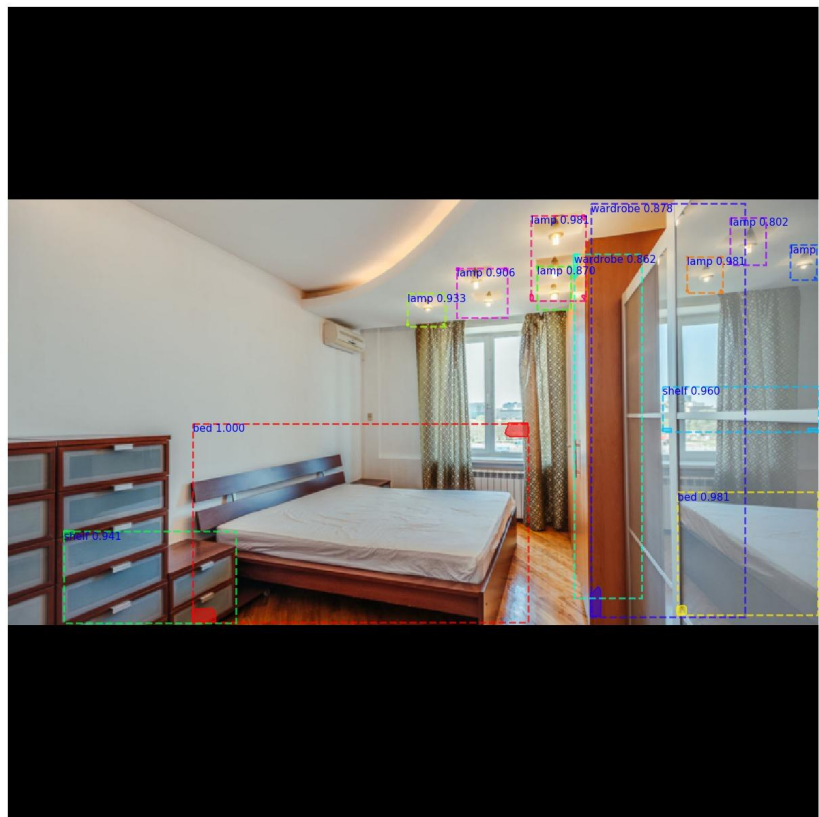
- В ночь с пятницы на субботу на 1ой 1080ti
- В субботу около 6 часов на 2ух 1080ti
- В ночь с субботы на воскресенье трейн упал
- В воскресенье 6 часов на 2ух 1080ti

Решение 2ого места











FasterRCNN

Что не получилось?
Что еще можно было попробовать?





Final LB

#	Team	Participants	Score	Last	Entries
1	Burger	  	0.13551	2018-03-18 18:55	20
2	OnlySVM	   	0.13439	2018-03-18 18:58	20
3	vla		0.11749	2018-03-18 17:09	14
4	OilStone	 	0.11125	2018-03-18 18:57	3



Спасибо за внимание