**Отчёт по производственной практике**

**1. Общие сведения**

* Хохряков Кирилл Юрьевич
* Международный Восточный Европейский Колледж, Информационные системы и программирование
* Дис.232.3/21
* 26.05-22.06
* АйТиЛаб
* Гусаров Фаддей Григорьевич
* Пашкина Любовь Владимировна

**2. Цель и задачи практики**

* Общая цель практики:
* Приобретение навыков разработки веб-приложений в реальных условиях
* Получение опыта разработки не только веб-приложений, но и использований ии
* Конкретные задачи:
  + Изучение технологического стека
  + Реализация функционала: Код с помощью которого через фото машины с номерным знаком, ии найдёт и сохранит номера автомобиля

**3. Описание выполняемой работы**

* Проект, в рамках которого велась работа:
  + AutoPlateID — система автоматического распознавания автомобильных номеров на основе компьютерного зрения и искусственного интеллекта. Проект позволяет детектировать и считывать номера транспортных средств с изображений в реальном времени, поддерживая российский стандарт номерных знаков.
* Разрабатываемый функционал
  + Проект AutoPlateID представляет собой решение для автоматического обнаружения и распознавания автомобильных номеров с изображений.
* Основные задачи:
  + Детекция номерных знаков – точное определение местоположения номеров на изображении с помощью нейросетей
  + Распознавание текста – извлечение символов с учетом российского стандарта (А123БВ 116)  
    Обработка изображений и видео – поддержка одиночных файлов и пакетной обработки
  + Экспорт данных – сохранение результатов в JSON с координатами и распознанным текстом
  + Визуализация – отрисовка bounding boxes и текста на выходных изображениях.
* Этапы выполнения
* 1. Анализ требований и проектирование
  + Определение формата входных/выходных данных
  + Выбор алгоритмов детекции и распознавания
  + Проектирование структуры кода
* 2. Детекция номерных знаков
  + Использование YOLOv8 для обнаружения автомобилей и номеров
  + Оптимизация модели для работы с российскими номерами
  + Улучшение качества детекции с помощью OpenCV (CLAHE, бинаризация)
* 3. Распознавание текста
  + Настройка EasyOCR для работы с кириллицей
  + Фильтрация шумов и обработка изображений перед распознаванием
  + Постобработка текста (приведение к стандартному формату)
* 4. Интеграция и обработка данных
  + Реализация пакетной обработки изображений из папки
  + Сохранение результатов в JSON
  + Визуализация результатов (bounding boxes + текст)
* 5. Тестирование и оптимизация
  + Проверка точности на реальных данных
  + Улучшение скорости обработки
  + Доработка под сложные случаи (плохое освещение, наклон номеров)

**4. Что получилось и что не получилось**

* Что было реализовано полностью
* Детекция автомобилей и номерных знаков
  + Успешно интегрирована YOLOv8 для обнаружения номеров даже на сложных фонах.
  + Реализована оптимизация области поиска (фокус на нижней части автомобиля).
* Распознавание текста
  + EasyOCR корректно работает с кириллицей и российским форматом (А123БВ 116).
  + Добавлена постобработка текста (удаление шумов, форматирование под стандарт).
* Обработка изображений
  + Поддержка пакетной обработки (изображения из папки).
  + Улучшение качества через CLAHE, бинаризацию и морфологические операции.
* Экспорт и визуализация
  + Результаты сохраняются в JSON с координатами и текстом.
  + Реализована отрисовка bounding boxes и подписей на изображениях.
* Интеграция
  + Готовый Python-скрипт с интерфейсом командной строки.
* Что не удалось завершить и по каким причинам
* Точность на низкокачественных изображениях
  + При плохом освещении или малом разрешении бывают ошибки в детекции и распознавании.  
    Причина: Ограничения стандартной YOLOv8 (без до обучения на специфичных данных).
* Распознавание наклонных номеров
  + Если номер повернут более чем на 15–20°, текст может не распознаться.  
    Причина: EasyOCR не всегда корректно обрабатывает сильные искажения.
* Поддержка видео в реальном времени
  + Реализована только обработка статичных изображений.  
    Причина: Требуется оптимизация для потокового видео (не хватило времени).

**5. Сложности и способы их преодоления**

* Технические сложности
* Проблема: Низкая точность YOLOv8 на российских номерах.

Решение:

* + Добавлена предобработка изображений (CLAHE, бинаризация).
  + Введен фильтр по геометрии (исключение слишком узких/широких bbox).
* Проблема: EasyOCR путал похожие символы (Р/Р, 8/В).

Решение:

* + Ограничение алфавита через allowlist.
  + Ручная постобработка (замена 0 на О, 3 на З при уверенности >80%).
* Проблема: Падение производительности на 4K-изображениях.  
  Решение:
  + Масштабирование до 1920x1080 с сохранением пропорций.
* Коммуникационные сложности
  + Уточнение ТЗ перед стартом разработки.

**6. Что было легко / что понравилось**

* Понравилось:
  + Работа с EasyOCR — библиотека хорошо документирована и быстро интегрируется.
  + Визуализация результатов через Pillow — простое рисование bbox и текста.
  + Обработка аргументов командной строки через argparse (удобно для тестирования).
* Далось легко:
  + Настройка YOLOv8 для детекции (благодаря готовым примерам Ultralytics).

**7. Новые знания и навыки**

* Освоенные технологии:
  + YOLOv8 — обучение работе с Ultralytics API, тонкая настройка детекции.
  + EasyOCR — настройка под кириллицу, кастомизация параметров распознавания.
  + OpenCV — методы улучшения изображений (CLAHE, адаптивная бинаризация).

**8. Выводы**

* Опыт практики показал:
  + Компьютерное зрение сильно зависит от качества данных.
  + Готовые модели (YOLO, EasyOCR) требуют дополнительной настройки под конкретную задачу.

**9. Приложения**

* <https://github.com/Nerchik12/-/blob/19bb4f631dbbd1e9b1f7078f386999bf39e320cd/AutoPlateID.py>