Audion Traffic Counter

English · Руководство

Содержание

- Зачем
- Как устроено
- Что внутри
- Какие записи берёт
- Запуск
- Что ставится и что дают гигабайты
- С Claude и Codex
- Чего программа не делает
- Из чего сделано

Считает машины по записям обычных камер: сколько легковых, автобусов, грузовых и мотоциклов сделали каждый манёвр. Линии счёта проводятся мышью прямо на кадре, записи ставятся в очередь, и очередь считается сама — хоть всю ночь. Счёт идёт на вашей машине: записи никуда не отправляются.

Зачем

Обследование транспортных потоков — это часы записи с каждой точки, которые кто-то пересматривает и считает вручную. Программа делает этот счёт сама: специалист проводит линии на кадре каждой записи и возвращается к готовым таблицам.

Она рассчитана на то, что есть на деле: старые записи низкого разрешения с бытовых камер, десятки и сотни файлов подряд.

Как устроено

- **Линия** проводится поперёк дороги и считает машины, которые пересекают её по стрелке. Стрелка стоит в середине линии и разворачивается одной кнопкой.
- **Ворота** — для перекрёстка и кольца: по ним программа сводит въезд и выезд одной машины в манёвр «откуда → куда».
- **Зона** увеличивает дальнюю часть кадра, где машины занимают несколько пикселей.
- У каждой записи расстановка своя. По записям ходят кнопками под видео; линии соседней записи копируются одной кнопкой.
- Каждый счёт сохраняется отдельным прогоном: прогоны разных моделей стоят в одной таблице, рядом с ручным счётом, если он есть.

Что внутри

Раздел	Что делает
ЗАПИСИ	Список видео: добавить файлы или папку, отметить, что считать. Файлы остаются на месте
КАМЕРА	Видео с односторонними и двусторонними линиями, воротами и зонами поверх него (у зоны может быть свой детектор — например, для машин за отбойником); переход по записям; FFplay с линиями
СЧЁТ	Модель (YOLO26N … YOLO26X), на чём считать, сколько записей разом, очередь, пробный прогон, ход счёта
ИТОГИ	История прогонов, таблица по манёврам и классам с общим числом машин, сравнение прогонов, сверка с ручным счётом отчёт по всем записям файлами Word и Excel и листы с картинкой каждого грузового, автобуса, мотоцикла или легковой по сечению
ОБУЧЕНИЕ	Модель классов (легковые, грузовые, автобусы): одна идёт с программой и включена с самого начала, свою можно обучить под ваши камеры; и вырезки спорных машин, которым класс ставится одной кнопкой
УСТАНОВКА	Видеокарты машины и кнопка, которая ставит ровно то, чего ей не хватает
НАСТРОЙКИ	Как работает сама программа: скорость счёта (сколько записей разом, потоки, приоритет) и где хранится файл проекта с линиями записей

Какие записи берёт

Файлы `.avi`, `.mp4`, `.mov`, `.mkv`, `.mts`, `.m2ts`, `.mpg`, `.wmv`, `.ts`, `.webm` с обычными кодеками: H.264, H.265, AV1, MJPEG, MPEG-2, MPEG-4. Файлы с другими расширениями в список не попадают. Проверена программа на записи MJPEG AVI 640×480, AV1 — на коротком пробном ролике; остальные форматы читают те же библиотеки, но на записях они не испытаны. Чересстрочную запись (DV 720×576) перед счётом нужно перевести в прогрессивную.

Запуск

Запустите `Start.exe`, откройте раздел УСТАНОВКА и нажмите **УСТАНОВИТЬ НУЖНОЕ**: программа скажет, что скачает под вашу машину, и спросит, потом поставит свой Python, PyTorch под вашу видеокарту, TensorRT для видеокарты RTX, библиотеку распознавания и FFmpeg в свою папку. В Windows ставится одно: Microsoft Visual C++ Redistributable, который нужен PyTorch, — если его там нет. Дальше — по руководству.

Основной путь счёта — видеокарта NVIDIA серии RTX через TensorRT: получасовая запись 640×480 на RTX 5070 считается около пяти минут. Видеокарта NVIDIA GTX считает через PyTorch (CUDA), в несколько раз медленнее. Без NVIDIA программа считает через DirectML на любой видеокарте с DirectX 12, ещё медленнее.

Что ставится и что дают гигабайты

Любой машине ставятся свой Python, библиотека распознавания, веса модели (20 МБ) и FFmpeg (около 0,7 ГБ). Остальное — по видеокарте:

	Без видеокарты NVIDIA	NVIDIA GTX	NVIDIA RTX
Как считает	DirectML, на любой видеокарте с DirectX 12	PyTorch с CUDA	TensorRT поверх PyTorch с CUDA
Что ставится под неё	PyTorch для процессора и DirectML	PyTorch с CUDA, около 3 ГБ скачивания	PyTorch с CUDA и TensorRT, по 3 ГБ скачивания
Место на диске	около 5 ГБ	не измерялось	около 15 ГБ на время установки, 11,5 ГБ после неё

	Без видеокарты NVIDIA	NVIDIA GTX	NVIDIA RTX
Получасовая запись 640×480	около полутора часов	24 минуты (см. ниже)	5 минут
Очередь	записи по одной	—	четыре записи разом: двести получасовых записей — около 12 часов
Обучение модели типов автомобилей, 25 эпох на 3500 вырезках	полчаса	не измерялось	5,6 минуты

Времена — автора. Столбец без NVIDIA — ноутбук со встроенной графикой Intel (счёт) и его процессор (обучение); столбец RTX — RTX 5070. На видеокарте GTX программа не запускалась: 24 минуты в её столбце — тот же путь счёта, PyTorch с CUDA без TensorRT, измеренный на RTX 5070; карта GTX слабее и считает дольше. Видеокарта NVIDIA старше GTX 10-й серии считается как её отсутствие: эта сборка PyTorch на ней не считает, счёт идёт через DirectML.

Откуда размер. На машине с RTX из 11,5 ГБ: PyTorch с CUDA — 7 ГБ (в нём библиотеки NVIDIA для вычислений на видеокарте), TensorRT — 3,3 ГБ, FFmpeg — 0,7 ГБ, остальное — Python, библиотека распознавания и модели. Без видеокарты NVIDIA вместо первых двух ставится PyTorch для процессора, около 3 ГБ. Скачанные установочные файлы — ещё около 3,5 ГБ на машине с RTX — после установки удаляются.

Для чего. TensorRT собирает модель в один движок под вашу видеокарту — при первом счёте каждой модели около минуты — и считает в 4,7 раза быстрее, чем PyTorch с CUDA на той же карте: 140 кадров в секунду против 30. Числа счёта от пути почти не зависят: между TensorRT, CUDA и DirectML они расходятся на единицы автомобилей.

С Claude и Codex

С программой можно работать вместе с ИИ-агентом — Claude Code или Codex. Руководство для агента — `Docs\AI_AGENT_GUIDE_RU.md`. Откройте папку программы в агенте, попросите прочитать его — и он расставит линии по кадрам записей, посчитает запись, разложит спорные машины на листы и обучит модель классов под ваши ракурсы: нарежет вырезки, разметит их, обучит и проверит на машинах, которых модель не видела. Как это устроено и что агент при этом видит — в руководстве.

Чего программа не делает

Не собирает и не отправляет данные; в сеть выходит только за установкой и весами модели. Не ставит ничего в систему: всё лежит в её папке. Не читает номера машин. Не делит грузовые по числу осей: классы — легковой, автобус, грузовой, мотоцикл.

Точность зависит от размера машины в кадре: вблизи камеры счёт сходится с ручным в пределах одного-двух процентов, вдали часть машин теряется, совсем мелкие не видны вовсе. Легковые отделяются от остальных точно, автобусы и грузовые — приблизительно.

Из чего сделано

Окно — C# на Avalonia 12 (.NET 10). Движок счёта — Python: Ultralytics YOLO и трекер ByteTrack, PyTorch, TensorRT, ONNX Runtime с DirectML, OpenCV; видео читает FFmpeg. Значки — Lucide, шрифты — JetBrains Mono и Inter. Сторонние программы и библиотеки скачиваются по вашей кнопке и остаются под своими лицензиями.

На снимке окна — кадр свободного видео, а не запись заказчика: «Motorway A40: on bridge above the traffic», автор Kathinka Engels (Sounds of Changes), Wikimedia Commons, лицензия CC BY 3.0.