Команда 49

Тема: Детекция, трекинг пешеходов

Куратор: Соборнов Тимофей

Состав:

- Брежнева Ангелина
- Зиязиев Ильназ
- Петров Андрей
- Янышев Дмитрий

Постановка задачи

Строим сервис **автоматического обнаружения пешеходов на видео**, расчета пешеходного трафика, анализа демографических характеристик (пол, возраст) с использованием моделей глубокого обучения.

Этапы проекта:

1. Решение задачи детекции
   a. Эксперименты с базовым подходом (Resnet + SVM)
   b. Применение продвинутых подходов (DL-модели)
2. MVP сервиса для детекции
3. Решение задачи трекинга
4. Доработка сервиса под трекинг и потоковую обработку

Специфика данных, EDA

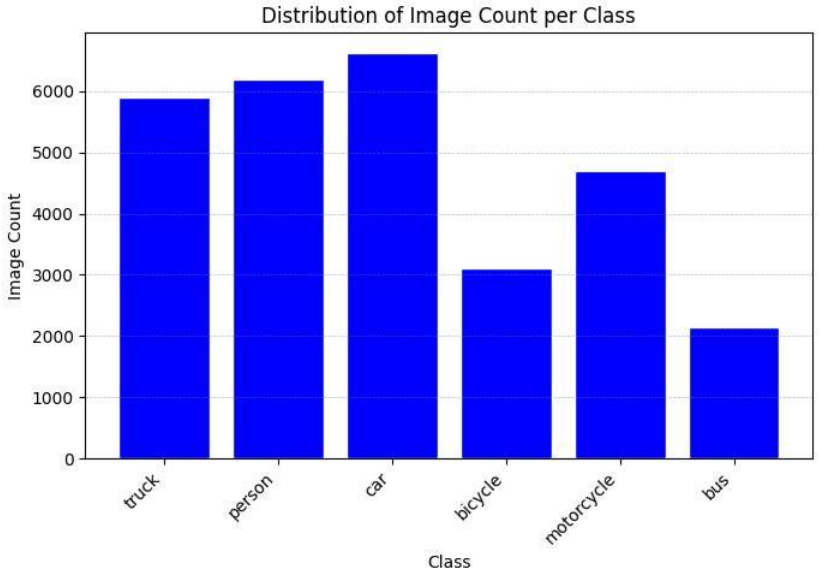
Датасет для обучения взят с **roboflow**, содержит 7 тыс. картинок с 6 классами объектов, включая интересующий нас класс "person".

Картинки — фотографии улиц и находящихся на них объектов с высоты камер видеонаблюдения.

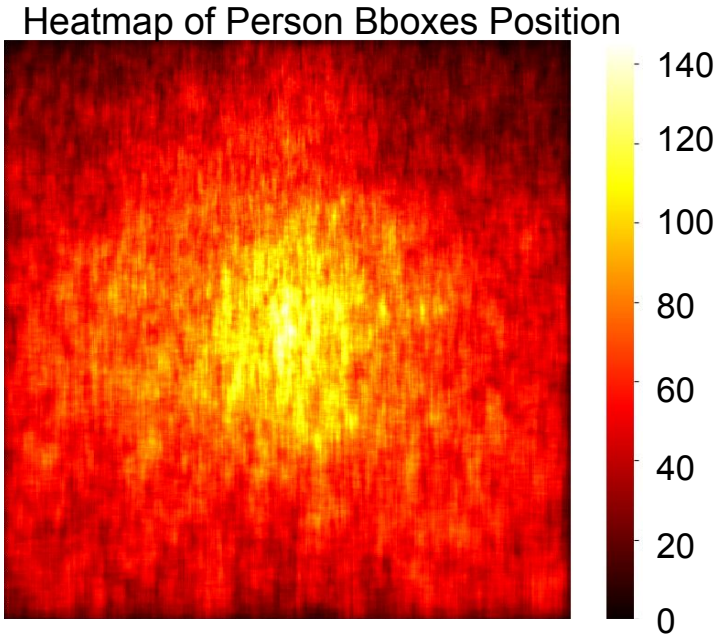
пример из датасета с большим количеством людей

Специфика данных, EDA

Преимущества



датасет хорошо сбалансирован по числу изображений на класс



таргет-класс расположен преимущественно по центру изображений

Специфика данных, EDA

Сложности

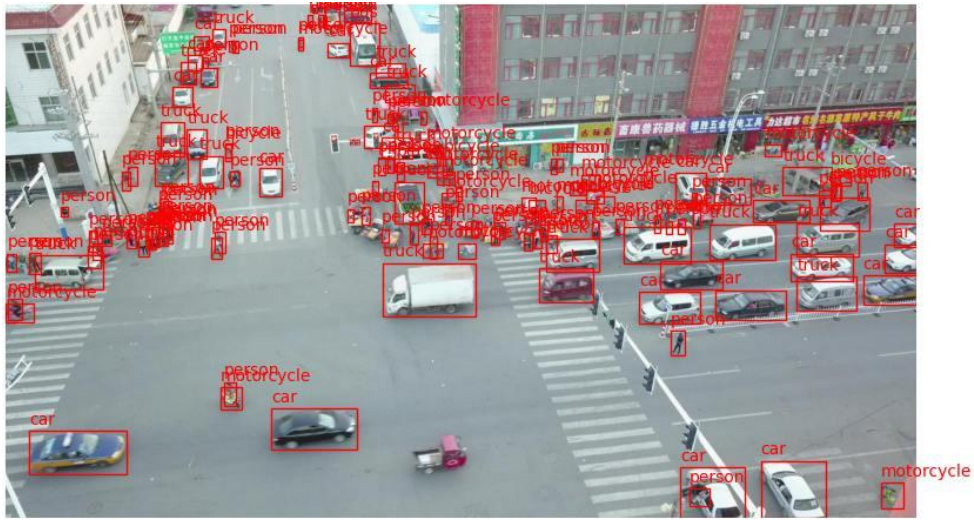
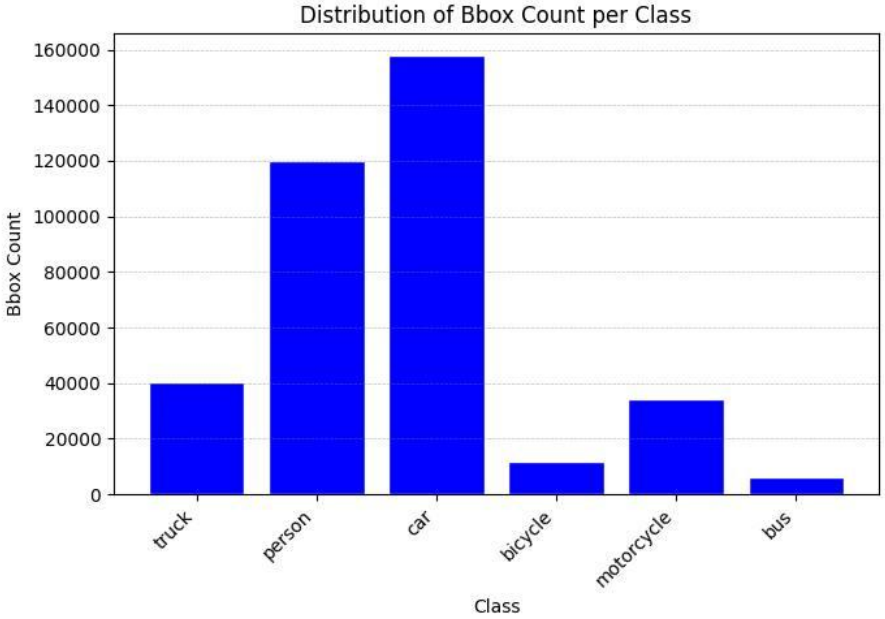


Фото сделаны с высоты:

- фигуры людей мелкие и плохо различимы
- нетривиальный ракурс, предобученные модели могут выдавать качество ниже ожидаемого
- на многолюдных снимках фигуры перекрываются, это создает сложности для детекции

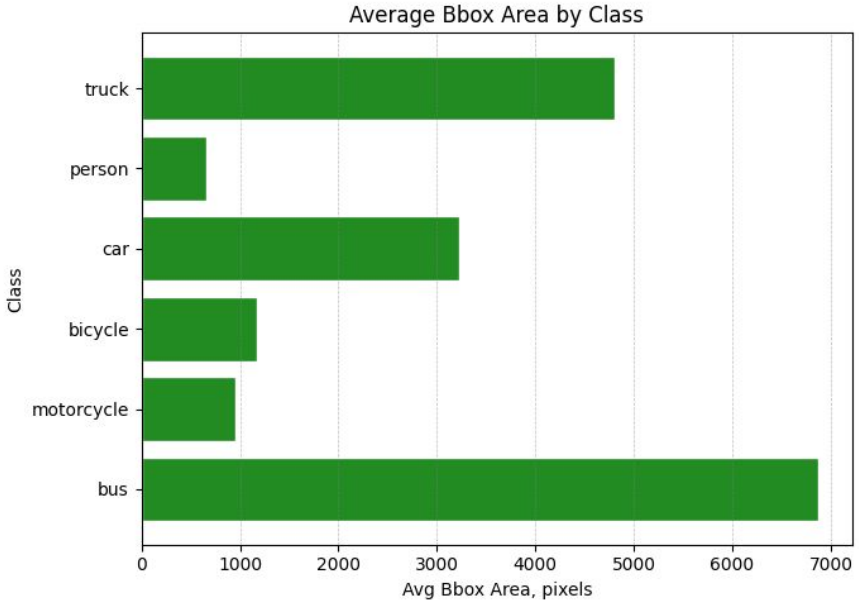
Специфика данных, EDA

Сложности



таргет-класс "person" не является максимальным по числу bbox на фото

таргет-класс "person" занимает наименьшую площадь среди всех классов в датасете



Детекция: бейзлайн

ResNet + SVM — решаем задачу детекции одного целевого класса

Обучающая выборка	Модель	Параметры	Метод извлечения регионов	mIoU (>= 0.5)
bboxes	SVM	kernel = ‘linear’, C=1	KMeans	0.1402
bboxes	SVM	kernel = ‘linear’, C=1	Selective Search	~0.0000
bboxes + kmeans на трейне	SVM	kernel = ‘linear’, C=1	KMeans	0.1537
bboxes + kmeans на трейне	SVM	kernel = ‘rbf’, C=1, class_weight= ‘balanced’	KMeans	0.1686

mIoU для центрального бибокса — 0.0006

Детекция: область интереса (бейзлайн)

- **Sliding Window:** подача участков фиксированного окна в классификатор. Не наш вариант: bbox с людьми маленькие —> алгоритм становится слишком затратным
- **KMeans:** кластеризуем пиксели картинки в группы; получившиеся группы — области интереса. **Реализован в бейзлайне**
- **Selective Search:** выделяем сегменты изображения, объединяем похожие. Не перформит (низкий масштаб целевого класса, специфика датасета): mIoU —> 0

Улучшение детекции: DL-модели

Решаем задачу детекции всех классов датасета (люди + транспорт)

Модель	Аугментации	Параметры	Качество	Время обучения
Detectron2 + faster_rcnn_X_101_32x 8d_FPN_3x	default	lr 0.00025 max_iter 1000 batch 4	mAP IoU 0.50: 0.402 mAP IoU 0.50-0.95: 0.089	~30 минут
yolo12	default	imgsz 640 epoch 100 batch 12	mAP IoU 0.50: 0.516 mAP IoU 0.50-0.95: 0.322	~7 часов
yolo12	crop, mosaic	imgsz 960 epoch 46 batch 4	mAP IoU 0.50: 0.597 mAP IoU 0.50-0.95: 0.380	~8 часов
yolo12	crop, mosaic	imgsz 640 epoch 100 batch 12	mAP IoU 0.50: 0.532 mAP IoU 0.50-0.95: 0.333	~8 часов
Deformable-DETR	default	lr 0.0001 epoch 10 batch 2	mAP IoU 0.50: 0.154 mAP IoU 0.50-0.95: 0.065	~5 часов
RT_DETR R50	default	in progress	in progress	in progress

DL-модели дают кратное увеличение качества (или как минимум сравнимое из коробки), даже с учетом детектирования нескольких классов

Next Steps

- Повысить качество детекции —> продолжаем эксперименты с YOLO

- Переход от детекции к трекингу —> интеграция алгоритмов трекинга (Deep SORT, ByteTrack), настройка ассоциации между кадрами

- Разработка формата выходных данных —> статистика проходимости (количество уникальных пешеходов, временные метки)

- Доработка MVP сервиса для работы с видео —> обработка видеопотока в реальном времени

- Обучение второй модели поверх детекции (nice to have) —> определение пола пешехода по атрибутам