

# Инструкция по маркировке датасета для распознавания гаджетов и прочих устройств

## Введение

Для маркировки изображений мы будем использовать инструмент - OpenLabeling, который запускается как скрипт Python, содержащего каталог входных изображений и текстовый файл с именами всех необходимых классов. Этот инструмент также помогает делать маркировки на видео.

## Требования

1. Перед запуском программы соберите базу данных с изображениями. Поскольку работа посвящена теме детектирования читерства во время экзамена, вам необходимо понять какие гаджеты могут быть применены для списывания (смартфоны, планшеты, телефоны, smartчасы и прочее), а так же учитывать разрешенные девайсы (клавиатуры, мышки, калькуляторы). Разрешенные девайсы также должны быть включены в модель, чтобы нейронная сеть не могла спутать их с запрещенными.
2. Количество изображений для тренировки могут варьировать в зависимости от количества классов. В лучшем случае для одного класса необходимо иметь примерно 300-500 фото (в худшем 150-200). Например, 5 классов = 1500-2000 фото. Чем больше фото вы будете использовать, тем точнее в дальнейшем будет работать ваша модель.
3. Немаловажную роль играет качество фото. Все фотографий должно быть примерно одинакового размера. Например, фотографий с разрешением 300x300 и 3500x2400 недопустимы для одной и той же маркировки.
4. Как только у вас сформировался список девайсов (классов) вы можете приступить к запуску программы.

## Запуск программы

1. Сделайте клон данного репозитория (или скачайте zip файл).

<https://github.com/techzizou/OpenLabeling>

```
git clone https://github.com/techzizou/OpenLabeling.git
```

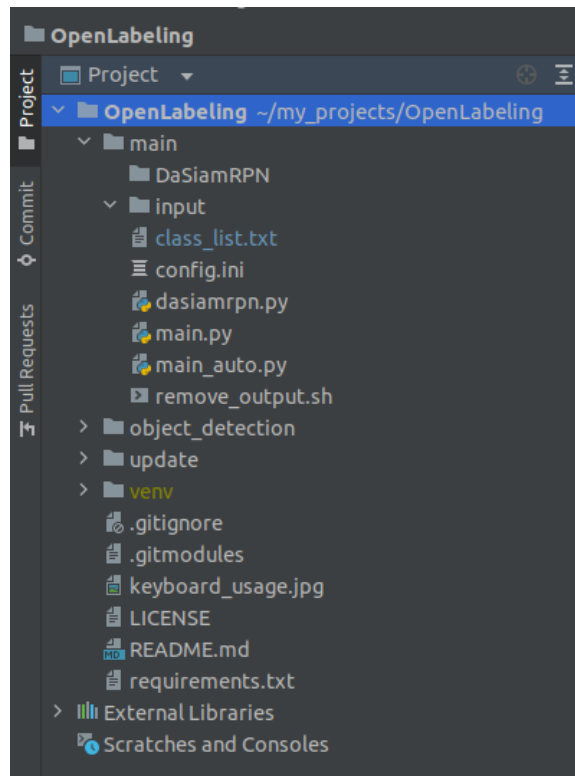
2. Перейдите к папке `OpenLabeling`, создайте виртуальную среду и запустите скрипт для установки дополнительных модулей из файла `requirements.txt`:

```
cd OpenLabeling/  
python3 -m venv venv  
source venv/bin/activate  
pip install -r requirements.txt
```

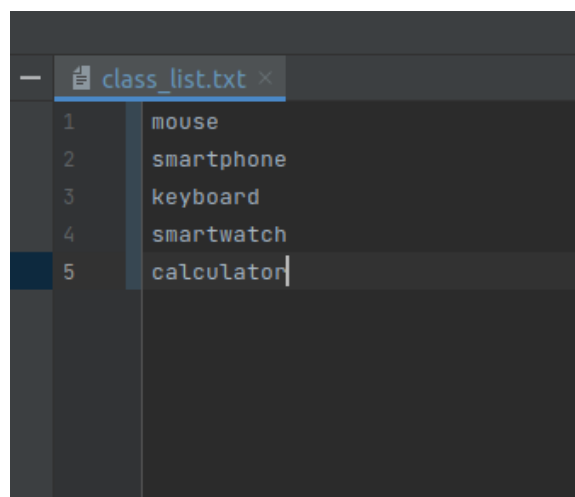
P.S. Если у вас произошла ошибка при установке последнего скрипта, установите модули вручную:

```
pip install lxml  
pip install numpy  
pip install opencv-python  
pip install tqdm
```

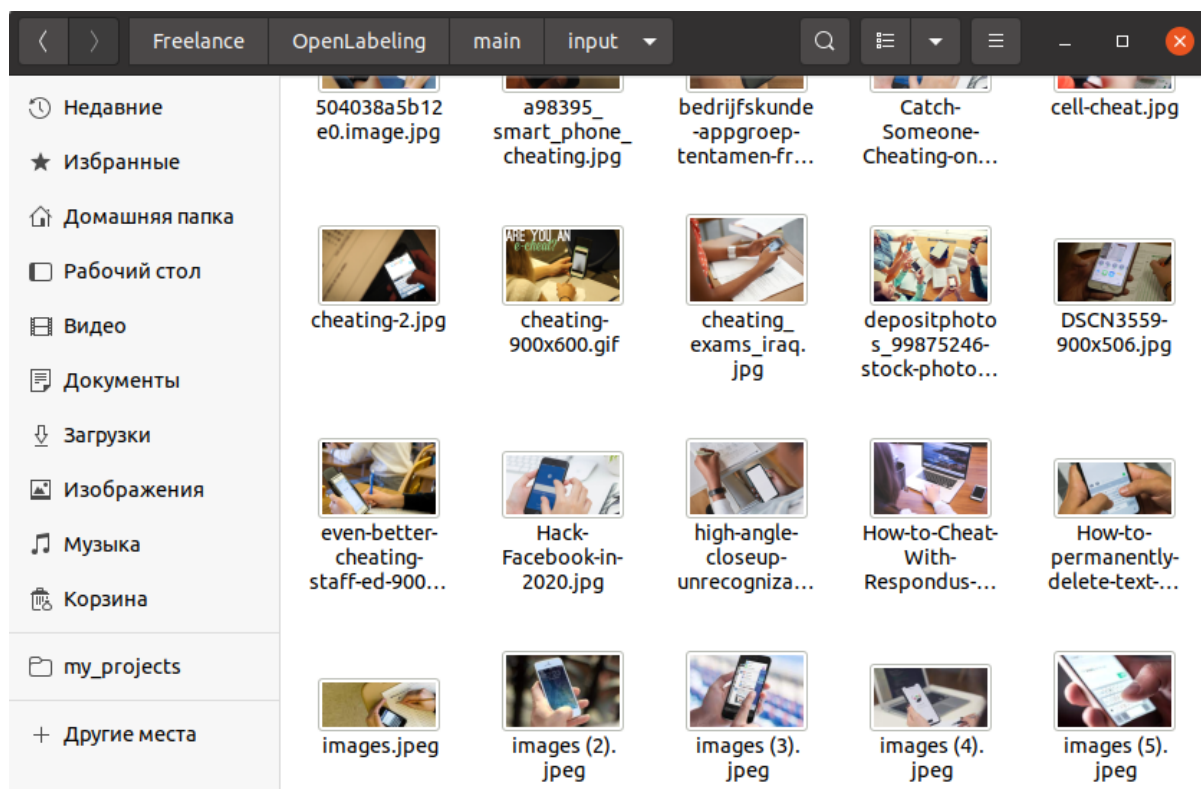
3. Перейдите в папку `main`. Очистите содержимое папки `input` и содержимое `class_list.txt`. В итоге ваша папка должна выглядеть следующим образом:



4. Откройте `class_list.txt` и напишите названия нужных классов. По одному в каждой строке без пробела между словами класса.



5. Поместите все ваши изображения в папку `input`



Постарайтесь найти изображения максимально хорошего качества. Качество данных имеет большое значение для определения качества результата. От качества зависят веса которые вы получите при обучении. Постарайтесь найти/сделать фотографий максимально похожие на экзаменационные кадры.

Примеры можете посмотреть тут →

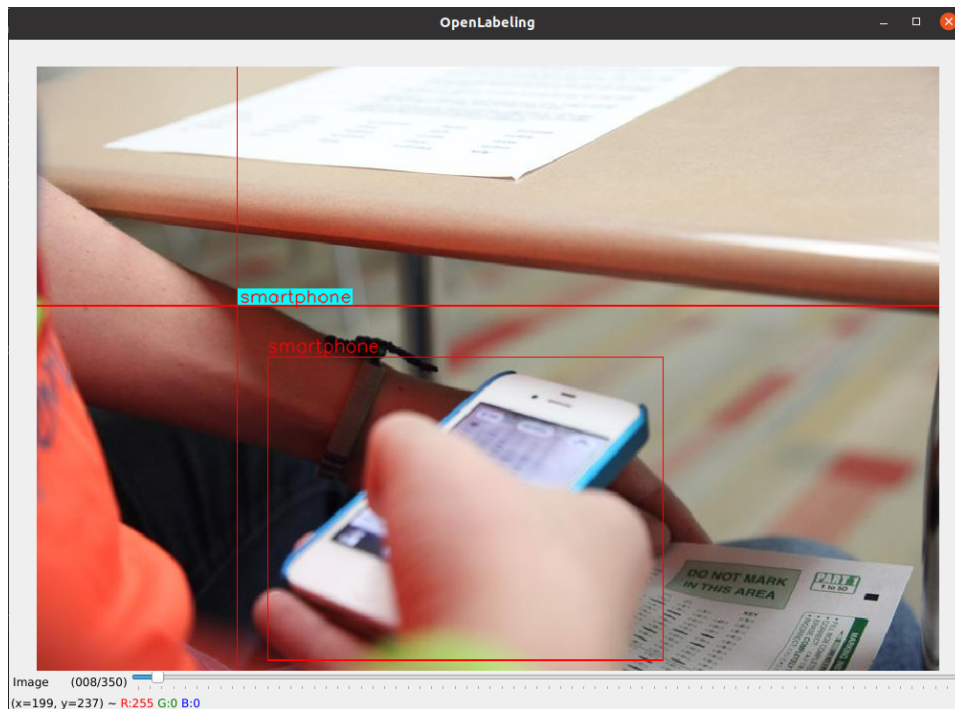
<https://drive.google.com/drive/folders/1vvfkIEhv4utoP8MYNMkm9iYjduj0CsuV?usp=sharing>

Подобные картинки могут быть использованы для распознавания смартфонов.

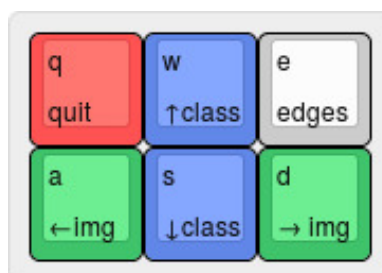
6. Перейдите к папке `main` и запустите скрипт:

```
python main.py
```

После чего у вас откроется окно с вашими изображениями



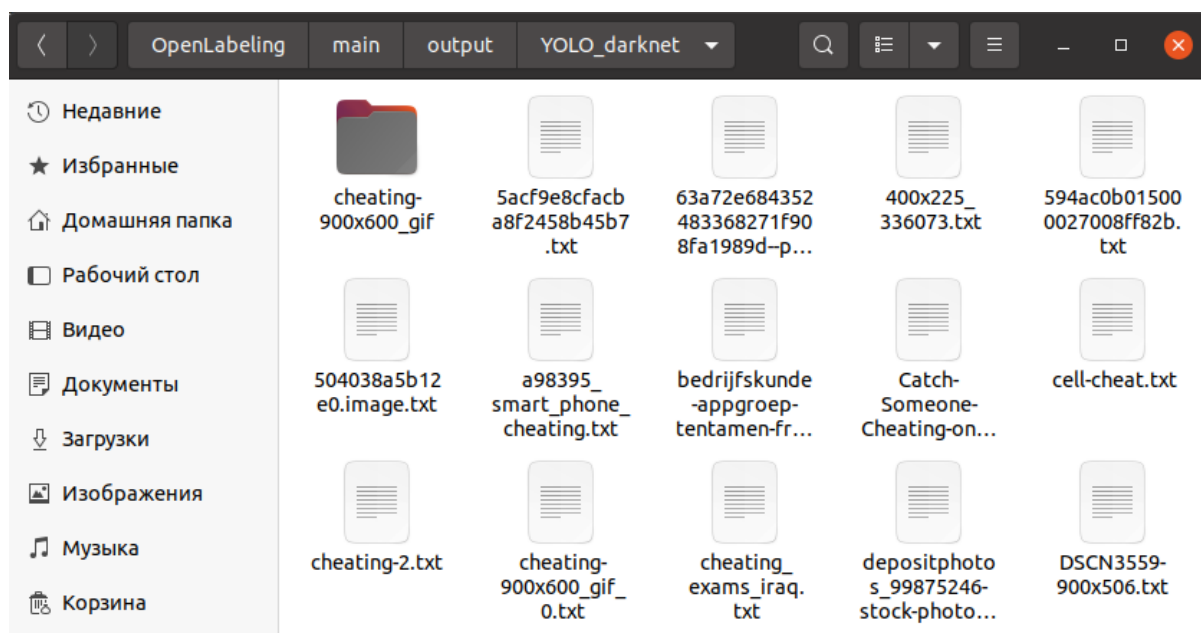
P.S. Для того чтобы перемещаться между изображениями и классами используйте следующие горячие клавиши.



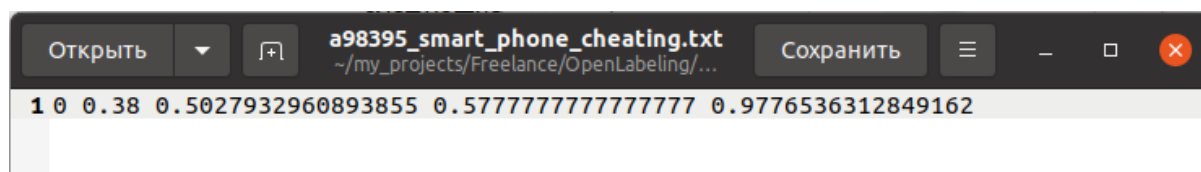
- h - Подсказки
- q - Закрыть окно
- e - Просмотр границ
- w - Предыдущий класс
- s - Следующий класс
- a - Предыдущее изображение
- d - Следующее изображение

7. КClick on the link below to know more about the labeling process and other software for it: ак только вы начнете помечать изображения, у вас должна появиться папка `output`, где содержатся папки `PASCAL_VOC` и `YOLO_darknet`.

Для следующего этапа распознавания будут использоваться данные второй папки. Данные `YOLO_darknet` выглядят следующим образом:



Данные текстовые файлы имеют одинаковые названия с соответствующими изображениями, будьте уверены что вы не переименовывали названия после процесса маркировки. Содержимое файлов выглядят следующим образом:



Как видите, файл содержит 5 значений

`<object-class> <x_center> <y_center> <width> <height>`.

Первое значение `<object-class>` — это идентификатор класса, как указано в файле `class_list.txt`.

Остальные 4 значения в файле `<x_center> <y_center> <width> <height>` — это координаты объекта.

## Источники

- ☐ **DATASET LABELING/ANNOTATION.** <https://medium.com/analytics-vidhya/image-dataset-labeling-annotation-bec3390eda2d>