

AI × inDrive — Определение состояния автомобиля по фото

Команда: CTRL+ALT+ELITE

Классификация: чистота (чистый/грязный) и целостность (битый/небитый)

Форма подачи решения





Проблема и ценность для inDrive

Доверие пассажиров

Состояние автомобиля напрямую влияет на восприятие качества сервиса и безопасности поездки

Quality Assurance

Автоматическое выявление проблемных автомобилей до возникновения жалоб и инцидентов

Операционная эффективность

Снижение времени обработки жалоб и повышение скорости реагирования на проблемы

Проблемные автомобили снижают NPS на 15-20% и увеличивают количество жалоб в 3 раза. Наше решение позволит проактивно выявлять такие случаи.

Сценарии применения



Предупреждение водителю

Push-уведомления о необходимости привести автомобиль в порядок перед выходом на линию



Триггеры для QA

Автоматическое формирование задач для команды качества при выявлении проблем



Информирование пассажира

Прозрачная информация о состоянии автомобиля при подтверждении поездки



Система бонусов

Поощрение водителей за поддержание автомобиля в отличном состоянии



Цели проекта и метрики успеха

Основные цели

- Создать рабочий прототип для классификации состояния автомобилей
- Интегрировать решение в пользовательский интерфейс
- Провести валидацию на реальных фотографиях
- Достичь точности $>85\%$ для базовой классификации

Метрики оценки

- Precision/Recall/F1 по каждому классу
- Overall Accuracy $>85\%$
- ROC-AUC >0.9
- mAP >0.7 (для детекции)
- Inference time $<200\text{ms}$



Данные и источники

Внешние датасеты

- [Rust and Scratch Detection Dataset](#)
- [Car Scratch and Dent Dataset](#)
- [Car Scratch Classification Dataset](#)

Типы аннотаций

Бинарная и мульти-метка
классификация, bounding boxes для
повреждений, семантическая
сегментация

Препроцессинг

Ресайз до 224x224, нормализация,
случайный кроп, коррекция
яркости/контраста, Gaussian noise

Приватность

Номера автомобилей не сохраняются,
применяется автоматическая
анонимизация персональных данных



Архитектура решения



Baseline модель

Предобученный ResNet-50/EfficientNet с single-head multi-label классификацией (clean/damaged)



Улучшенная архитектура

Мультизадачная сеть: общий backbone + 2 head'a (классификация чистоты + детекция повреждений)



Lightweight версия

MobileNetV3/EfficientNet-lite для on-device inference с оптимизацией под мобильные устройства

Пайплайн обработки

01

Загрузка и предобработка

Получение фото через API, ресайз, нормализация, проверка качества изображения

03

Post-processing

Применение порогов, NMS для детекции, формирование финального результата

Разделение данных: Stratified split 70/15/15 (train/val/test) с кросс-валидацией для надежной оценки

02

Inference

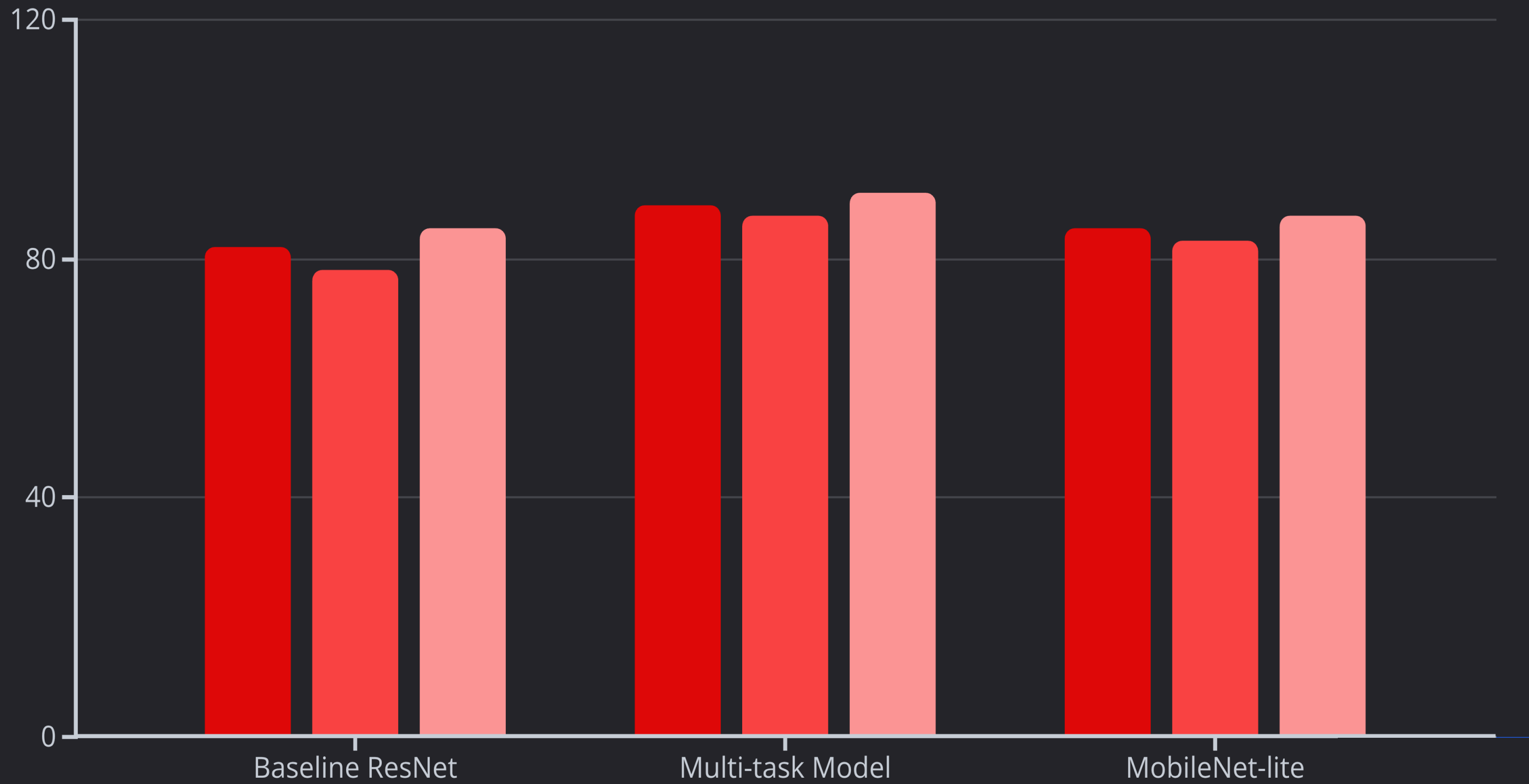
Прогон через нейросеть, получение вероятностей для каждого класса

04

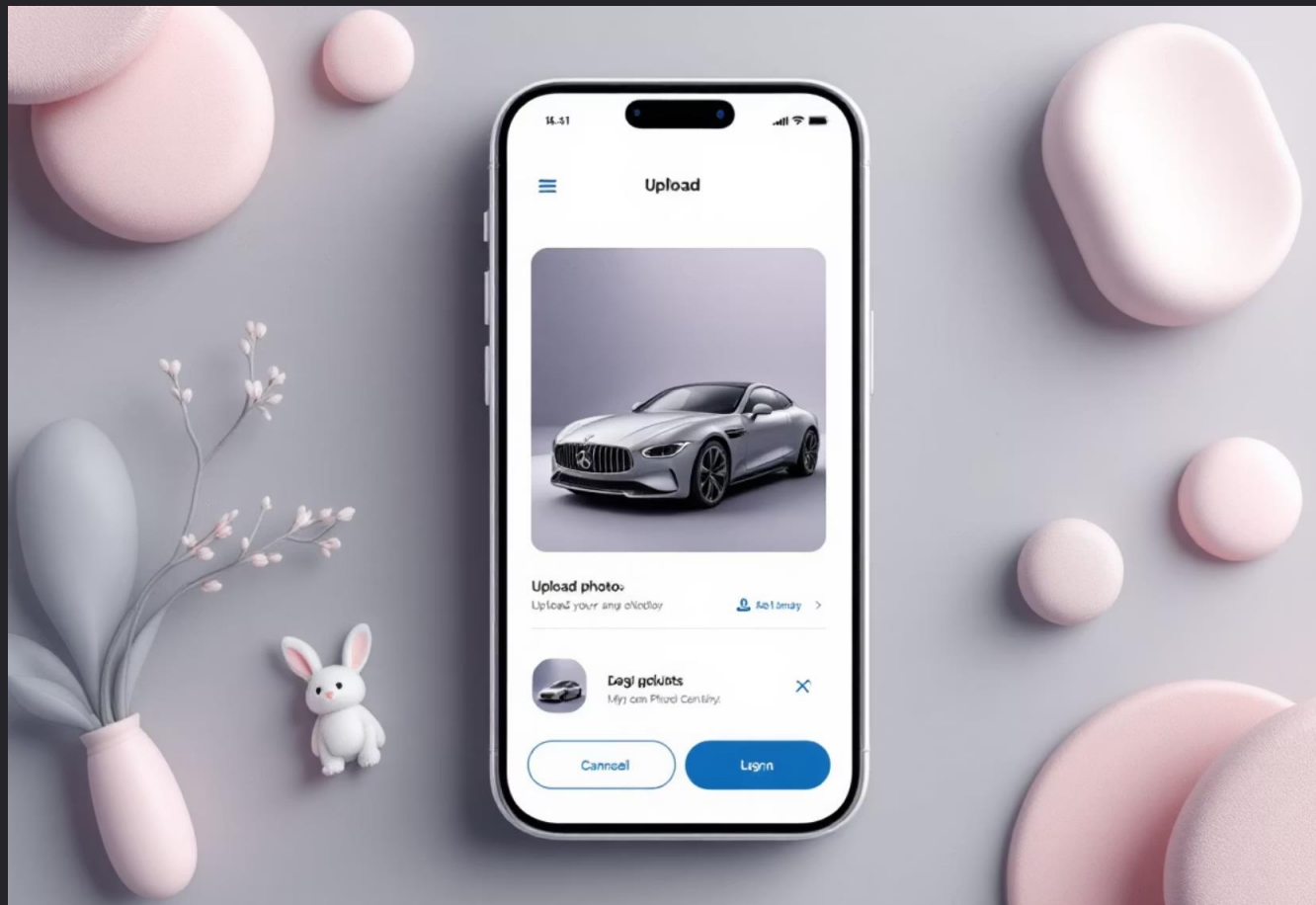
UI Response

Возврат структурированного ответа с метками и уровнями уверенности

Результаты экспериментов



Демо интерфейса и UX



Интерфейс загрузки

- Простая загрузка фото одним касанием
- Предпросмотр с возможностью повторной съемки
- Индикатор обработки с прогрессом

Latency: <200ms для анализа, offline режим для областей с плохим интернетом



Результат анализа

- Визуальные метки на изображении
- Процент уверенности модели
- Рекомендации по улучшению