

Отчёт по производственной практике

Тема: Мониторинг судов в порту

Вид практики: Производственная практика (системы искусственного интеллекта)

Сроки: 19.01.2026 – 01.02.2026

Кафедра: МКиИТ

Руководитель: ассистент Тетерин Н.Н.

Студент: <ФИО>

Группа: <ГРУППА>

GitHub: <вставить ссылку на GitHub репозиторий>

Цель практики — освоить полный цикл разработки системы искусственного интеллекта для обработки изображений: от выбора архитектуры нейронной сети до внедрения предобученной модели в веб-приложение с визуализацией результатов, сохранением истории и генерацией отчётов.

1. Постановка задачи

Вариант задания: «Мониторинг судов в порту». Требуется обнаруживать суда на изображениях/видео/видеопотоке с камеры, отображать результаты детекции в веб-интерфейсе, сохранять историю запросов и формировать отчёты (PDF/Excel).

2. Выбор архитектуры нейронной сети

Для детекции выбран Ultralytics YOLO (семейство YOLOv8) как современная архитектура одноэтапной детекции, обеспечивающая баланс точности и скорости. Используется предобученная модель на датасете COCO; класс boat трактуется как «судно». При наличии собственных весов (например, yolo26.pt) приложение поддерживает их подключение.

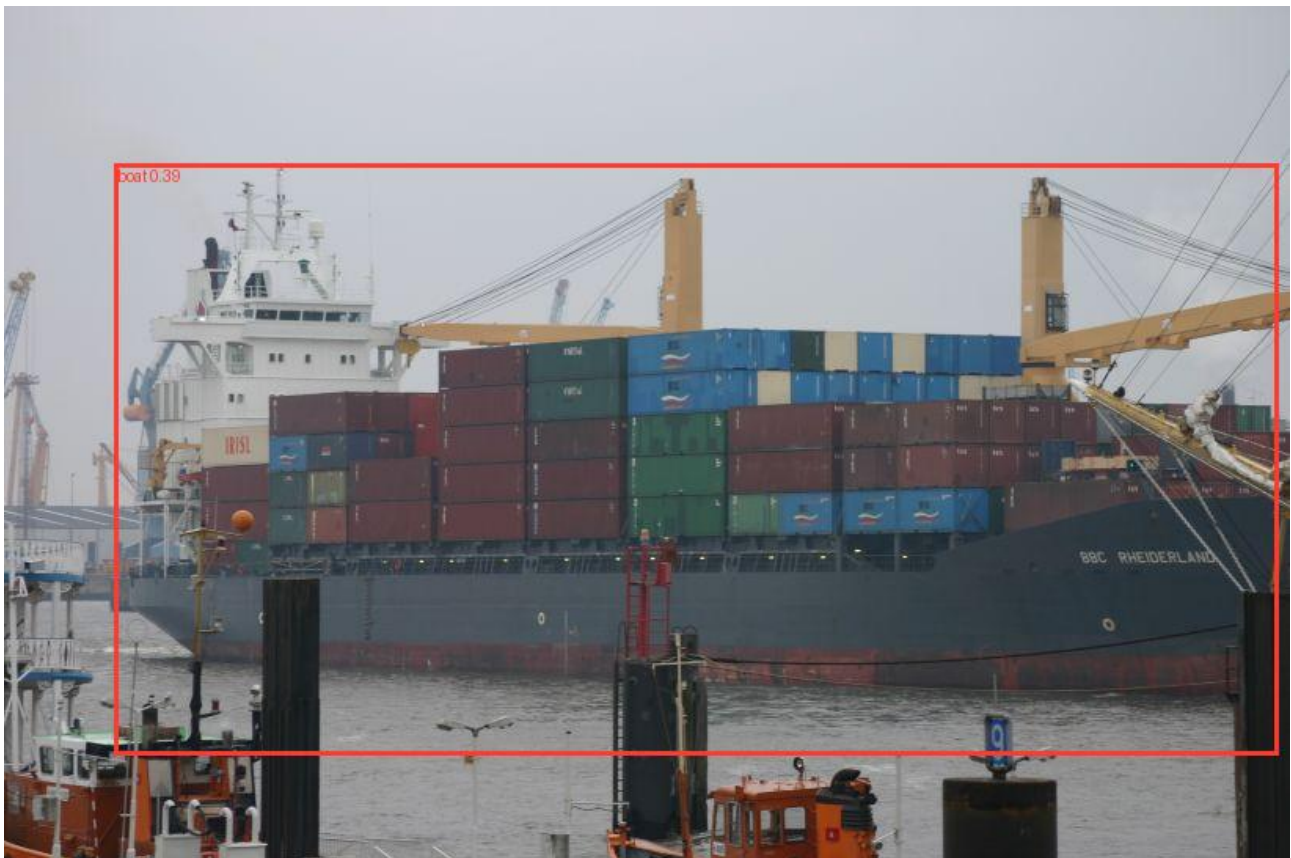
3. Реализация веб-интерфейса

Веб-интерфейс реализован на Gradio (работает в браузере). Поддерживает: загрузку изображений, загрузку видео, получение снимка с веб-камеры, кнопку запуска обработки, вывод результата с bboxes, а также вкладку статистики/истории и генерацию отчётов.

4. Интеграция предобученной модели

Инференс реализован модулем `ship_monitoring.inference`: функция `predict_image` выполняет детекцию по изображению и сохраняет оригинал/результат, `predict_video` обрабатывает видео (можно брать каждый N-й кадр), строит аннотированное видео и статистику.

Пример результата (демо)



5. История запросов и статистика

История запросов хранится в JSON (data/history.json). Для каждого запроса сохраняются: время, тип входа, параметры модели, количество судов, пути к файлам результатов и время обработки. Во вкладке «История и статистика» отображаются KPI и графики (Plotly): динамика количества судов и распределение по запросам.

6. Генерация отчётов

Для каждого запроса формируются отчёты: Excel (pandas/openpyxl) и PDF (reportlab). Также доступен сводный отчёт по истории. Эти функции используются прямо из веб-интерфейса.

7. Тестирование

Добавлены автотесты pytest: проверка истории (JSON round-trip), генерации отчётов (PDF/Excel) и расчёта KPI. Тесты выполняются командой: `python -m pytest`.

8. Инструкция запуска

1) Создать виртуальное окружение: `python -m venv .venv` 2) Установить зависимости: `.venv\Scripts\python -m pip install -r requirements.txt` 3) Запуск приложения: `.venv\Scripts\python -m ship_monitoring` 4) Открыть интерфейс в браузере и выполнить несколько запросов (изображение/видео/камера).