|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное автономное образовательное учреждение высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Информатика, искусственный интеллект и системы управления

КАФЕДРА Системы обработки информации и управления

**ОТЧЕТ**

***ПО ДОМАШНЕМУ ЗАДАНИЮ***

***НА ТЕМУ:***

**Распознавание и классификация изображений с помощью модели нейронной сети YOLOv7**

Студент \_\_ИУ5-22М\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_П.А. Бибиков\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_А.И. Канев\_\_**

(Подпись, дата) (И.О.Фамилия)

2025 г.

ВВЕДЕНИЕ

Распознавание и классификация изображений являются фундаментальными задачами компьютерного зрения, находящими применение в самых разнообразных сферах, от систем безопасности до медицинской диагностики и автономного транспорта. Среди множества подходов к решению этих задач особое место занимают сверточные нейронные сети (Convolutional Neural Networks, CNNs), а одной из наиболее эффективных и популярных архитектур для обнаружения объектов в реальном времени является YOLO (You Only Look Once).

YOLOv7 – это передовая итерация в семействе моделей YOLO, представляющая собой одноэтапный детектор объектов (one-stage object detector). Ключевая концепция YOLO заключается в том, что модель обрабатывает все изображение целиком за один проход нейронной сети, одновременно предсказывая ограничивающие рамки (bounding boxes) для всех объектов на изображении и классифицируя их. Это кардинально отличает YOLO от двухэтапных детекторов (например, R-CNN и его производных), которые сначала генерируют набор потенциальных регионов-кандидатов, а затем классифицируют каждый из них. Такой "однопроходный" подход позволяет YOLO достигать очень высокой скорости обработки, что критически важно для приложений реального времени.

**СОДЕРЖАНИЕ**

[Задание 4](#_Toc197720572)

[Выполнение работы 5](#_Toc197720573)

[1. Разметка датасета 5](#_Toc197720574)

[2. Обучение модели 7](#_Toc197720575)

[3. Разработка SPA-приложения 12](#_Toc197720576)

[Вывод 14](#_Toc197720577)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 15](#_Toc197720578)

# **Задание**

Сохраняется тема ДЗ-1. Требуется разработать SPA приложение для object detection ваших классов. Разметить изображения набора данных и обучить модель Yolo.

Классы:

1. Кабриолет
2. Пикап
3. Седан

**Выполнение работы**

1. **Разметка датасета**

Требуется скачать и установить специальный фреймворк для разметки изображений CVAT.

Создали проект «PHC\_hw2». В нем создали 3класса:

* Кабриолет– фиолетовые метки;
* Пикап – оранжевые метки;
* Седан – зеленые метки.

Далее в проекте создали 3 задачи – по одной на каждый класс. Подготовили выборку из 100 изображений каждого класса.

Выбираем такие изображения, где представлен единичный или несколько объектов одновременно, желательно, не занимающий все изображение. Все изображения сконвертировали в .jpg.

В домашней работе будут классифицироваться женщины разных возрастов: девочки, мамы, бабушки. Выборка для каждого из классов состоит из 100 изображений. Небольшая подвыборка из исходных изображений показана на рисунке 1. Пример размеченного изображения представлен на рисунке 2.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1 — Исходные изображения

Изображение выглядит как Наземный транспорт, транспортное средство, колесо, текст

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2 — Пример размеченного изображения

1. **Обучение модели**

Обучение модел Yolo производилось с различными гиперпараметрами: размер батча менялся 8/16, количество эпох 50/100/30/75. При обучении удалось достигнуть следующих результатов, представленных на рисунках 3-6.

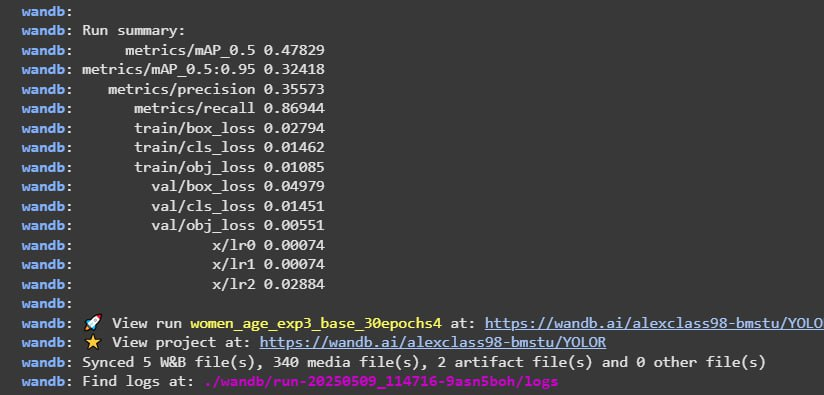


Рисунок 3 — модель: размер батча 8, эпох 30

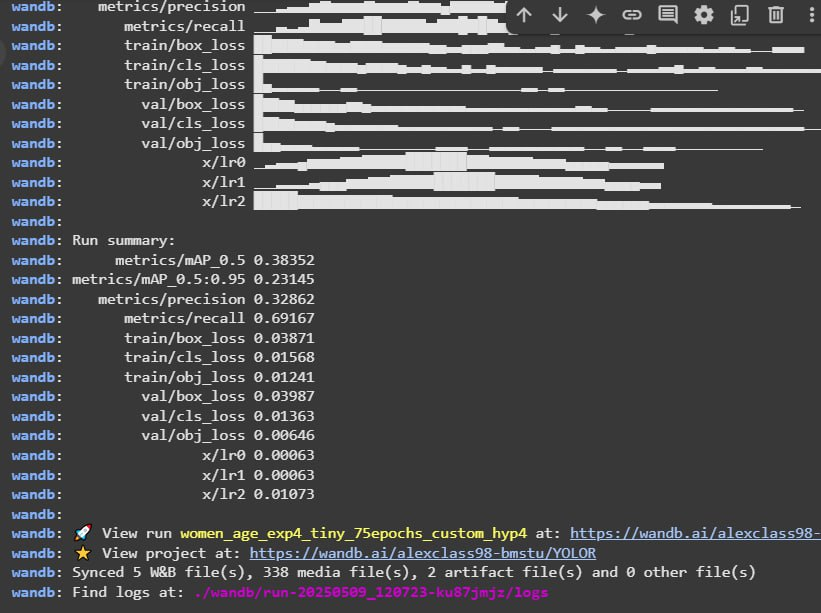


Рисунок 4 — модель: размер батча 16, эпох 75

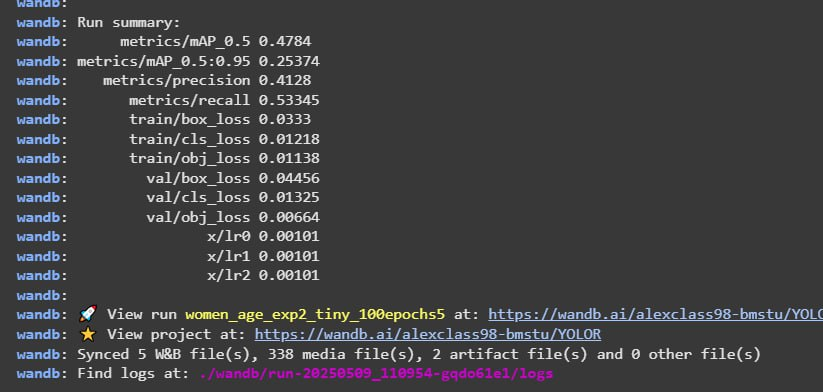


Рисунок 5 — модель: размер батча 16, эпох 100

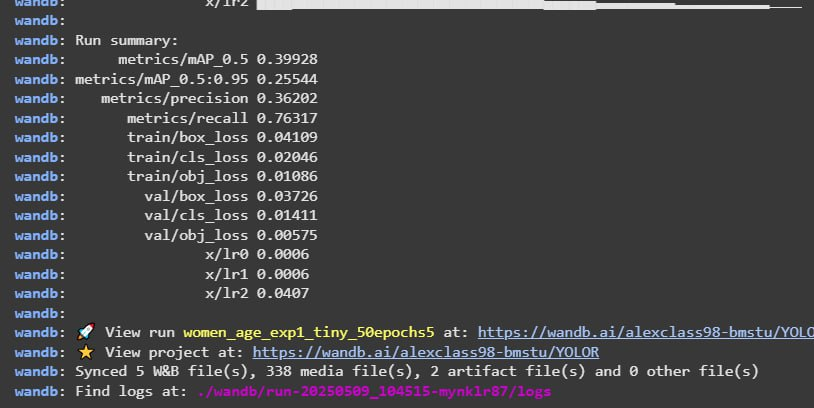


Рисунок 6 — модель: размер батча 16, эпох 50

На рисунках 7-8 представлены метрики для лучшей модели.

Изображение выглядит как диаграмма, текст, линия, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 7 — метрика F1

Изображение выглядит как текст, диаграмма, линия, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 8 — метрика PR

Рассмотрим метрики качества для каждого класса, который может предсказать наша модель. Ключевыми метриками для сравнения служили средняя точность (mAP) при различных порогах IoU (Intersection over Union), а также общие показатели точности (Precision) и полноты (Recall) для всех классов вместе. Кроме того, детально рассматривалась производительность моделей на каждом из целевых классов: кабриолет, пикап и седан, для которых вычислялся mAP при IoU=0.5. Это позволяет понять, насколько хорошо каждая модель справляется с распознаванием конкретных категорий объектов. Метрики качества на обучающей выборке представлены в сводной таблице 1.

Таблица 1. Сводная таблица результатов всех экспериментов

| **№** | **Модель** | **Эпохи** | **Гиперпараметры** | **mAP@.5 (all)** | **mAP@.5:.95 (all)** | **Precision (all)** | **Recall (all)** | **кабр.** | **пик.** | **седан** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | YOLOv7-tiny | 50 | Стандарт | 0.399 | 0.255 | 0.362 | 0.763 | 0.357 | 0.484 | 0.357 |
| 2 | YOLOv7-tiny | 100 | Стандарт | **0.478** | 0.254 | **0.413** | 0.533 | 0.298 | 0.458 | **0.680** |
| 3 | YOLOv7 | 30 | Стандарт | **0.478** | **0.324** | 0.356 | **0.869** | 0.407 | 0.515 | 0.513 |
| 4 | YOLOv7-tiny | 75 | Кастомные | 0.384 | 0.231 | 0.329 | 0.692 | 0.197 | **0.542** | 0.411 |

Лучшими моделями можно выделить модель 3 (YOLOv7 с 30 эпохами обучения) и модель 2 (YOLOv7-tiny со 100 эпохами обучения). У модели 3 самый высокий mAP@.5:.95 (all) – 0.324), самый высокий Recall (all) – 0.869 и лучший результат для класса кабриолет – 0.407). Эта модель выглядит как наиболее сбалансированная и сильная модель в целом, однако YOLOv7 достаточно тяжелая в ранении с YOLOv7-tiny. Модель 2 имеет самый высокий Precision (all) – 0.413 и лучший результат для класса седан – 0.680), а также один из лучших mAP@.5 (all) – 0.478. Так как модель 2 является более легкой, то сделаем выбор в ее пользу. Марица ошибок для лучшей модели (YOLOv7-tiny со 100 эпохами обучения) представлена на рисунке 9.

Изображение выглядит как снимок экрана, Прямоугольник, текст, прямоугольный

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 9 — Матрица ошибок для YOLOv7-tiny со 100 эпохами обучения

1. **Разработка SPA-приложения**

Обученная модель была экспортирована в формате ONNX и загружена в SPA. Результаты работы модели в веб-приложении для разных случаев представлены на рисунках 10 – 12.

Изображение выглядит как транспортное средство, Наземный транспорт, колесо, транспорт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 10 — Классификация кабриолета на изображении

Изображение выглядит как шина, транспортное средство, колесо, Наземный транспорт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 11 — Классификация пикапа на изображении

Изображение выглядит как Наземный транспорт, транспортное средство, колесо, шина

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 12 — Классификация седана на изображении

Изображение выглядит как текст, графический дизайн, снимок экрана, Бренд

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 13 — Классификация нескольких объектов на одном изображении

Вывод

В результате выполнения домашнего задания были закреплены навыки, освоенные в процессе выполнения лабораторных работ, а также освоены новые, необходимые для интеграции модели в веб-приложение. В ходе выполнения домашнего задания были размечены 300 изображений (по 100 на каждый класс). Экспортирована и интегрирована в веб-приложение модель Yolo, обученная на размеченных данных.

Лучшей моделью была выбрана YOLOv7-tiny со 100 эпохами обучения.

Успешнее всего она справляется с распознаванием ребёнка и бабушек. Это связано с тем, что дети часто имеют более отчетливые визуальные характеристики по сравнению со взрослыми (например, меньший размер, другие пропорции тела, особенности лица, характерная одежда или атрибуты вроде игрушек). В свою очередь бабушки также имеют достаточно четкие и уникальные визуальные характеристики, такие как седые волосы, морщины, определенные прически, часто пучки или короткие стрижки. Хуже всего распознается на изображении класс “мамы”. Это может быть связано с тем, что это очень широкий возрастной диапазон и большое разнообразие стилей, внешности, телосложения. Молодые мамы могут быть похожи на девушек-подростков. Мамы старшего возраста могут начать приобретать некоторые черты, характерные для класса "бабушки", что может приводить к путанице.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Методические указания к лабораторным работам. Источник: <https://github.com/iu5git/Deep-learning>
2. Материал видео-лекций по предмету “Разработка нейросетевых систем”. Источник: [Deep learning - YouTube](https://www.youtube.com/playlist?list=PLLELLTvDgUQ_d9eUj_3XVpAdGByuU37kT)