Региональный этап чемпионата по профессиональному мастерству «Профессионалы» в 2024г.

Документация к API

написанного для нейросетевых моделей по определению и распознаванию автомобильных номеров

Выполнил: Руднев Никита Владимирович

Конкурсант работающий за компьютером №4

14.03.2024г.

# Содержание

Документация к API

[Содержание 2](#_Toc161326004)

[1. Получение изображения с помощью функции 3](#_Toc161326005)

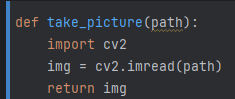
[2. Получение результата предсказаний модели 5](#_Toc161326006)

[3. API 7](#_Toc161326007)

[4. Установка API 9](#_Toc161326008)

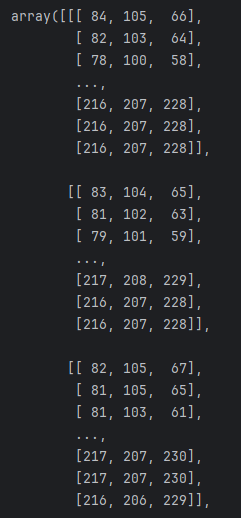
# Получение изображения с помощью функции

В начале работы с API к нашим моделям нужно было создать функцию, которая смогла бы получить фотографию для отображения и использования как данные, получаемые нашим кластером моделей. Такую функцию я назвал take\_picture. В коде она выглядит именно так:



Она принимает на вход путь к файлу изображения, импортирует cv2(python-opencv), функцией imread() библиотека читает наш файл по пути к картинке и после этого выводит получившиеся значения.

Такие значения выглядят вот так:



Это информация об цвете каждого пикселя в RGB градации. Такие данные потом можно либо вывести как картинку:

Либо импортировать в модель и получить распознавание номера. О чём наш следующий модуль документации.

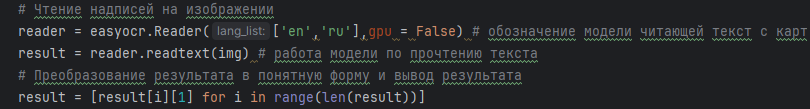
# Получение результата предсказаний модели

После загрузки изображения в код нам нужно внести в модель yolov5 данные из картинки, чтобы она распознала на ней автомобильный номер. Результаты распознавания модели выглядят как различного рода прямоугольники как на следующей картинке:

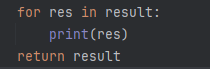
Из такого распознавания в коде можно вынести две координаты xyxy формата (x1,y1,x2,y2).

И используем эти данные, чтобы обрезать наше изображение именно на автомобильном номере. Пример такой обрезки:

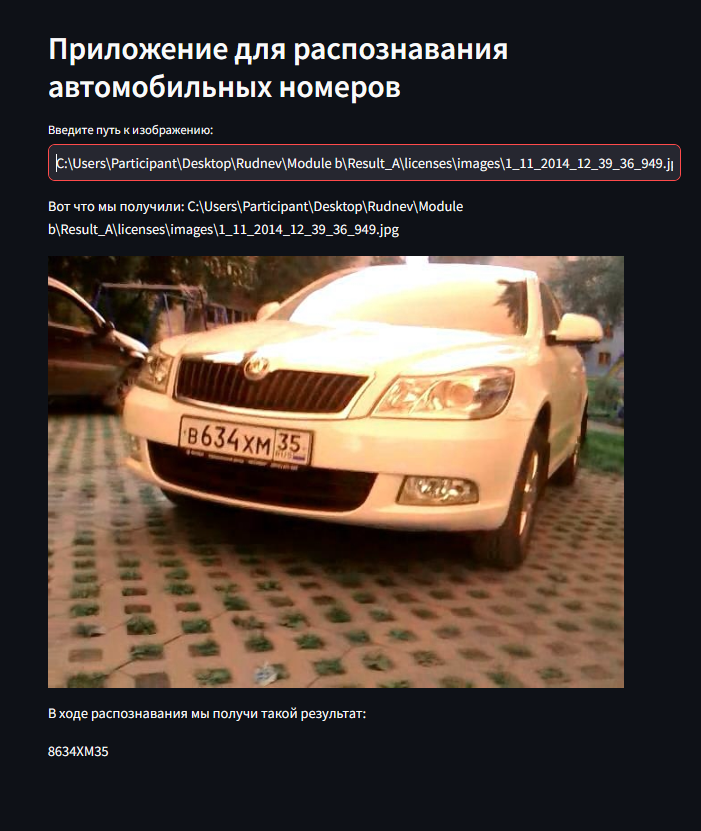
Далее обрезанная, распознанная лицензия переходит в модель чтения текста с автомобильного номера. Код:



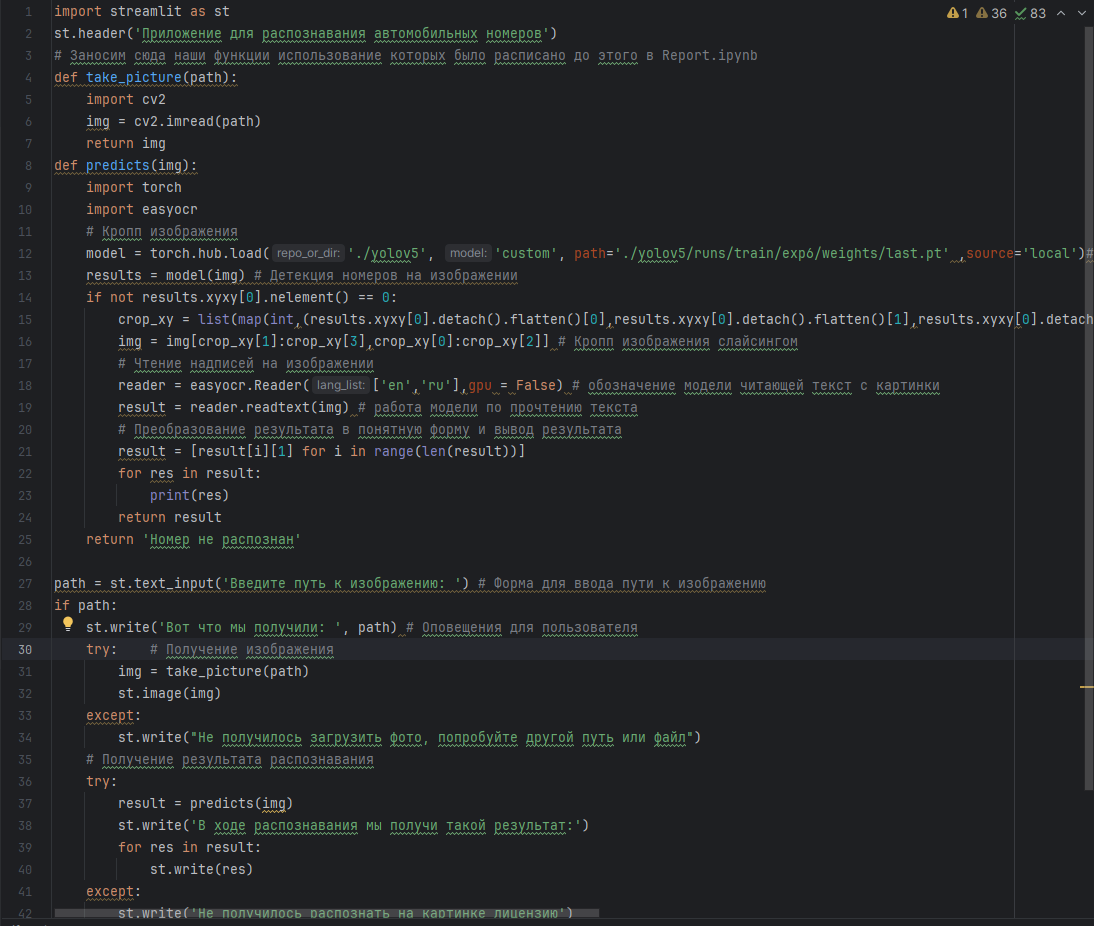
И с помощью функции readtext номер автомобиля читается и переходит дальше, преобразуется в читаемый вид и выводится функцией:



# API

Соединив две эти функции и написав с помощью библиотеки streamlit каркас веб приложения я получил полноценно API в которое мы вводим путь к изображению автомобиля и получаем на выход попытку распознавания номера этого автомобиля:

Код получившейся API:



Можно найти его в ./api/main.py

# Установка API

Для установки моего API нужно скачать папку api, располагающаяся в Module b. В ней находится директория yolov5 с моделью yolo, используемой для детекции номеров, main.py -–сам код API, который нужно будет запустить и requirements.txt – библиотеки, которые я использовал на всех модулях. Из этих библиотек некоторые нужные для API, но выделить именно нужные не имею возможность. Импортируемые библиотеки, которые нужно скачать:

pip install streamlit

pip install python-opencv (вроде(это cv2))

pip install torch

pip install easyocr

Для начала работы нужно создать или выполнять код в созданной среде python. В терминале с помощью команд cd переходим в папку api и скачиваем библиотеки с помощью pip как выше, либо с помощью команды pip install -r requirements.txt.

Далее в том же терминале прописываем команду streamlit run main.py, которая должна запустить мою API. Если потребует email можно смело в терминале нажать enter.

Если всё было сделано правильно и я не ошибаюсь в своих предположениях API должно открыться.