

Оценка популяции морских львов по снимкам

Тренировка по машинному обучению, 15 июля

NOAA Fisheries Steller Sea Lion Population Count

- Костя Лопухин (Scrapinghub)
- 4 месяца, 28 марта – 28 июня (я начал 29 апреля)
- 88 сабмитов, 2 место на private / 3 на public

Самое интересное

- не совсем стандартная задача, очень разные подходы
- большие отрывы
- странная валидация



Постановка задачи

Подсчет количества 5 типов сивучей (далее - котики):

- adult males
- subadult males
- adult females
- juveniles
- pups

Фото с дрона, размер $\sim 5500 \times 3500$ (1K train, 18K test)

Метрика: RMSE усредненная по колонкам













Разметка точками

- adult males
- subadult males
- adult females
- juveniles
- pups

+ координаты с форума











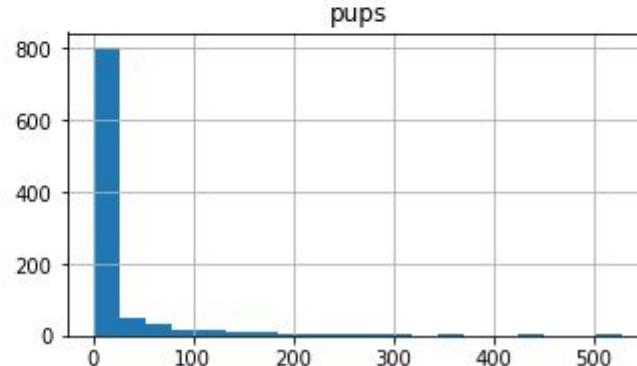
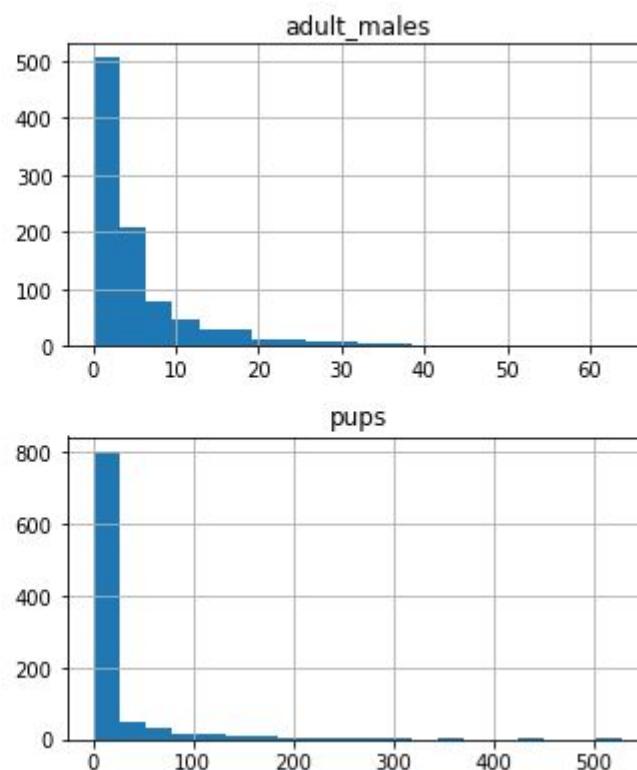
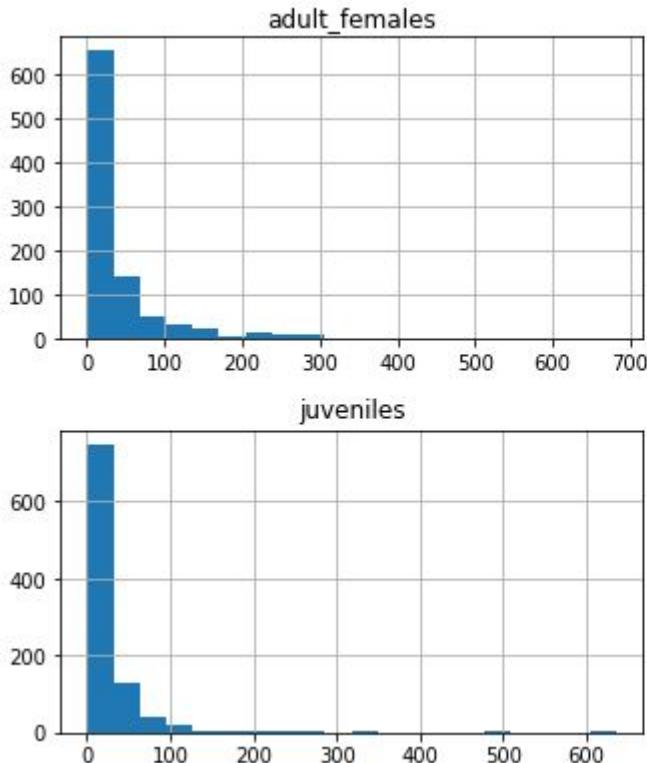


Как же их отличить?

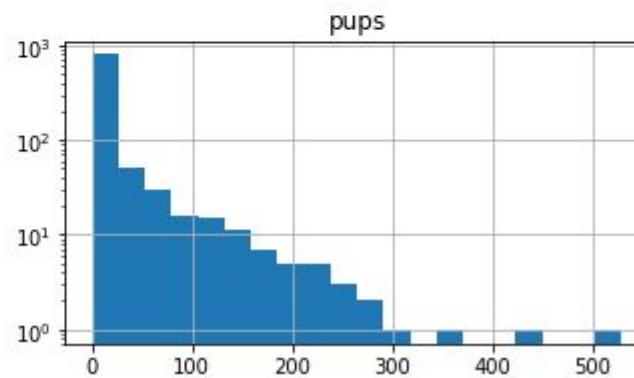
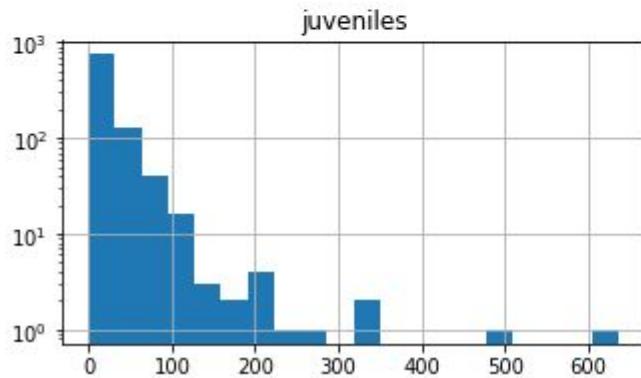
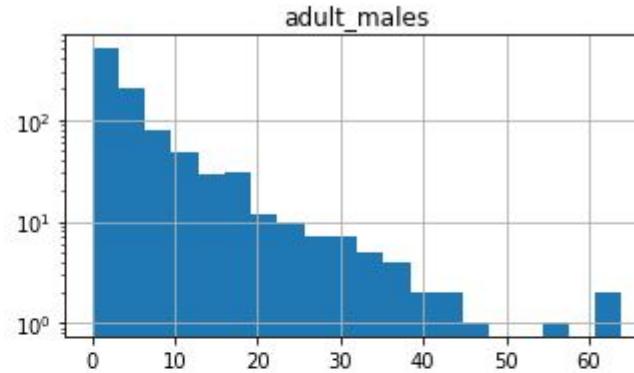
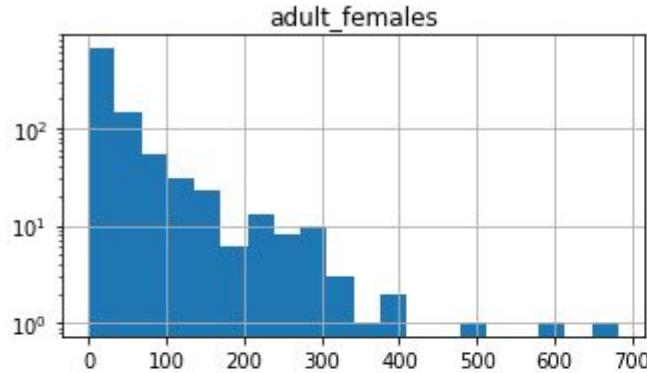
Katie, biologist at NOAA:

We distinguish by size. Sometimes we may deduce by 'behavior' that we can infer from where they are on the site. For example, sub-adult males may be near the size of adult males but on a site where there are many pups, the sub-adult males may be on the outer edges of the aggregation of animals and off on their own.

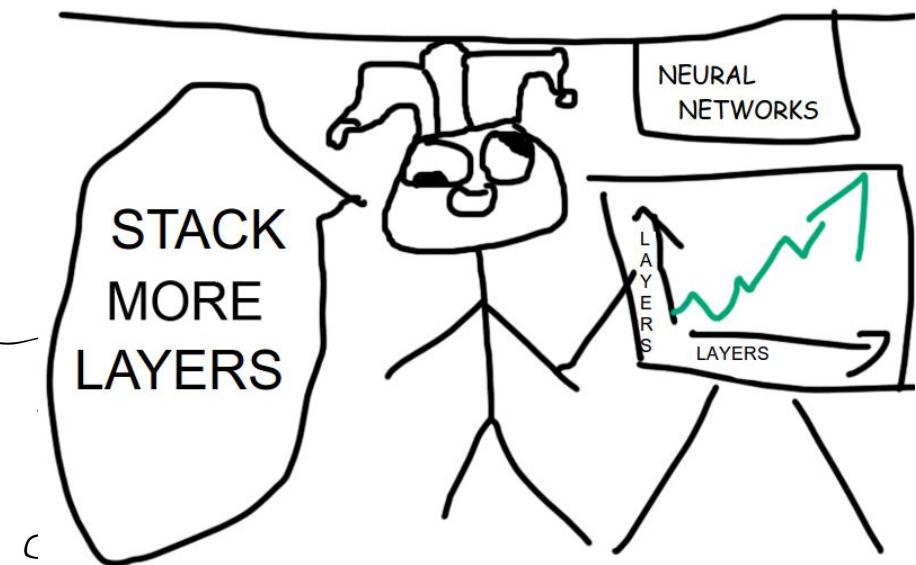
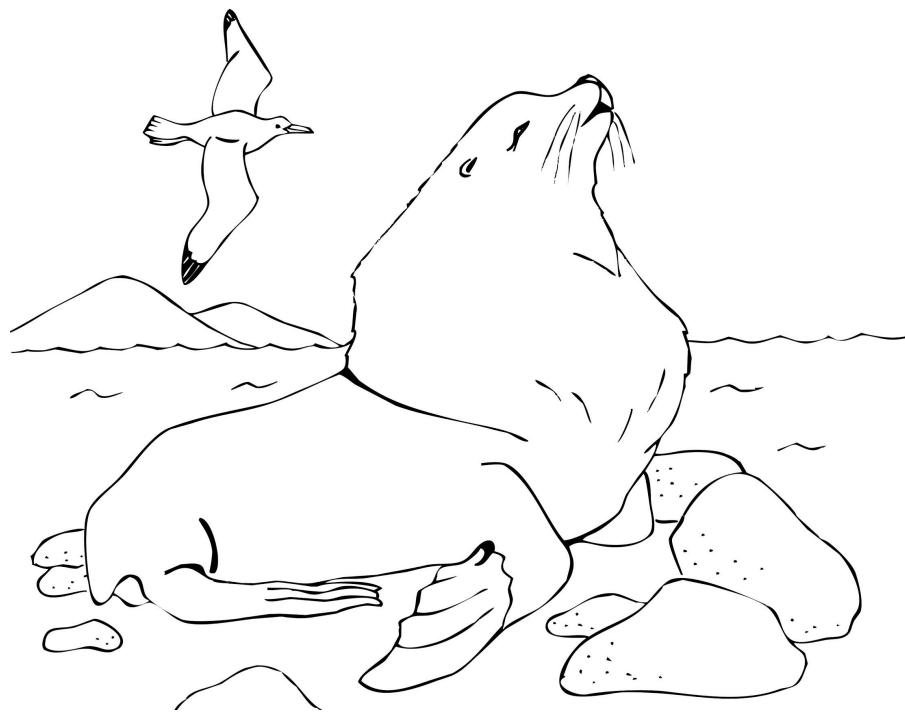
Распределение классов



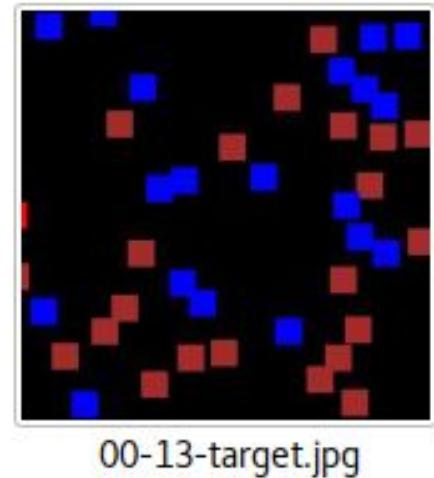
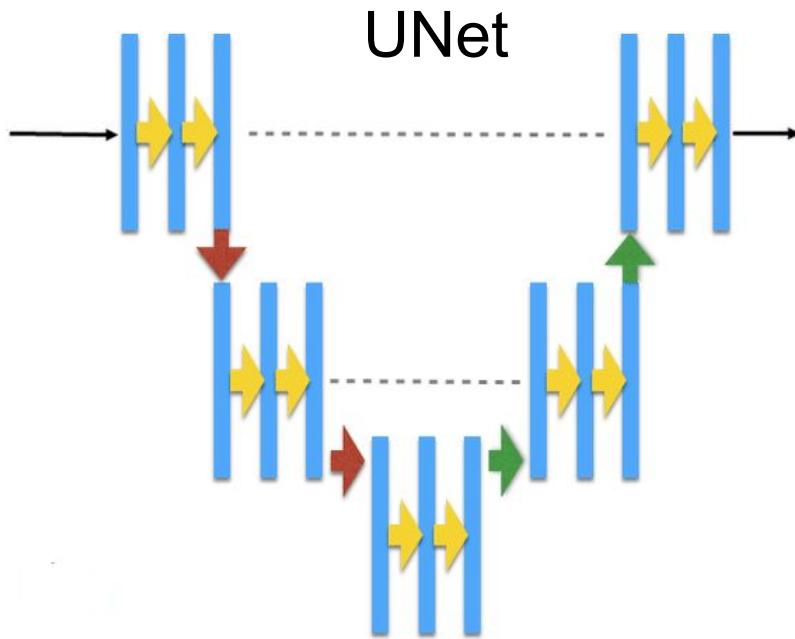
Распределение классов (log)



Решение задачи



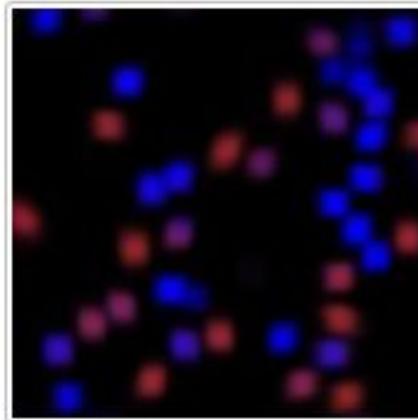
Шаг 1: сегментация на патчах 256×256



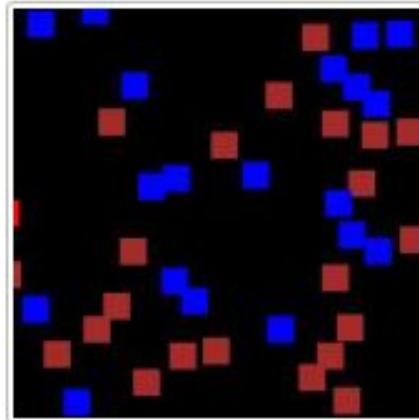
Шаг 1: сегментация на патчах 256×256



00-13-input.jpg

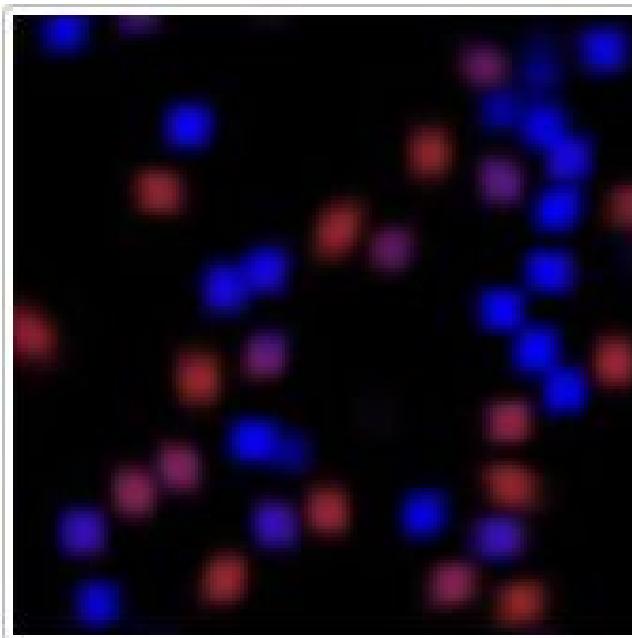


00-13-output.jpg

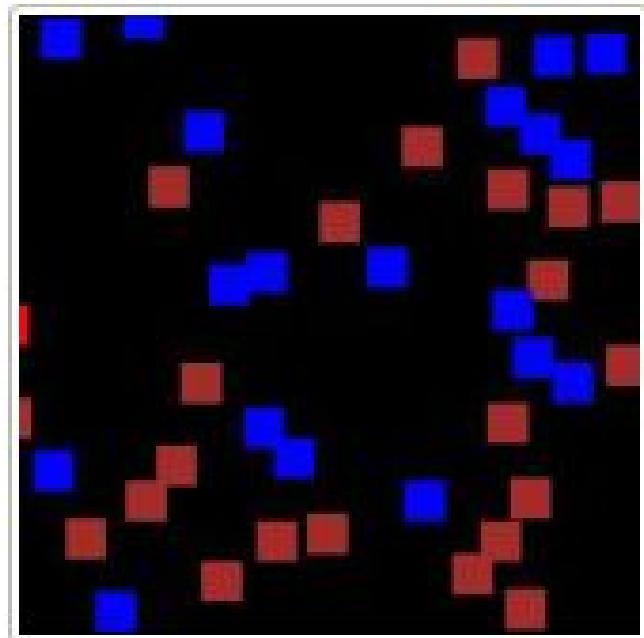


00-13-target.jpg

Шаг 1: сегментация на патчах 256×256

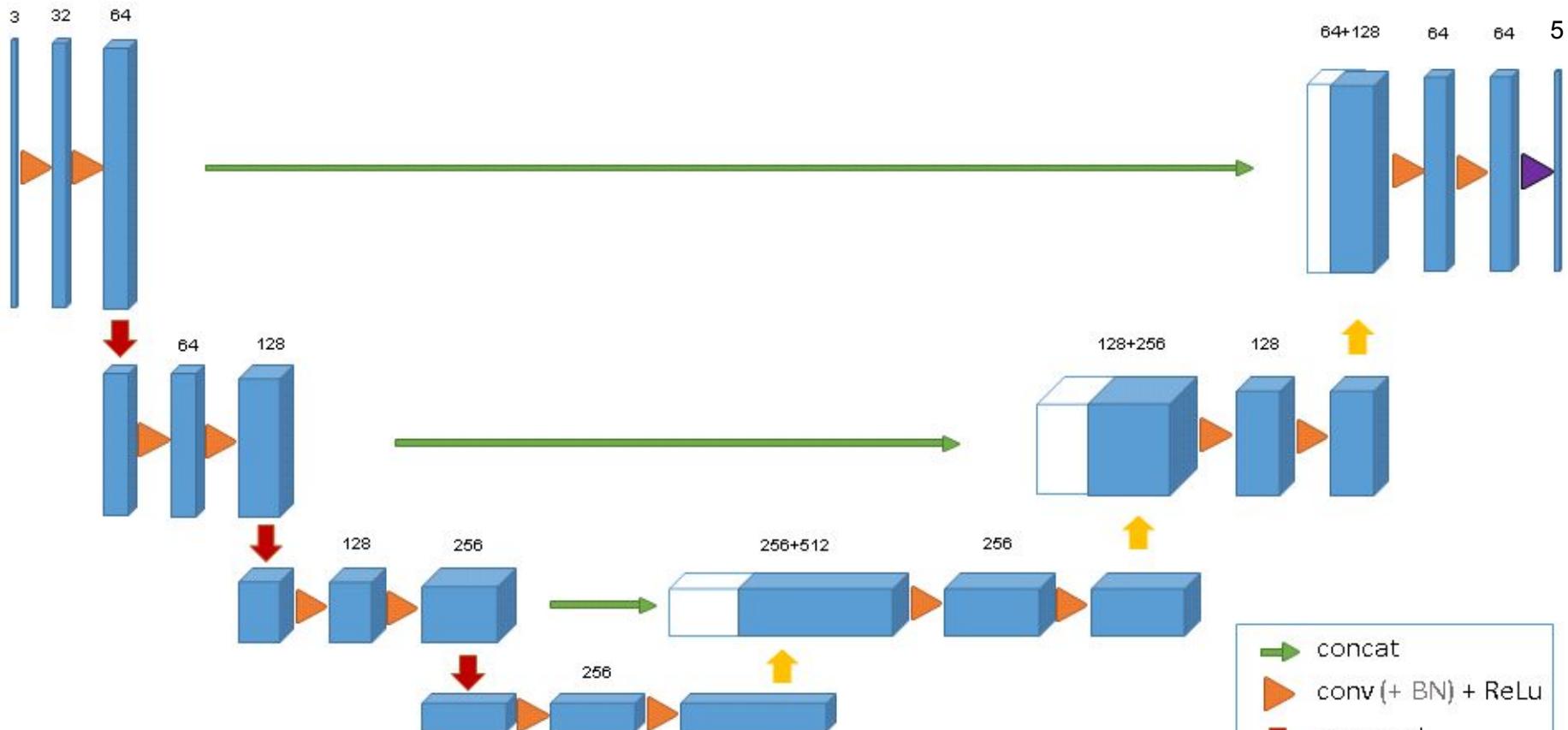


00-13-output.jpg



00-13-target.jpg





- concat
- △ conv (+ BN) + ReLu
- ↓ max pool
- ↑ up-conv
- conv

UNet: детали

- 4 “уровня” up/down
- на нижнем уровне 4x pooling вместо 2x
- upsampling вместо deconvolution
- batch size 32, batch normalization
- Adam, fixed lr
- softmax по 6 классам (с фоном), log loss
- 10-20 эпох, ~час на эпоху

UNet: что не получилось

- предсказание маски в 4 раза меньше
- dice/jaccard loss

Детали обучения

- вырезка патчей на лету, вращения
- 0.2 oversampling патчей рядом с котиками
- 0.8 – 1.6 scale augmentation

Детали обучения

- вырезка патчей на лету, вращения
- 0.2 oversampling патчей рядом с котиками
- 0.8 – 1.6 scale augmentation

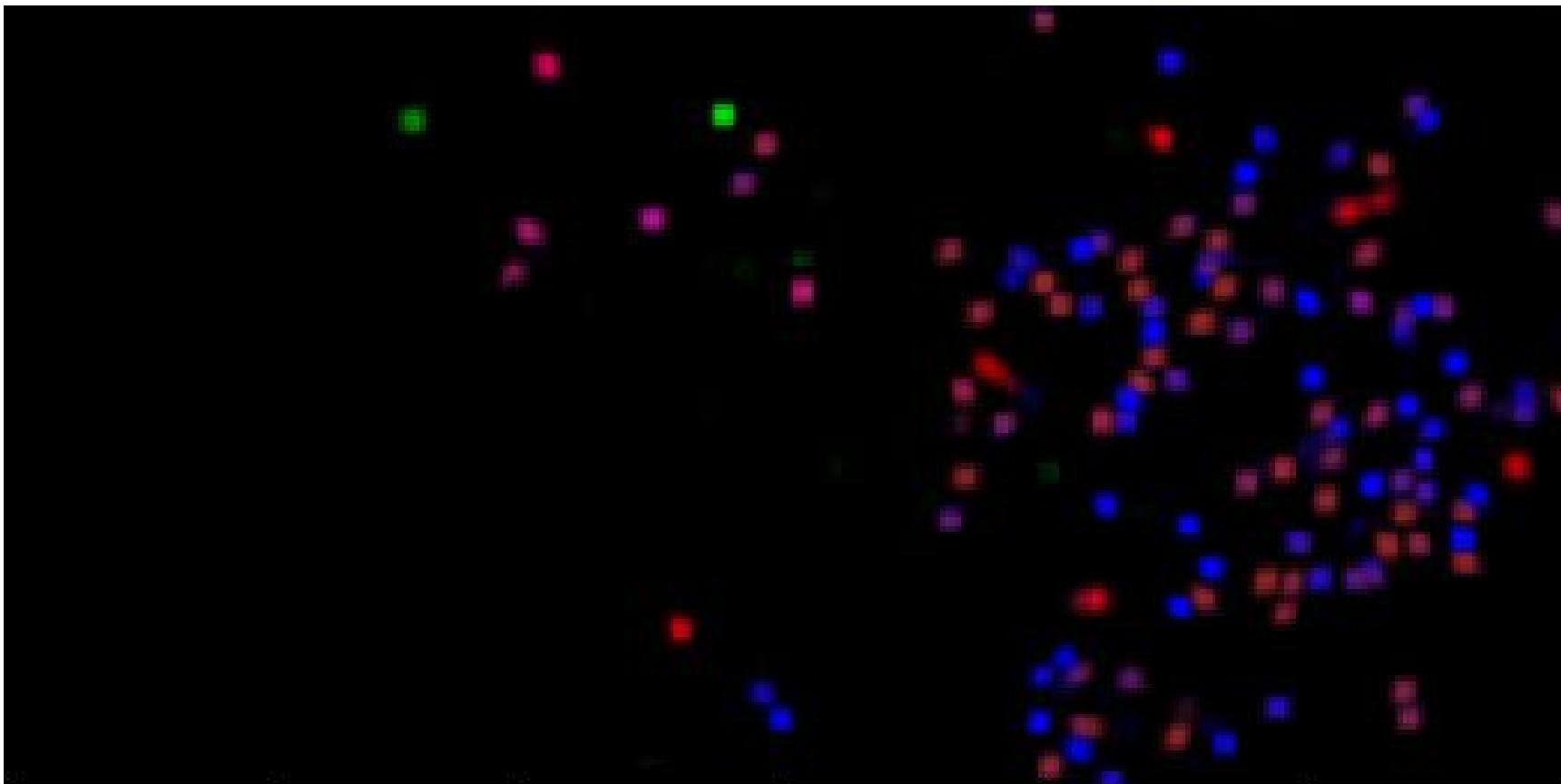


Шаг 2: регрессия на масках

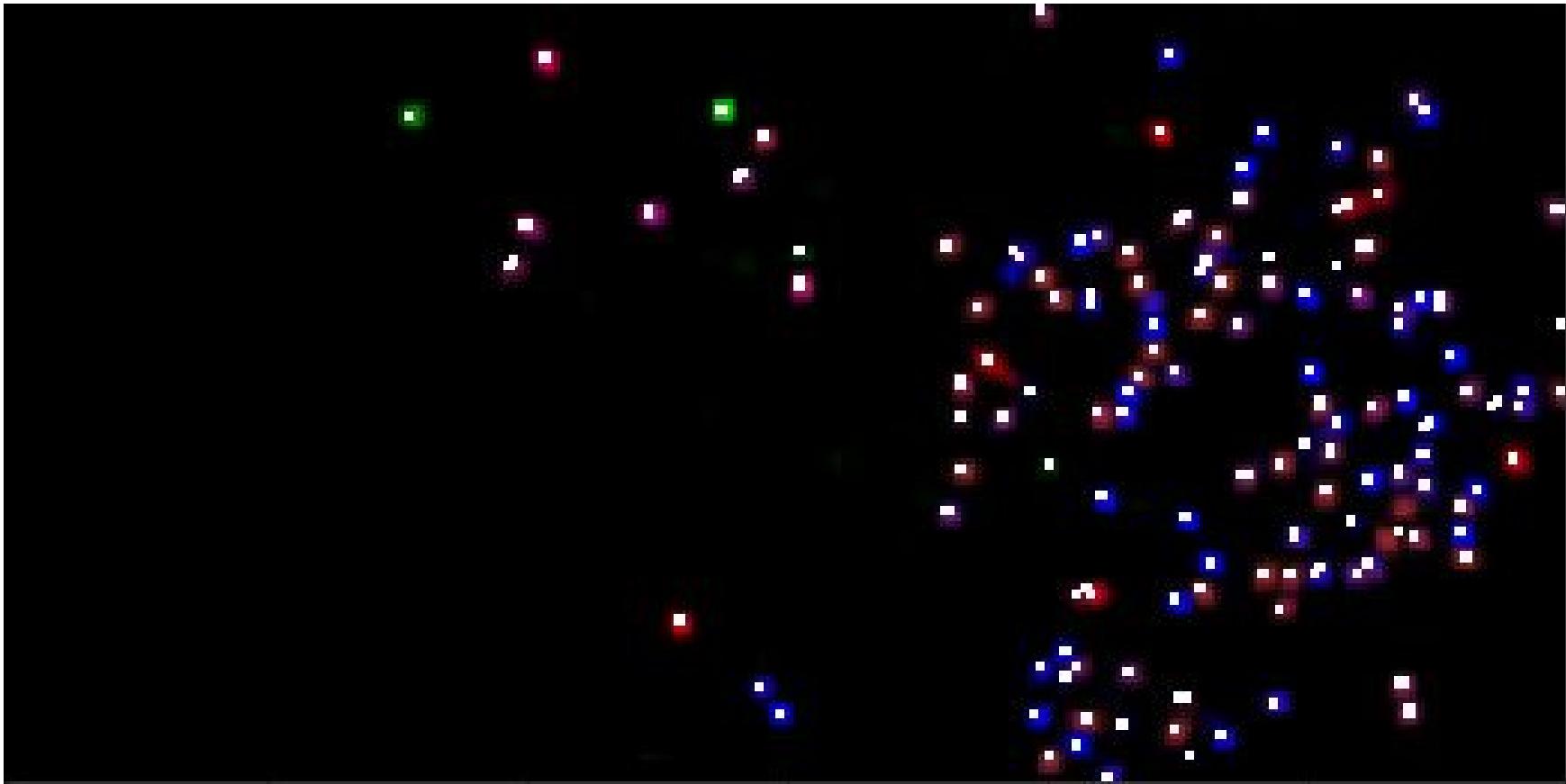
Фичи из патчей 240×240 для каждого класса:

- сумма предсказанных вероятностей
- сумма выше порога (0.04, 0.08, 0.24)
- число и сумма для блобов с двумя порогами

Шаг 2: регрессия на масках



Blob detection (`skimage.feature.blob_log`)



Шаг 2: регрессия на масках

Фичи из патчей 240×240 для каждого класса:

- сумма предсказанных вероятностей
- сумма выше порога (0.04, 0.08, 0.24)
- число и сумма для блобов с двумя порогами

Target: предсказать число котиков в патче.

2×ExtraTrees (sklearn), XGBoost

Валидация

Первые попытки:

- 12 RMSE local \Rightarrow 24 RMSE public.

Валидация

Первые попытки:

- 12 RMSE local \Rightarrow 24 RMSE public.

Итого:

- 10 local \Rightarrow 13.8 public, 13.0 private.
- 13 – 16 local \Rightarrow **13.4 public, 12.5 private.**

Улучшения в процессе

- уменьшение test в 2 раза (чтобы приблизить к train)
- регрессия по патчам, а не по всему снимку
- scale augmentation
- oversampling

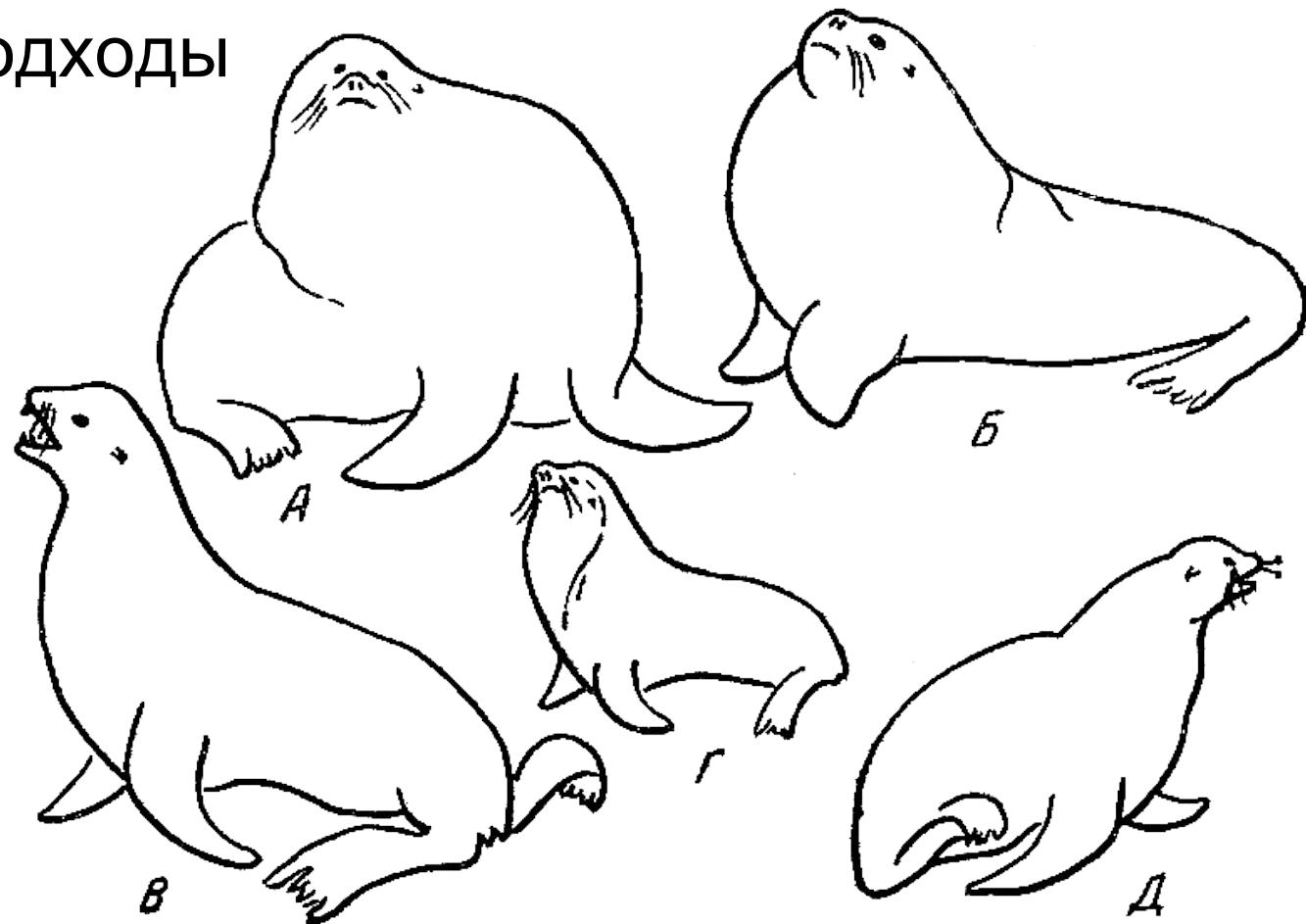
Заблуждения

- разделение плотно лежащих котиков – проблема (на самом деле проблема различение типов котиков)
- public LB нельзя верить (на самом деле можно)

Железо и время на обучение

- обучение (GTX 1070): 10-20 часов
- предсказание масок для 18K картинок: 12 часов
- предсказание количества: 3 часа (2 Xeon)
- Python 3.5, PyTorch, OpenCV, skimage, multiprocessing.

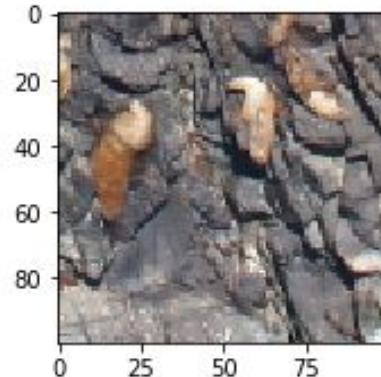
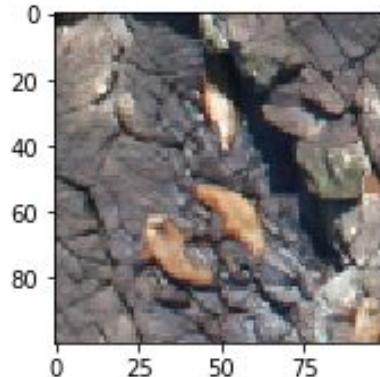
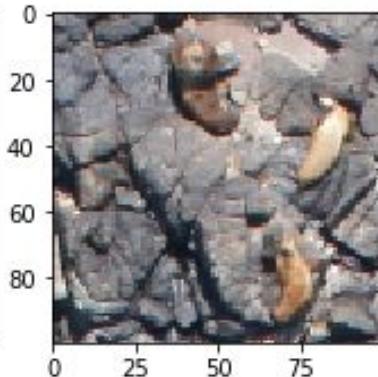
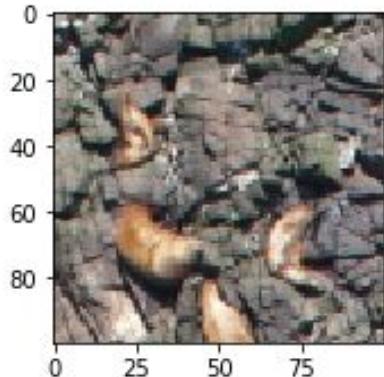
Другие подходы



Регрессия на тайлах

- outrunner (1 место)
- Артем Санакоев, Дмитрий Котовенко (2 public / 4 private)

```
[[1 0 3 0 1]
 [1 0 2 0 2]
 [0 0 4 0 3]
 [1 0 2 0 4]]
```



Регрессия на тайлах

outrunner:

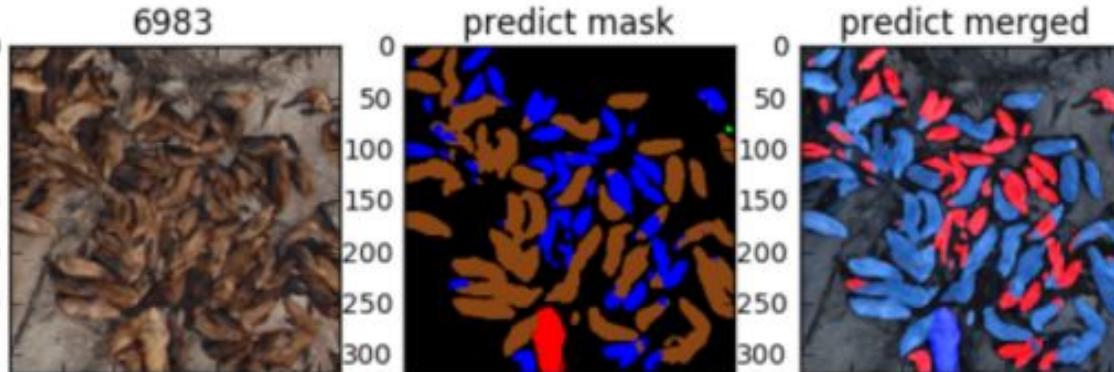
VGG head + LB-driven post-processing hacks.

Артем Санакоев, Дмитрий Котовенко:

Inception Resnet v2, scale augmentation, gaussian GT count.

bestfitting, 3 место: самое красивое решение

- UNet для предсказание наличия любого котика
- UNet для предсказания контура и типа котика
- GT контуры предсказаны другой UNet
- сложности с подсчетом количества



R&D&G, 5 место: самое правильное решение

- разметили масштабы изображений
- SSD для определения размеров котиков
- XGBoost для определения масштаба
- масштабирование картинок к одному масштабу
- сегментация VGG / ResNet + UNet
- подсчет на основе сегментации

outrunner trick

#	△pub	Team Name	Score
1	—	outrunner	10.85645
2	▲ 1	Konstantin Lopuhin	12.50888
3	▲ 1	bestfitting	13.03257
4	▼ 2	DL Sucks	13.18968
5	▲ 1	R & D & G	13.92761
6	▼ 1	LeiLei-WeiWei	13.97143
7	—	harshml	14.93910
8	—	Dmitriy Umerenkov	15.21353
9	▲ 1	Kyle	15.41015
10	▲ 1	pet's corner	15.88009

“outrunner trick”

```
test_hack = test.copy()
test_hack['juveniles'] *= 1.5
test_hack['adult_females'] -= 0.5 * test['juveniles']
test_hack['pups'] *= 1.2
```

outrunner: ? \Rightarrow 10.9

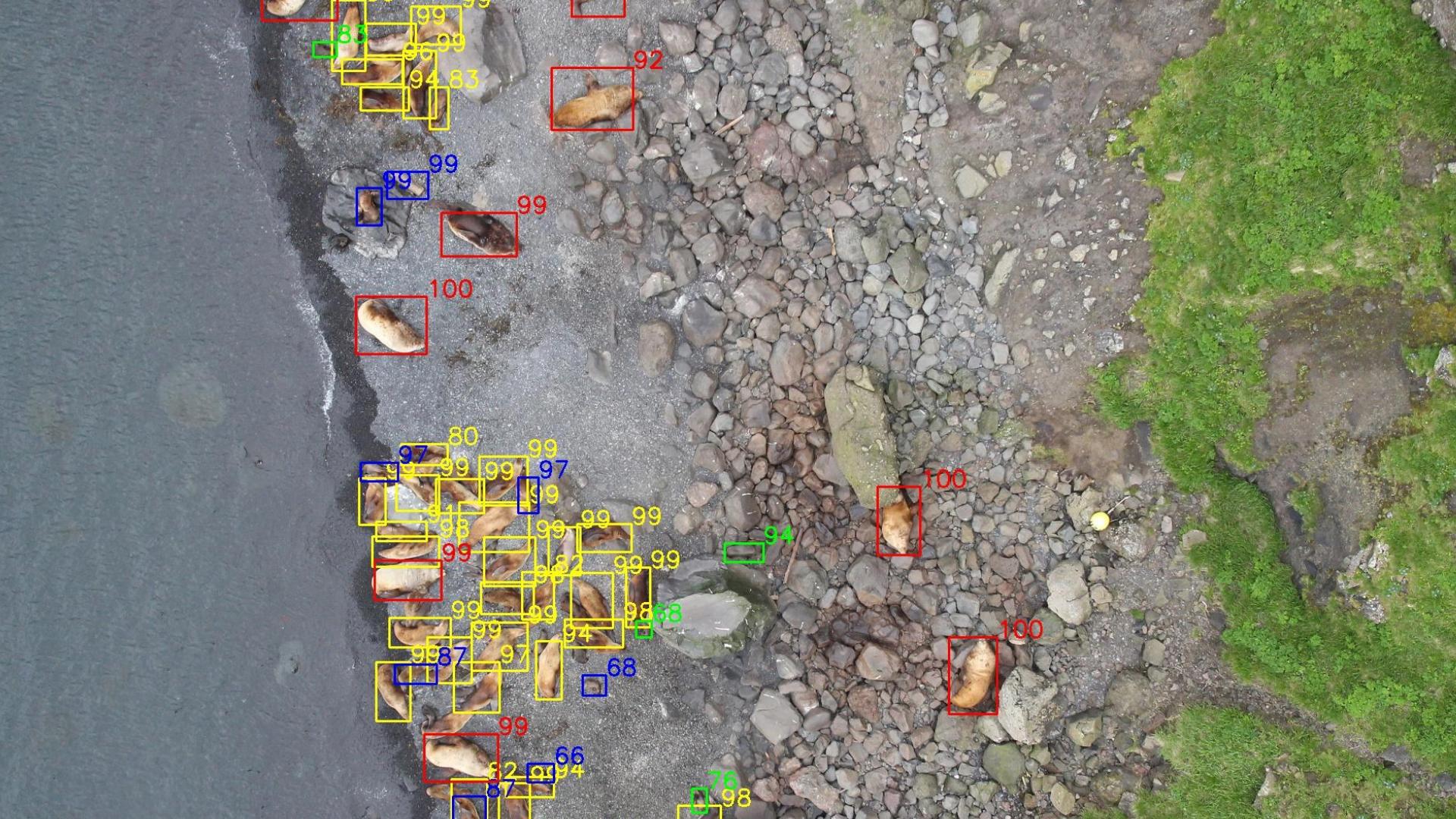
R&D&G: 13.9 \Rightarrow 12.1

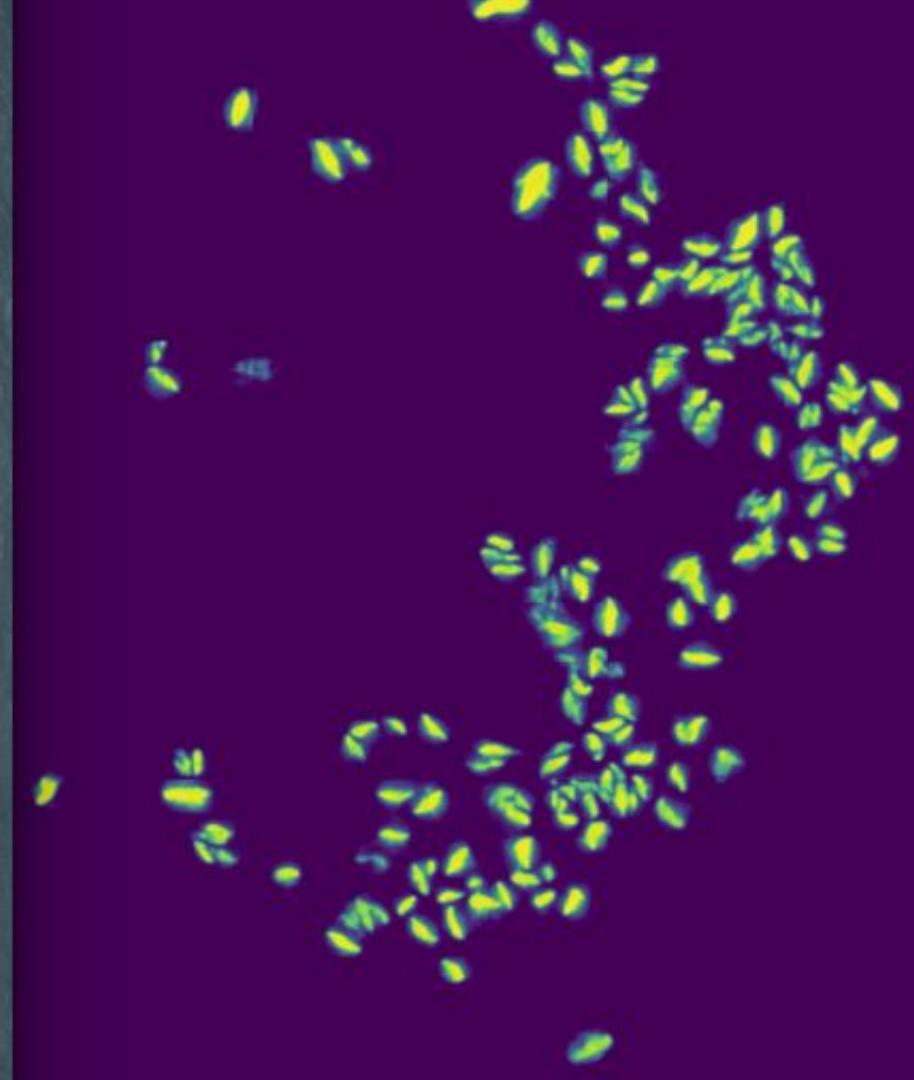
мой второй сабмит (лучший на CV): 13.0 \Rightarrow 12.6

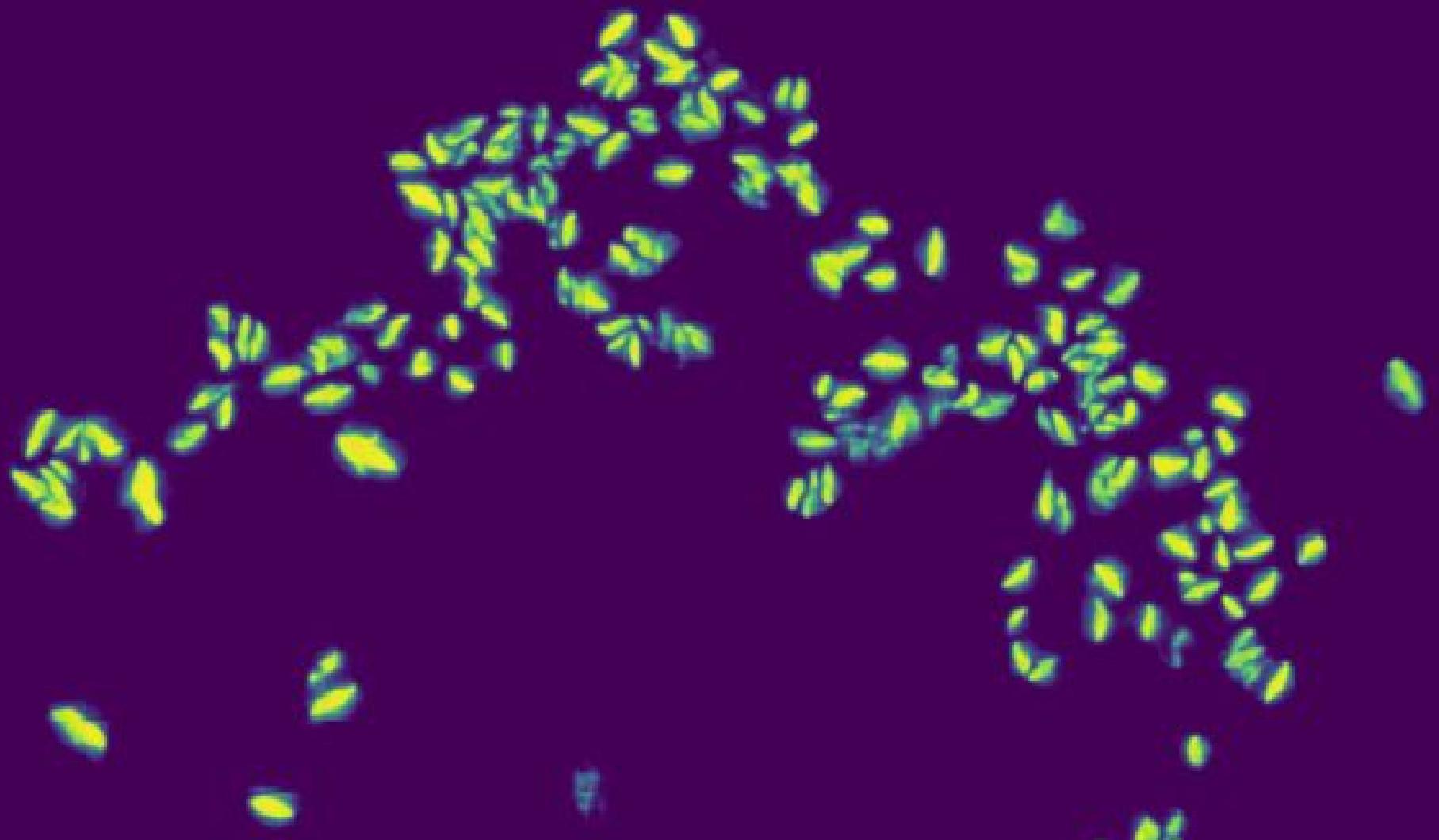
```
test_hack['pups'] *= 1.2
```

Артем Санакоев, Дмитрий Котовенко: 13.2 \Rightarrow 12.6

мой лучший сабмит: 12.5 \Rightarrow 12.0







Credits

Александр Буслаев, Артем Санакоев, Артур Кузин,
Владимир Игловиков, Дмитрий Котовенко, bestfitting,
Dmytro Poplavskiy, Gilberto Titericz Junior, outrunner, Russ
W.

Картинки с сайта заповедника
komandorsky.ru

#ODS



Спасибо! Вопросы?

