

Выбранное соревнование - IWildCam2019 (<https://www.kaggle.com/c/iwildcam-2019-fgvc6>).

Главная сложность возникла с распаковкой архива трейновых картинок: они не помещались в папку `/data/kaggle-iwildcam-2019/train-images/`, из-за чего в процессе обучения вылезала ошибка о том, что картинка имеет тип `None Type`. Проблема была решена путем распаковки картинок в корневую папку `/data/`.

Картинки необходимо классифицировать по 14 классам, при этом классы не принимают значения от 0 до 13. Поэтому был написан `encoder`, кодирующий существующие классы цифрами от 0 до 13:

Encoded classes	Initial classes
1	0
6	1
2	3
4	4
3	8
12	10
7	11
5	13
10	14
8	16
9	17
11	18
0	19
13	22

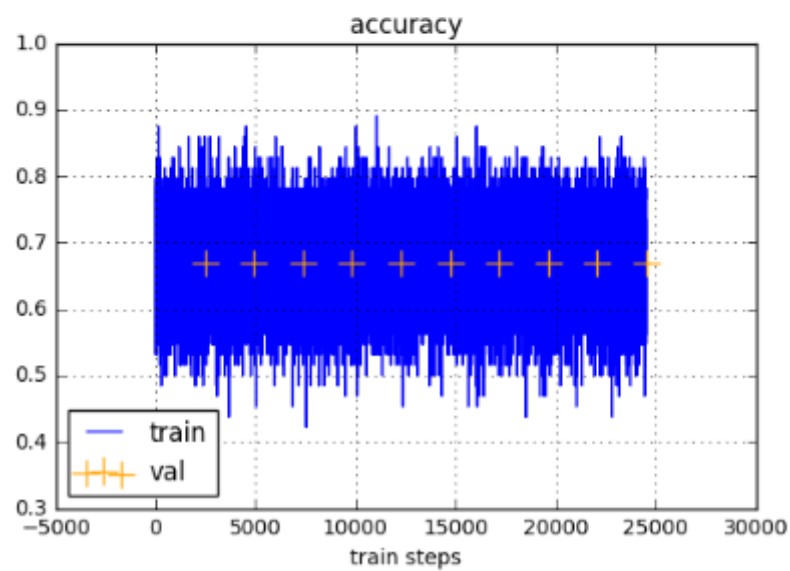
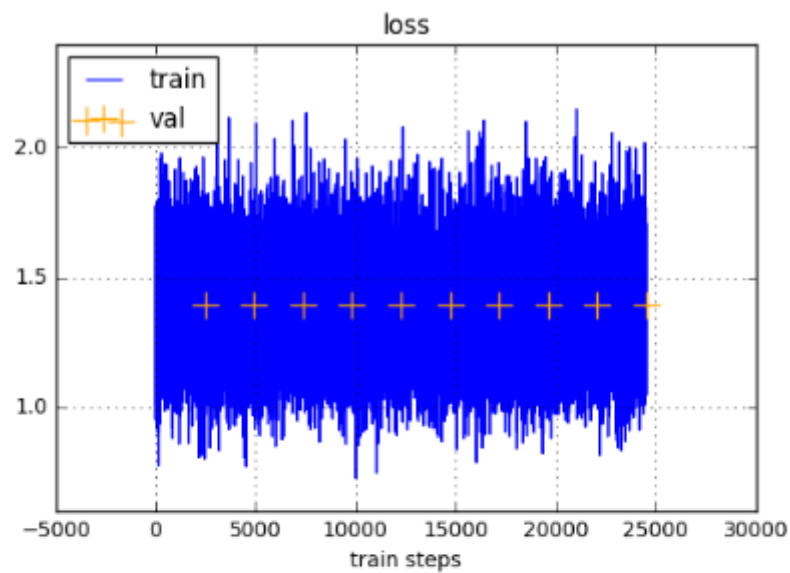
Далее, был написан класс `IWildDataset` для формирования трейн и тест датасетов. К ним применялась следующая трансформация:

- Обрезание картинки до размера 64*64
- Перевод в тензор
- Нормализация

Был написан наш стандартный код для обучения и валидации модели, который также визуализирует `loss` и `acc` на трейне и валидации.

Далее, была написана примитивная сетка. Для проекта по DL будет применена более адекватная модель.

Результат модели лучше случайного классификатора.



Видно, что сетка не особо учится (затухающие градиенты?), но с эти разберемся уже в рамках проекта по DL