

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**ОПРЕДЕЛЕНИЕ СКОРОСТИ ТРАНСПОРТНОГО СРЕДСТВА  
БЕЗРАДАРНЫМ МЕТОДОМ**

Автор: Смирнова Валентина Сергеевна

Направление подготовки: 01.04.02 Прикладная  
математика и информатика

Квалификация: Магистр

Руководитель ВКР: Фильченков А.А., к.ф.-м.н

Санкт-Петербург, 2022 г.

Обучающийся Смирнова Валентина Сергеевна  
Группа М42381с Факультет ИТиП

Направленность (профиль), специализация  
Технологии разработки программного обеспечения

Консультанты:

- а) Белашенков Н.Р., канд. физ.-мат. наук, без звания \_\_\_\_\_  
б) Беззубик В.В., без степени, без звания \_\_\_\_\_

ВКР принята «\_\_\_\_\_» 20\_\_\_\_ г.

Оригинальность ВКР \_\_\_\_%

ВКР выполнена с оценкой \_\_\_\_\_

Дата защиты «6» июня 2022 г.

Секретарь ГЭК Павлова О.Н. \_\_\_\_\_

Листов хранения \_\_\_\_\_

Демонстрационных материалов/Чертежей хранения \_\_\_\_\_

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**УТВЕРЖДАЮ**

Руководитель ОП  
проф., д.т.н. Парфенов В.Г. \_\_\_\_\_  
«\_\_\_\_\_» 20\_\_\_\_ г.

**ЗАДАНИЕ  
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ**

**Обучающийся Смирнова Валентина Сергеевна**

**Группа М42381с Факультет ИТиП**

**Квалификация:** Магистр

**Направление подготовки:** 01.04.02 Прикладная математика и информатика

**Направленность (профиль) образовательной программы:** Технологии разработки программного обеспечения

**Тема ВКР:** Определение скорости транспортного средства безрадарным методом

**Руководитель** Фильченков А.А., к.ф.-м.н, доцент факультета информационных технологий и программирования, Университет ИТМО

**2 Срок сдачи студентом законченной работы до:** «27» мая 2022 г.

**3 Техническое задание и исходные данные к работе**

Требуется разработать и внедрить алгоритм определения скорости движущегося по дорогам общего пользования транспортного средства безрадарным методом, то есть по видео с камеры фиксации нарушений правил дорожного движения. Алгоритм должен определять скорость с точностью до 2 км/ч или до 3% при скоростях выше 100 км/ч.

**4 Содержание выпускной квалификационной работы (перечень подлежащих разработке вопросов)**

Выпускная квалификационная работа должна показывать результат, соответствующий заявленной точности определения скорости транспортного средства.

**5 Перечень графического материала (с указанием обязательного материала)**

Графические материалы и чертежи работой не предусмотрены

**6 Исходные материалы и пособия**

- а) ГОСТ Р 50577-2018 «Знаки Государственные Регистрационные Транспортных Средств»;
- б) Jozef Gerát. Vehicle Speed Detection from Camera Stream Using Image Processing Methods;
- в) Joko Siswantoro. Real World Coordinate from Image Coordinate Using Single Calibrated Camera Based on Analytic Geometry

**7 Дата выдачи задания «01» сентября 2020 г.**

Руководитель ВКР \_\_\_\_\_

Задание принял к исполнению \_\_\_\_\_ «01» сентября 2020 г.

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**АННОТАЦИЯ**  
**ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**

**Обучающийся:** Смирнова Валентина Сергеевна

**Наименование темы ВКР:** Определение скорости транспортного средства безрадарным методом

**Наименование организации, в которой выполнена ВКР:** Университет ИТМО

**ХАРАКТЕРИСТИКА ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**

1 Цель исследования: Разработать и внедрить алгоритм определения скорости транспортного средства по видеопотоку с камеры фиксации нарушений правил дорожного движения.

2 Задачи, решаемые в ВКР:

- а) реализация rc-парсера (выходного формата прибора);
- б) генерация синтетического датасета;
- в) разработка алгоритма определения скорости;
- г) внедрение алгоритма в тестовый прибор;

3 Число источников, использованных при составлении обзора: 0

4 Полное число источников, использованных в работе: 0

5 В том числе источников по годам:

Отечественных			Иностранных		
Последние 5 лет	От 5 до 10 лет	Более 10 лет	Последние 5 лет	От 5 до 10 лет	Более 10 лет
0	0	0	0	0	0

6 Использование информационных ресурсов Internet: нет

7 Использование современных пакетов компьютерных программ и технологий:

Пакеты компьютерных программ и технологий	Раздел работы
<ul style="list-style-type: none"><li>а) Пакеты numpy, pandas и json языка Python3 для разработки.</li><li>б) Программа Blender для работы с 3D сценами.</li><li>в) Пакет bpy языка Python3 для генерации датасета через Blender.</li><li>г) Программа RecordsView для работы с rc-файлами.</li></ul>	

8 Краткая характеристика полученных результатов

Разработан и внедрён действенный алгоритм определения скорости транспортного средства безрадарным методом.

9 Гранты, полученные при выполнении работы

По теме данной работы гранты отсутствуют.

10 Наличие публикаций и выступлений на конференциях по теме выпускной работы

По теме данной работы публикации отсутствуют.

Обучающийся Смирнова В.С. \_\_\_\_\_

Руководитель ВКР Фильченков А.А. \_\_\_\_\_

«\_\_\_\_» \_\_\_\_\_ 20\_\_ г.

**СОДЕРЖАНИЕ**

## ВВЕДЕНИЕ

На сегодняшний день фиксация нарушений правил дорожного движения почти полностью автоматизирована и осуществляется посредством специальных приборов, в их числе приборы производства компании «Симикон»:

- Комплекс «КОРДОН-КРОСС»
- Комплекс «КОРДОН.ПРО»М
- Комплекс «КОРДОН.ПРО»В
- Комплекс «ПАРКОН-А»
- Комплекс «КОРДОН-М»
- Система контроля «ЗПИ»
- Приспособление «МИРАЖ»
- Комплекс «КОРДОН-ТЕМП»
- Комплекс «КОРДОН-М”КР»
- Видеорегистратор «ГРОМ-1»
- Имитатор «ИС-24/3»
- Имитатор «ИС-24»Д



Рисунок 1 – Комплекс «КОРДОН.Про»М в стационарном режиме — на столбе.

Для фиксации нарушений скоростного режима приборы в обязательном порядке оборудованы радаром и камерой. Радар служит для измерения скорости транспортного средства с точностью до 1 км/ч, а камера — для распознавания Государственного регистрационного знака (далее — номер или номерной знак) и предоставления доказательства нарушения. Полученная с камеры и радара информация обрабатывается на приборе и формируется коллаж с информацией о проезде. После чего прибор, в соответствии с ограничениями на участке, принимает решение о том, был ли данный проезд нарушением или нет и отправляет нарушения в соответствующие органы.



Рисунок 2 – Комплекс «КОРДОН.Про»М в передвижном режиме — на триноге.

Основные приборы, используемые для фиксации скоростных нарушений — комплексы семейства «КОРДОН», далее в работе под «приборами» будем подразумевать именно их. КОРДОНЫ оборудованы радаром, камерой, ИК- прожектором и непосредственно железом, на котором производятся вычисления и хранятся нарушения. Данные приборы могут работать в 3 режимах: стационарный (Рисунок ??), передвижной(Рисунок ??) и мобильный(Рисунок ??). В работе мы будем иметь в виду стационарный режим. Далее можно будет масштабировать предложенное решение на оставшиеся режимы. Технические характеристики прибора представлены в таблице ??.



Рисунок 3 – Комплекс «КОРДОН.Про»М в мобильном режиме — на заднем сидении.

Так как использование радара в приборах — достаточно затратно, а наличие камеры в приборе обязательно для распознавания номера, формирования коллажа и предоставления доказательства нарушения, то закономерно возникает задача определения скорости транспортного средства безрадарным методом исключительно по данным с камеры прибора, другими словами — распознавание скорости транспортного средства по видео.

Таблица 1 – Технические характеристики комплекса «КОРДОН.Про»М.

ПАРАМЕТР	ЗНАЧЕНИЕ
Диапазон измеряемых скоростей	2 - 300 км/ч
Пределы допускаемой абсолютной погрешности измерений скорости	± 1,0 км/ч
Пределы допускаемой абсолютной погрешности синхронизации внутренней шкалы времени с UTC(SU)	± 5 мкс
Потребляемая мощность (при положительных температурах)	не более 25 Вт
Масса датчика	не более 6,0 кг
Габаритные размеры	не более 460×180×280 мм

## ГЛАВА 1. ОБЗОР ВОЗМОЖНОСТЕЙ КОМПЛЕКСА «КОРДОН.ПРО»М

В данной главе мы введём ключевые понятия, подробно рассмотрим технологии, по которым на сегодняшний день работает прибор семейства «КОРДОН» и рассмотрим существующие решения для определения скорости объекта по видео.

### 1.1. Определения и ключевые понятия

Для начала введем определения, ключевые понятия и некоторые соглашения, которые будут использоваться в работе:

- **транспортное средство (ТС)** — любой участник дорожного движения, обладающий Государственным регистрационным знаком (автомобиль, мотоцикл, автобус, грузовик)
- **Государственный регистрационный знак (ГРЗ)** — номерная пластина на транспортном средстве, соответствующая ГОСТу Р 50577-2018 «Знаки Государственные Регистрационные Транспортных Средств»
- **правила дорожного движения (ПДД)** — свод правил, регулирующих обязанности участников дорожного движения (водителей транспортных средств, пассажиров, пешеходов и так далее), а также технические требования, предъявляемые к транспортным средствам, для обеспечения безопасности дорожного движения, актуальные на момент написания работы
- **нарушение** — любое нарушение правил дорожного движения, однако в данной работе под «нарушением» будем понимать нарушение скоростного режима на рассматриваемом участке

Также необходимо определить, что выход прибора — не просто видеофайл, а специальный формат .rc, который содержит в себе всю информацию с прибора, как, например, битовые представления кадров, радарные данные, проекционная матрица камеры, время, информация о распознанных номерах, данные о GPS, карты грубины, данные о настройках камеры. В данной работе мы будем обрабатывать именно этот формат, поэтому, рассмотрим его подробнее.

### 1.2. Содержание RC-файла

RC-файл представляет из себя битовый файл, состоящий из frame-ов (рисунок ??). Каждый фрейм описывает 1 кадр из записи. В начале каждого фрейма

ма находятся 2 поля: `magic` и `length`, за ними идёт непосредственно информация о фрейме — `frame data`. Поле `magic` служит исключительно для проверки корректности чтения, оно имеет константное значение `0x11223307` для всех фреймов. Поле `length` определяет длину `frame data`(включая `magic` и `length`), то есть сколько последующих бит будет относиться к текущему фрейму.

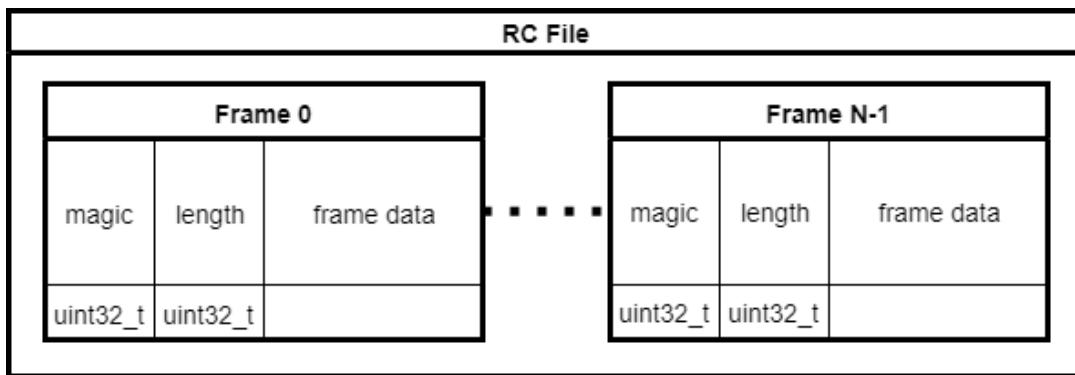


Рисунок 4 – Представление RC-файла.

Каждый фрейм содержит в себе набор данных соответствующих типов, схема представлена на рисунке ???. В начале каждого фрагмента данных присутствует 2 поля: `id` и `length`. Поле `id` определяет тип читаемых данных во фрагменте, это может быть примитивный тип, как, например, `timestamp` или целые структуры, как, например, `sCameraData`. Подробное описание каждого типа представлено в приложении ???. Поле `length` аналогично одноимённому полю в RC описывает длину последующей последовательности.

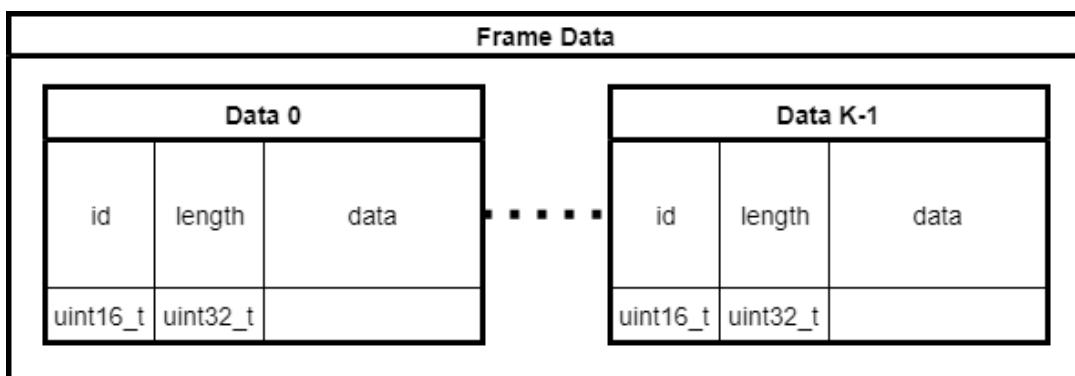


Рисунок 5 – Представление frame-a.

В таблицах ?? и ?? приложения представлено подробное описание к рисункам ?? и ?? соответственно. Теперь подробнее остановимся на 2 типах данных из таблицы ?? — информации о распознанных номерах и радарных целях.

**Результаты распознавания номера** Результаты распознавания номера соответствуют ID=5 из таблицы ?? и представляют собой массив типа TRectNumber. Реализация структуры описана в листинге ??.

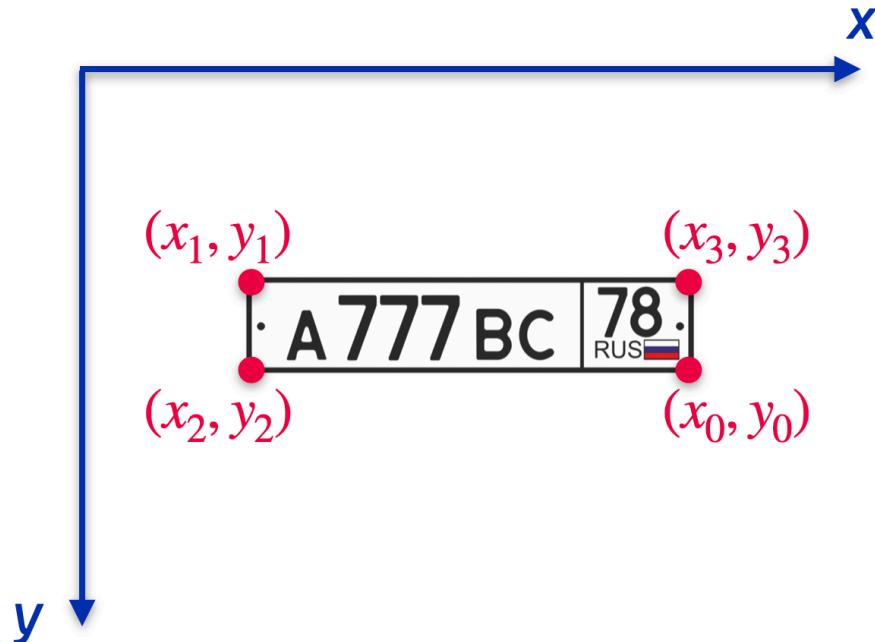


Рисунок 6 – Координаты номера в структуре TRectNumber.

Листинг 1 – Реализация структуры TRectNumber.

```
#define MAX_SYMBOL_NUM (16)

typedef int16_t TInt3[4];

typedef struct TRectNumber
{
    int numFormat;
    int n_symbols;
    uint16_t text16[MAX_SYMBOL_NUM];

    int16_t allCert;
    int16_t certList[MAX_SYMBOL_NUM];
    TInt3 x, y;

} TRectNumber;
```

Здесь numFormat — тип номерной рамки, складывается из страны и формата (однострочный/двустрочный), n\_symbols — количество распознан-

ных символов, `text16` — непосредственно распознанный текст в кодировке `UINT-16`, `allCert` — общая вероятность распознавания, `certList` — массив вероятностей распознавания каждого символа, `x` и `y` — массивы координат распознанного номера, порядок отображён на рисунке ??.

**Цели с радара** Результаты распознавания номера соответствуют `ID=8` из таблицы ?? и представляют собой массив типа `umrr_target`. Реализация структуры описана в листинге ??.

Листинг 2 – Реализация структуры `umrr_target`.

```
class umrr_target
{
public:
    std::string doString() const;
    double absSpeed() const;

    int id;
    double x;
    double y;
    double xspeed;
    double yspeed;
    double len;

    // calc values
    double imgx;
    double imgy;
    int numw;

    double imgx_left;
    double imgx_right;

    double imgy_top;
    double imgy_bottom;
} __attribute__((__packed__));
```

Здесь `id` — идентификационный номер радарной цели от 0 до 63 (количество одновременно отслеживаемых целей), `x` — координата `x` в плоскости дороги, `y` — координата `y` в плоскости дороги, `xspeed` — мгновенная скорость по оси `x`, `yspeed` — мгновенная скорость по оси `y`, `len` — длина транспортного средства. Система радарных координат располагается в основании столба, на котором установлен прибор, и считается по правилу правой руки, визуализация представлена на рисунке ??.

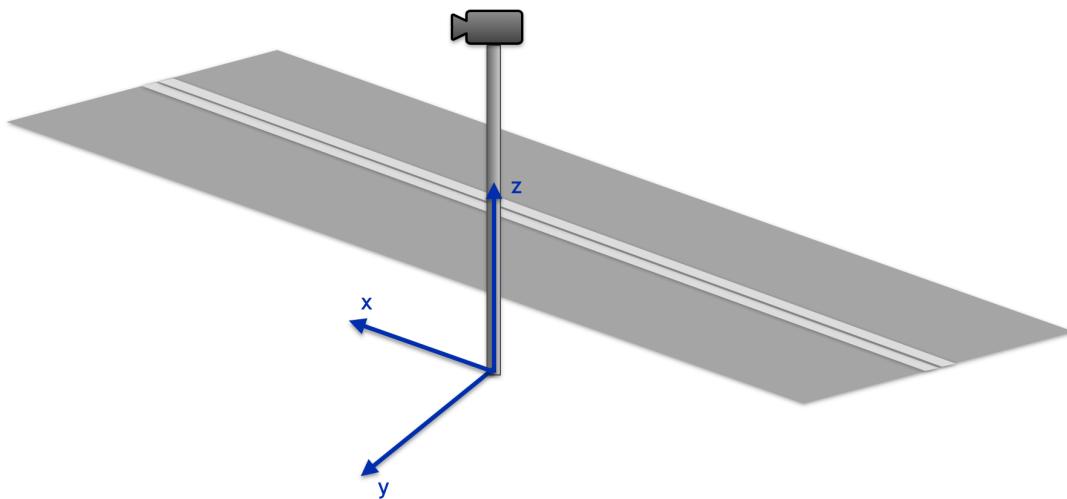


Рисунок 7 – Расположение системы радарных координат.

### 1.3. Возможности прибора

Для того, чтобы начать разработку нового решения, необходимо разобраться с тем, как устроен работающий прибор. Во введении мы определили основные характеристики и возможности прибора. Теперь чуть более подробно рассмотрим основные функции и алгоритм его работы, чтобы определить, какую информацию можно использовать для разработки. Ниже представлен перечень всех функций и возможностей киниверсального комплекса «КОР-ДОН.Про»М:

#### **Автоматическая фотовидеофиксация**

- Автоматическая фотовидеофиксация нарушений ПДД в зоне контроля до шести полос движения одновременно в обоих направлениях.
- Измерение скоростей в диапазоне от 2 до 300 км/ч.
- Возможность измерения средней скорости на протяженных участках совместно с любыми другими комплексами семейства «Кордон».
- Возможность фиксации нарушения «Непредоставление преимущества пешеходу» при наличии дополнительного блока «Мера».
- Возможность дооснащения обзорной камерой для работы со знаками переменной информации.
- Отдельные пороги скорости для разных полос движения и для ТС категорий «В», «С» и «Д».
- Автоматическое сохранение фотоматериалов и видеоролика по каждому зафиксированному нарушению.

- Модуль ГЛОНАСС/GPS с автоматической коррекцией системного времени комплекса.

### **Распознавание номерных знаков и розыск ТС**

- Автоматическое распознавание номерных знаков многих стран мира, включая двусторонние номера и российские ГРЗ нового образца по ГОСТ Р 50577-2018.
- Возможность включения и выключения распознавания ГРЗ тех или иных государств.
- Проверка распознанных номеров по базам данных с оповещением оператора и голосовым озвучиванием номера.
- Специальный режим работы «Перехват» для розыска ТС с отсутствующими или нечитаемыми номерными знаками.

### **Классификация ТС**

- Автоматическое определение типа ТС и их классификация по четырём основным категориям (легковые, грузовые, автобусы, среднегабаритные).
- Автоматическое присвоение соответствующей категории ТС порога скорости по ПДД.
- Автоматический контроль запрета движения ТС для заданной категории (грузовые, автобусы и т.д.) по отдельным полосам или по дороге в целом.

### **Видеонаблюдение**

- Трансляция видеопотока с высоким разрешением по протоколу RTSP.
- Ведение непрерывной видеозаписи с возможностью скачивания видеоролика по заданному промежутку времени.

### **Передача данных**

- Передача данных на сервер ЦОД по зашифрованным проводным или беспроводным каналам связи (3G/4G).
- Автоматическое переключение на резервные каналы связи (Wi-Fi, 4G) при сбоях или отказе основного канала.
- Возможность параллельной передачи данных с комплекса на различные серверы.

## **Защита и безопасность**

- Защита данных и встроенного ПО от несанкционированных изменений.
- Экспортируемые данные защищены ЭЦП.
- Ведение журнала событий и действий пользователя комплекса.
- Передача уведомлений по SMS и электронной почте о зафиксированных фактах физических воздействий на прибор (удары, вибрация).
- Возможность защиты от огнестрельного оружия с использованием бронированного кожуха, сертифицированного на пулестойкость по классам «Бр2» (пистолеты СПС и ТТ) и «С1» (охотниче ружье со свинцовой пулей).

## **Телеметрия и диагностика**

- Самодиагностика, удаленная диагностика.
- Автоматическое отслеживание параметров комплекса и передача телеметрической информации в режиме реального времени.

## **Установка**

- Автоматическая проверка правильности монтажа комплекса.
- Поворотный кронштейн для быстрой стационарной установки на опоре.
- Удобный и простой веб-интерфейс для настройки.
- Различные варианты подключения к сетям электропитания. Возможность подключения к осветительной сети и обеспечения бесперебойной работы комплекса от АКБ.

## **Работа в ночное время**

- Встроенная инфракрасная подсветка для работы в ночное время.
- Дополнительный внешний ИК- прожектор для гарантированного определения марки ТС по изображению.

## **Статистика**

- Сбор статистических данных об интенсивности транспортного потока.
- Построение интерактивных графиков по выбранным статистическим параметрам.
- Анализ зафиксированных нарушений ПДД с разбивкой по видам нарушений и величине превышения скорости.

#### **1.4. Входные данные для исследования**

Из секции ?? и описания RC-файла в секции ?? можно вынести следующую полезную информацию, которая будет использоваться в дальнейшем в работе:

- а) из информации об установке прибора в стационарном режиме:
  - 1) высоту столба —  $H$
  - 2) расстояние до дороги —  $R$
  - 3) угол между направлением движения и направлении камеры —  $\alpha$
- б) из информации о самом приборе:
  - 1) физические размеры матрицы — высота  $h\_m$  и ширина  $w\_m$  в миллиметрах
  - 2) разрешение — высота  $h\_m$  и ширина  $w\_m$  в пикселях
  - 3) фокусное расстояние —  $f$
- в) из RC-файла:
  - 1) проекционная матрица камеры  $3 \times 4$  — `matrix`
  - 2) данные по распознанным номерам — `licnum`
  - 3) радарные данные — `radar_targets`

#### **Выводы по главе 1**

В первой главе мы подробно разобрали устройство и возможности прибора на примере комплекса «КОРДОН.Про»М. Стоит отметить, что примерно все прибора семейства «КОРДОН» обладают схожими характеристиками и возможностями. Например, любой прибор можно дооборудовать цветной камерой, дополнительным прожектором, источником питания или бронированным кожухом. Принципиальное отличие «КОРДОН.Про»М и «КОРДОН.Про»В в наличии радара, в «В» версии он отсутствует, что удешевляет производство, однако использовать такой прибор для фиксации скоростных нарушений пока невозможно. Данный факт непосредственно подтверждает актуальность внедрения решения данной работы.

Кроме того, в данной главе мы определили необходимые для вычислений переменные и структуры, которые будут использованы в последующих главах. Также, был подробно рассмотрен основной формат для данных для исследования — RC-файл, его содержание, подробное описание каждого типа данных и содержащихся в них структур. Стоит отметить, что данный формат будет использоваться как в качестве тренировочных данных (имеются в виду

записи с различных приборов в разных городах и при разных условиях в формате .rc), так и для внедрения решения непосредственно в реальный прибор.

## ГЛАВА 2. ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ

На сегодняшний день существует несколько подходов к решению задачи определения скорости по видео. В основном, используются технологии компьютерного зрения и линейная алгебра. В данной главе мы рассмотрим наиболее перспективные подходы, их преимущества и недостатки и наметим план решения.

### 2.1. Базовая геометрическая оптика

Один из самых простых подходов [5228496] основывается на базовой геометрической оптике. Вспомним некоторые базовые определения:

- а) *Фокус собирающей линзы* — это место, где все проходящие через линзу лучи света пересекаются.
- б) *Фокусное расстояние собирающей линзы*  $f$  — это отрезок от принятого центра линзы до её фокуса.
- в) *Объектом* называется физическое тело, участвующее в оптической системе. Обычно объект располагается перед линзой.
- г) *Изображением* называется проекция объекта на экран или матрицу (располагается после линзы).
- д) *Расстоянием до объекта*  $d$  называется отрезок между объектом и линзой.
- е) *Расстоянием до изображения* называется отрезок между изображением и линзой.

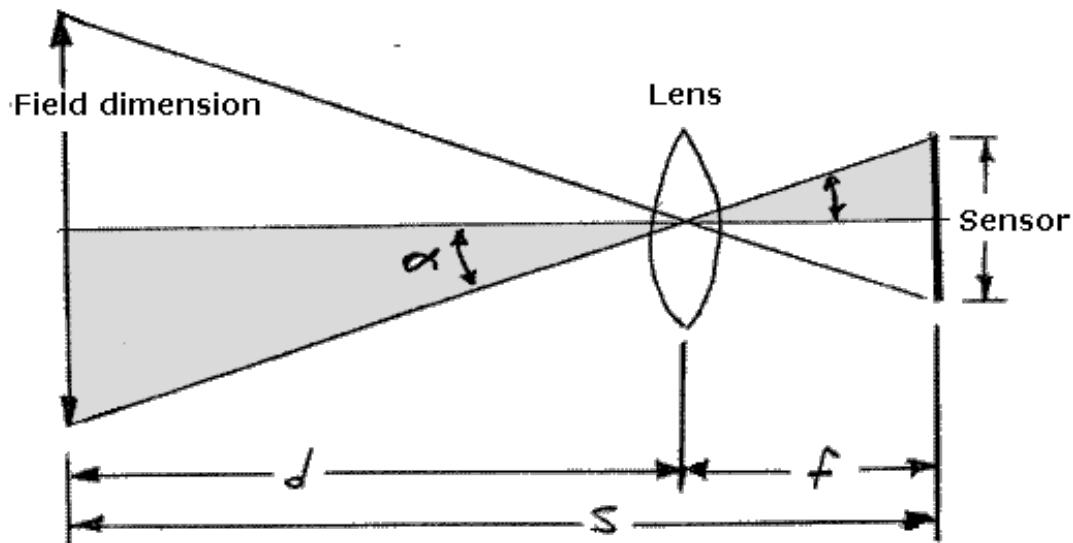


Рисунок 8 – Схема собирающей линзы.

Из законов базовой оптики известно, что зная физические размеры объекта и экрана (в нашем случае в качестве экрана выступает матрица камеры), также, зная фокусное расстояние линзы, можно найти расстояние до объекта. На рисунке ?? представлена схема собирающей линзы, а формулой ?? описывается выше упомянутое отношение.

$$v = \frac{dr}{dt} \quad (1)$$

В статье [5228496] описано решение, основанное именно на этом подходе и погрешность составляет 4%. Если переводить это на километры, то погрешность составит от 1 до 12 км/ч при заявленном диапазоне скоростей от 0 до 300 км/ч, что больше допустимой погрешности, поэтому данное решение недостаточно точное для данного исследования.

## 2.2. Компьютерное зрение

Остальные перспективные решения [Gawande20] основываются на работе с изображениями, а именно разбиением транспортного средства на составляющие и привязкой к одной из них.

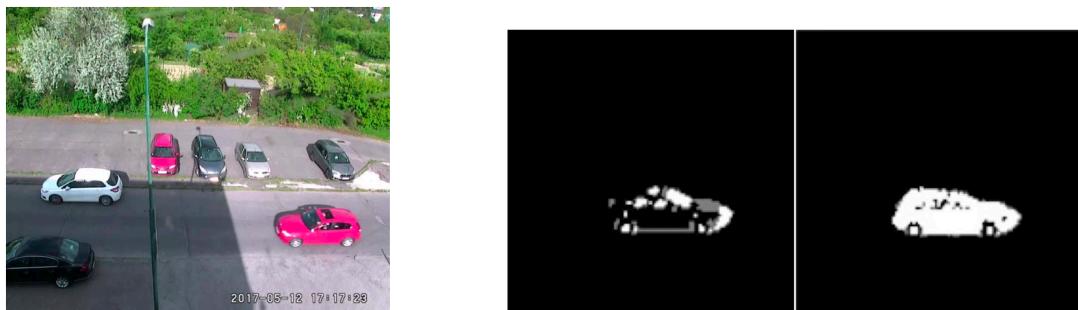


Рисунок 9 – Вырезка из источника [8124468], детект сбоку.

Например, в источнике [8124468] скорость автомобиля определяется сбоку (рисунок), то есть применены допущения, что камера установлена перпендикулярно дороге, что в реальном мире невозможно, поэтому данное решение также нам не подходит.

В другом источнике [6754885] машина разбивается на точки кривых и детект [surveillance] ведётся по крайней из этих точек. Очевидно, что из-за сложной формы автомобиля, цвета и бликов, крайние точки могут меняться, что вызовет большую погрешность. Не смотря на подходящий для нашей работы ракурс и заявленную точность определения скорости до 2 км/ч, на это



Рисунок 10 – Вырезка из источника [6754885], детект спереди под углом.

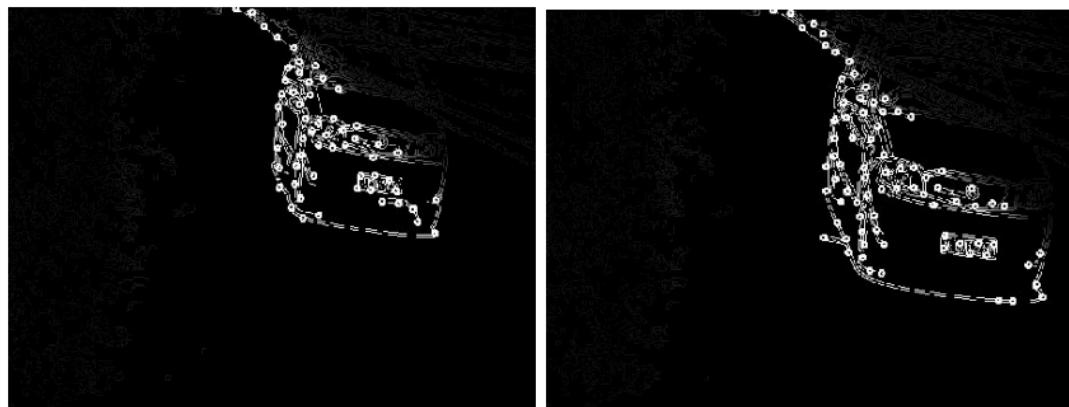


Рисунок 11 – Вырезка из источника [6754885], разбиение автомобиля на точки.

число нельзя ориентироваться, так как исследование [6754885] проводилось всего на нескольких видео и данный недостаточно для сбора подобной статистики. Также в этой статье были приняты некоторые допущения при вычислениях, как например замена поправочного коэффициента камеры на константу, зависящую от координат пикселя, что также недопустимо.

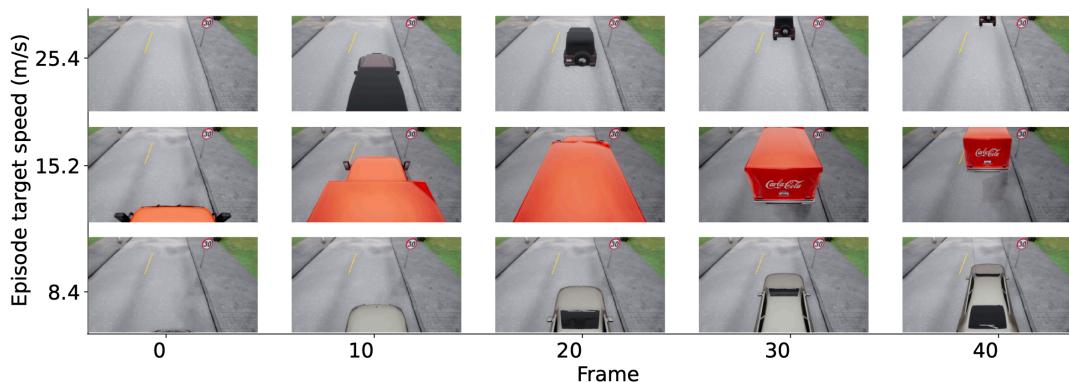


Рисунок 12 – Вырезка из источника [09903], гифмер синтетических данных.

Ещё один популярный подход [09903] основан на расположении камеры непосредственно над дорогой. Из данного положения, принятые некоторые допущения, исходя из которых детекция транспортного средства может проводиться исключительно по прямой траектории по одной полосе движения. Во-первых, расположение камеры над дорогой не является универсальным решением, так как далеко не на всех участках дороги это возможно. Во-вторых, мы хотим детектировать до 30 целей одновременно, а на одной полосе это невозможно. Однако так как данное решение основано на синтетических данных (рисунок ), то есть видеопоток является вручную генерированной 3D сценой, возьмём эту технику на заметку.

## Выводы по главе 2

В данной главе мы рассмотрели существующие подходы к решения задачи нахождения скорости по видео. Максимально достигнутая точность определения скорости — 2 км/ч, однако нельзя полагаться на данный результат, так как в статье [6754885] присутствуют некоторые неточности и не произведено достаточно большое количество экспериментов для получения статистически значимого результата.

Наиболее правдоподобный подход обеспечивает погрешность до 7 км/ч, однако он основывается на расположении камеры непосредственно над движущимся транспортным средством, что не позволит измерять скорость на много-полосной дороге и при перестройках, а только на одной полосе.

## ГЛАВА 3. ПРЕДЛОЖЕННОЕ РЕШЕНИЕ

В данной главе мы разобьём задачу определения скорости по видео на более мелкие подзадачи и приведём решение каждой из них.

### 3.1. Сбор входных данных. RC-парсер

В главе ?? мы вывели некоторый список входных данных, связанных с характеристиками прибора, которые необходимо определить отдельно для каждого RC-файла, например, фокусное расстояние, размеры матрицы камеры, высота установки прибора. Для этого соберём некоторый набор реальных данных, записанных на приборах при различных условиях. Для этого воспользуемся базой заранее записанных файлов, рисунок ??.

Название записи	Тип прибора	Матрица	Фокус	Положение	Расстояние от дороги	Высота установки	Ширина полосы	Кол-во полос	Ширина разделительной полосы	Угол поворота радара	Автобусы	Грузовики	Мотоциклы	Спецтехника	Дополнител
1527685066_2018.05.30_15-57-46.rc <a href="#">Скачать RC</a>	Стационарный	Imx249	50	Дорога справа	1,4	7,7	3,7	4	0	-13	1	0	0		
1531482918_2018.07.13_14-55-18.rc <a href="#">Скачать RC</a>	Стационарный	Imx249	50	Дорога справа	0,5	7,9	3,7	4	0	-8	0	1	0		синхронная запись в период 1531482920_2018.07.13_1

Рисунок 13 – WEB-интерфейс базы с RC-файлами.

Так как работать с RC-файлами достаточно неудобно, возникла задача извлечения релевантных для исследования данных и преобразования их в удобный формат. Для этого, на языке Python был реализован парсер RC-формата в привычный JSON (пример представлен на листинге ??). Подробнее остановимся на его содержании.

Поля `seconds` и `microseconds` описывают время в тиках прибора с начала UNIXTime. Поле `licnums` представляет собой массив с информацией о распознанных номерах. Поле `radar_targets` описывает радарные цели. Важно отметить, что в текущем состоянии не известно, какая радарная

Листинг 3 – Пример JSON-файла.

```
{
    "frames": [
        {
            "seconds": 1596722143,
            "microseconds": 379221,
            "licnums": [
                {
                    "format": 42139649,
                    "text": "МТВ684186",
                    "x": [450,399,400,449],
                    "y": [39,30,42,27]
                }
            ],
            "radar_targets": [
                {
                    "id": 40,
                    "x": 60.608,
                    "y": 4.5439,
                    "xspeed": -21.4,
                    "yspeed": -0.3,
                    "carlen": 4.0
                },
            ]
        },
        {
            "matrix_type": 0.0055,
            "focal_length": 35.0,
            "matrix3x4": [ 865.751108, -6388.572118, -544.713764,
                -301.569121, -841.281125, 344.588298, -6313.435574,
                54640.980513, 0.978972, -0.032996, -0.20131, -1.268419 ]
        }
    ]
}
```

цель какому распознанному номеру соответствует, эта проблема будет подобна разобрана в секции **??**. Поля `matrix_type` и `focal_length` описывают тип матрицы камеры и её фокусное расстояние соответственно. Из типа матрицы можно получить её физические размеры: ширину, высоту в пикселях и миллиметрах и разрешение. И, наконец, поле `matrix3x4` хранит в себе проекционную матрицу камеры  $3 \times 4$ , она понадобится нам в дальнейшем для восстановления координат.

### 3.2. Объект мониторинга

В главе **??** мы определили, что разбиение автомобиля на части в конечном итоге является неточным способом для определения скорости, так как могут появляться блики, части автомобиля могут быть закрыты, деформированы, сливаться с окружающей средой. Поэтому возникает вопрос: к какой точке

автомобиля необходимо привязываться, чтобы следить за её перемещением с максимальной точностью. В качестве такой точки был выбран номерной знак. Основные преимущества данного выбора заключаются в следующем:

- для любого кадра, где распознан номер, известны координаты 4 углов этого номера с точностью до нескольких сантиметров, а в случаях, когда номер не распознан, формирование нарушения невозможно
- для каждого номера известен его точный физический размер согласно ГОСТу (рисунок ??)



Рисунок А.1 — Регистрационный знак типа 1 с двухзначным кодом региона регистрации

Рисунок 14 – Габариты Государственного регистрационного знака согласно актуальному на момент написания работы ГОСТу.

### 3.2.0.1. Сбор датасета Государственных регистрационных знаков

Так как в работе мы будем использовать габаритные размеры номерной пластины, а их существует большое множество (рисунок ??), был собран датасет, который для каждого типа номерной пластины описывает её габаритные размеры согласно ГОСТу.

## 3.3. Алгоритм определения скорости

Для начала дадим определение скорости. *Скорость* — векторная физическая величина, характеризующая быстроту перемещения и направление движения материальной точки относительно выбранной системы отсчёта. По определению, равна производной радиус-вектора точки по времени. В СИ измеряется в метрах в секунду.

$$\vec{v} = \frac{d\vec{r}}{dt} \quad (2)$$

Так как точное время для каждого времена известно, остаётся найти перемещение объекта, чью скорость мы собираемся измерить и так как в секции

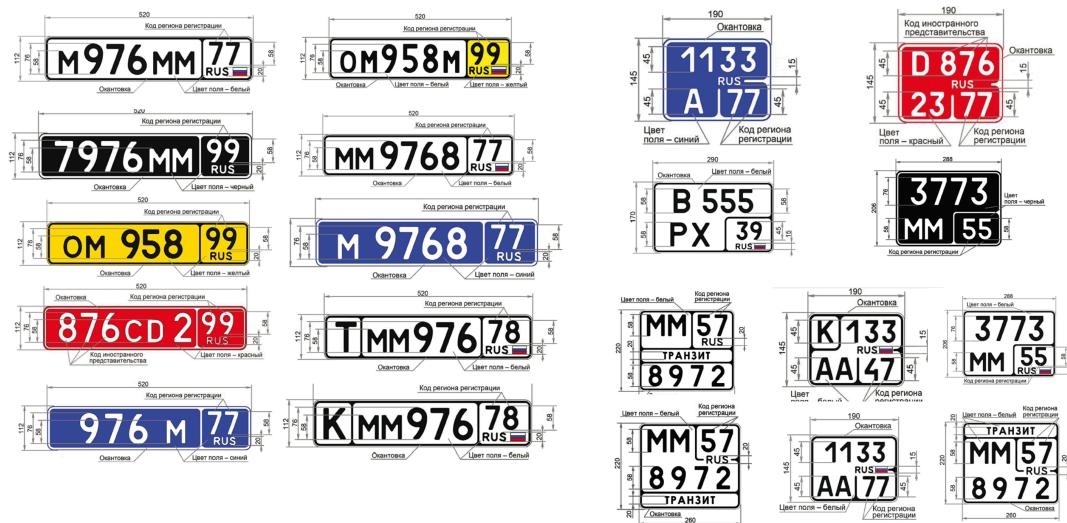


Рисунок 15 – Типы Государственных регистрационных знаков.

?? было принято наблюдать за номером, задача сводится к нахождению неремещения номерного знака между кадрами. Но между какими кадрами?

Согласно законодательству, для формирования штрафа, необходимо предоставить доказательную базу в виде коллажа с изображением транспортного средства и номерного знака с измеренной на момент кадра *моментальной* скоростью. Данное ограничение приводит нас к выводу, что перемещение необходимо измерять между 2 ближайшими фреймами. Теперь перейдём непосредственно к решению.

Рассмотрим положение номера в двух моментах времени —  $t_1$  и  $t_2$ . Схема представлена на рисунке ??.

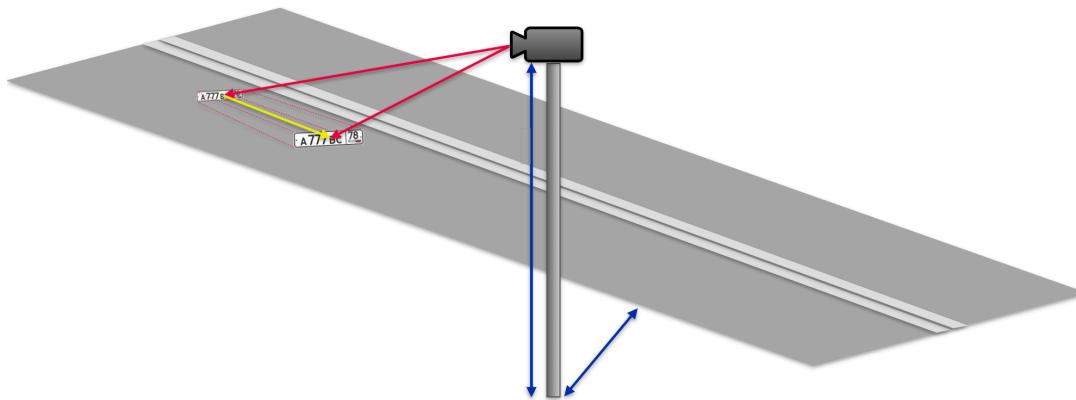


Рисунок 16 – Габариты Государственного регистрационного знака согласно актуальному на момент написания работы ГОСТу.

Из оптических отношений, зная физические размеры матрицы камеры, фактический размер объекта и фокусное расстояние, можно определить расстояние до объекта, то есть длины красных векторов на рисунке ??.

Из векторной геометрии очевидно, что чтобы найти перемещение  $r$  достаточно найти разность векторов  $a$  и  $b$ , однако для нахождения разности векторов, необходимо дополнительно знать либо угол между ними, либо координаты обоих векторов, а ни того, ни другого у нас нет, поэтому возникает промежуточная задача нахождения координат этих векторов в пространстве.

Так как мы рассчитали расстояние до центра номера, чтобы восстановить координаты проведённого вектора достаточно будет найти координаты комерной платины в пространстве. Для этого решим задачу триангуляции.

**Приведени задачи триангуляции** Классическая задача триангуляции [845377] восстанавливает 3D координаты [*inproceedings*] объекта по изображения с 2 камер. В общем случае уравнение триангуляции ?? представляет собой простое отношение, где  $x$  — координаты объекта на изображении,  $P$  — проекционная матрица камеры  $3 \times 4$ ,  $X$  — координаты объекта в пространстве.

$$x = PX \quad (3)$$

$$x \times PX = 0 \quad (4)$$

Так как векторы  $x$  и  $PX$  имеют одно и то же направления, их векторное произведение будет равняться нулю (уравнение ??).

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Рисунок 17 – Уравнение триангуляции.

Объединяя уравнения для 2 камер в одну систему, получаем следующее выражение.

В нашем случае объект — номерная пластина с четырьмя координатами углов, а камера всего лишь одна. Однако нам точно известны физические размеры номера, что позволяет компенсировать отсутствие второй камеры.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (5)$$

Введём некоторое допущение, что номер не движется по направлению движения, что позволит сократить координаты  $y$  и  $z$  в уравнении ??, что позволит выразить координаты одной точки номера через другую. Тогда система наших уравнений превращается в однородную.

Путём решения системы этих уравнений посредством сингулярного разложения (SVD) восстановим координаты номера и найдём перемещение как то же декартово расстояние между точками.

### 3.4. Генерация синтетических данных

Для качественной оценки работы алгоритма оценки скорости по видео, необходимо иметь достаточно большой набор данных. Кроме этого, распределение скоростей в этих данных должно располагаться в заявленном диапазоне, для наших приборов этот диапазон составляет промежуток от 0 до 300 км/ч. Так как все реальные данные собраны с тестовых приборов в черте города, где транспортные средства в большинстве своём движутся со скоростями 40-80 км/ч, то среди реальных данных почти невозможно найти примеры с очень большими (более 100 км/ч) или очень маленькими (менее 20 км/ч) скоростями. Поэтому возникает задача генерации синтетических данных с недостающими примерами для полноценной оценки работы алгоритма.

#### 3.4.1. Объект разработки

Прежде, чем приступить к процессу разработки, необходимо определить объект этой разработки. Цель данной секции — разработать программу (или связку программ/сервисов), на выходе которой мы получим RC-файл, идентичный по характеристикам с реальными записями. Теперь перейдём непосредственно к процессу разработки.

Основная часть RC-файла — битовые представления изображений с камеры прибора, из которых строится видео, другими словами, видео — это последовательный набор изображений. Изображение, в свою очередь, является

проекцией объектов пространства на матрицу камеры. Это значит, что для создания идентичного реальному RC-файлу набора данных, необходимо воссоздать 3D-сцену и её проекцию на виртуальную камеру.

После тщательного исследования существующих инструментов и форматов для работы с 3D-сценами, было принято решение воспользоваться программой Blender для создания 3D-сцен и форматом gltf для их последующей обработки. На рисунке ?? представлен пример созданной 3D-сцены в среде разработки Blender. Теперь подробнее остановимся на формате gltf.

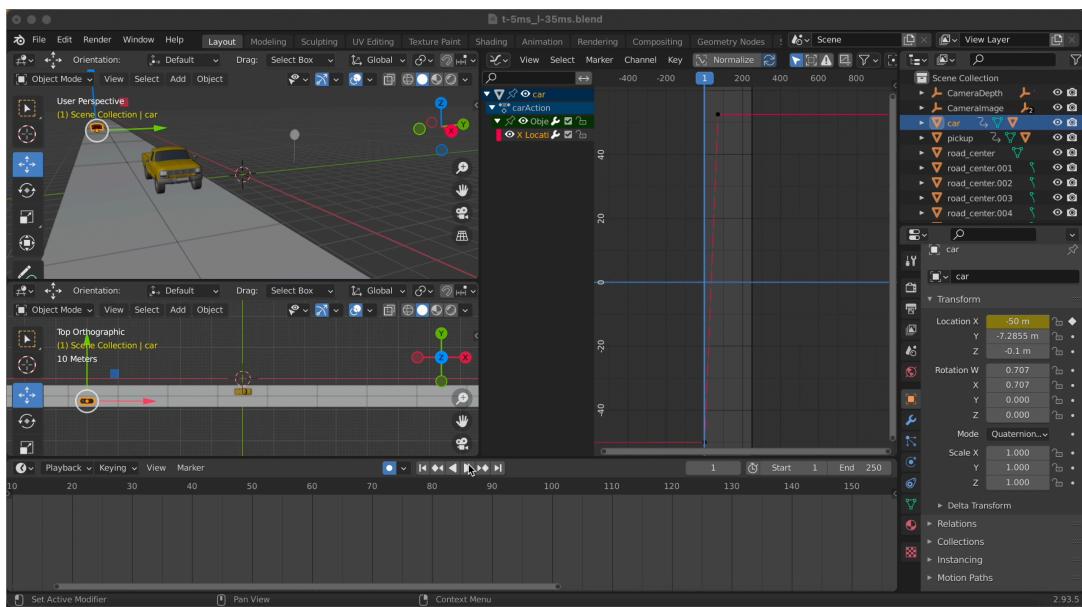


Рисунок 18 – Пример созданной 3D-сцены в среде разработки Blender.

### 3.4.2. GLTF-формат

Формат gltf представляет собой текстовый файл, напоминающий json, только хранит в себе не примитивные типы, а полную информацию о 3D-сцене, такую как формы объектов, их координаты, информация о материалах, текстурные карты, описание анимации, информация о камере (её положение, настройки) и другое. В нашем случае, основными преимуществами данного формата являются простая читаемость, полнота информации, возможность задания любых параметров. А преимуществом работы в программе Blender является то, что можно легко экспортить созданную сцену в необходимый gltf формат.

Таким образом мы можем получить готовый gltf-файл с необходимыми нам параметрами, как, например, скорость и госномер. Однако, создавать новую 3D-сцену для каждого набора параметров – весьма ёмкая задача, поэтом

возникает вопрос автоматизации данного процесса. Посредством модуля `bpy` для языка Python, можно работать с теми же 3D-сценами, задавая или изменяя параметры, тем самым автоматизируя процесс генерации данных. На листинге ?? представлен пример кода, подставляющий госномер в виде текстуры из файла. Аналогичный код был написан для подстановки типа и скорости анимации, далее эти отрывки были помещены в цикл и сгенерированы необходимые gltf-файлы.

Листинг 4 – Пример скрипта для подстановки ГРЗ из файла на объект в 3D-сцене.

```
import bpy
from bpy import context, data, ops

mat = bpy.data.materials.new(name="New_Mat")
mat.use_nodes = True
bsdf = mat.node_tree.nodes["Principled BSDF"]
texImage = mat.node_tree.nodes.new('ShaderNodeTexImage')
texImage.image = bpy.data.images.load("/Users/tina/dev/simicon/
    plates/Y999YY25.jpg")
mat.node_tree.links.new(bsdf.inputs['Base Color'], texImage.
    outputs['Color'])

ob = context.view_layer.objects.active

# Assign it to object
if ob.data.materials:
    ob.data.materials[0] = mat
else:
    ob.data.materials.append(mat)
```

### 3.4.3. Gplay

Итак, мы получили набор gltf-файлов со всеми интересующими нас параметрами, закономерно возникает вопрос: как преобразовать этот формат в необходимый нам полноценный RC. Для этой задачи был разработан сервис gplay, который берёт на вход gltf-файл и по описанной в нём анимации и информации и сцене генерирует RC-файл исключительно с информацией о картинках, то есть видео. Между тем, некоторые из необходимых нам типов данных никак не описаны в gltf-файле, а, соответственно, и в выходном RC. Речь идёт про радарные данные и информацию о распознавании номеров. Решается данная проблема достаточно просто: для восстановления радарных данных можем воспользоваться известными нам координатами объектов в 3D-сцене

(известны из gltf-файла) путём простого пересчёта координат в расстояние до камеры; для заполнения информации о распознанных номерах просто пропустим запись через эмулятор с сервисом распознавания номеров, который дополнит наш RC-файл. На рисунке ?? можно увидеть итоговый RC-файл с заполненными радарными данными и распознанными номерами.

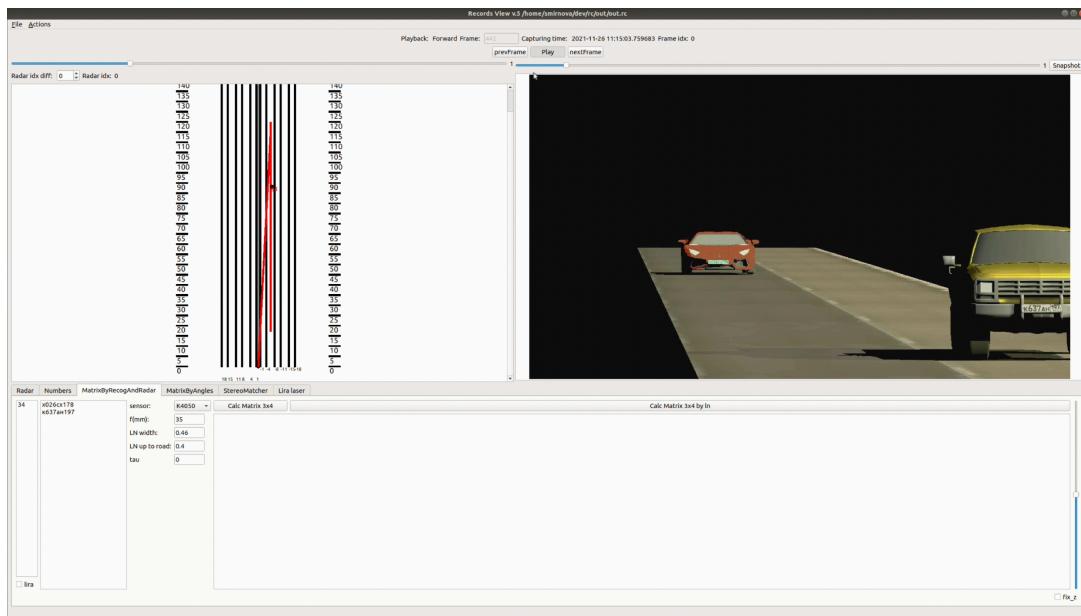


Рисунок 19 – Итоговый RC-файл в программе RecordView.

RecordsView – внутренний инструмент компании «Симикон» для просмотра RC-записей. Справа находится видео, слева – радарные данные, снизу – распознанные номера и возможность задать параметры камеры и рассчитать матрицу 3x4.

### 3.5. Оценка результатов

После того, как скорость транспортного средства посчитана, необходимо провести оценку качества. Из записей RC мы извлекли информацию о радарных целях, её и будем использовать для сравнения.

#### 3.5.1. Матчинг радарных данных

Так как радар — отдельная составляющая прибора, как, собственно, и камера, и работают они независимо друг от друга, мы не можем с точностью сказать, какому транспортному средству соответствует какой радарный трек. Остюда возникает задача матчинга радарных целей с распознанными номерами.

По триангулированным координатам номера мы можем получить положение этого номера в координатах дороги. Радарные цели также представлены в координатах дороги. Остаётся всего лишь определить ближайшие к радарным координатам координаты номера. Для этого просто попарно пройдёмся по всем радарным целем, посчитаем декартово расстояние (формула ??) и найдём ближайший ответ. В данном случае координату  $z$  номера можно опустить, так как нас интересует плоскость дороги.

### **Выводы по главе 3**

В данной части был описан алгоритм нахождения скорости безрадарным методом. В качестве объекта наблюдений был выбран номер как наиболее точно распознанный элемент транспортного средства. Посредством триангуляции, были рассчитаны координаты номера в пространстве, то есть в 3D, и рассчитано его перемещение между фреймами. Также были выбраны оптимальные для вычислений фреймы и рассчитана скорость.

## ЗАКЛЮЧЕНИЕ

В данной работе предложен, реализован и успешно внедрён алгоритм нахождения скорости безрадарным методом. В результате экспериментов на реальных данных, была получена погрешность в 5 км/ч, что является допустимым в рамках данной работы.

Также был сгенерирован синтетический датасет для проверки работоспособности алгоритма на очень больших и очень маленьких скоростях. Была доказана его эффективность.

## ПРИЛОЖЕНИЕ А. ОПИСАНИЯ ФОРМАТОВ

Таблица А.1 – Описание содержания RC-файла..

<b>Имя</b>	<b>Тип</b>	<b>Длина</b>	<b>Описание</b>
magic	uint32_t	4	0x11223307
length	uint32_t	4	Общая длина фрейма, включая поля magic и length
frame data	-	length - 8	-

Таблица А.2 – Описание содержания frame-a.

<b>Имя</b>	<b>Тип</b>	<b>Длина</b>	<b>Описание</b>
id	uint16_t	2	Тип описываемого поля
length	uint32_t	4	Длина данных
data	-	length	

Таблица А.3 – Расшивровка поля ID из frame data.

<b>ID</b>	<b>Данные</b>	<b>Тип/формат данных</b>	<b>Описание</b>
0	frame.capturingtime	struct timeval	Время съемки кадра
3	frame.idx	int32_t	ID кадра
4	frame.frameticks	uint32_t	Время съемки кадра в значениях 90КГц счетчика
5	Результаты распознавания ГРЗ	массив TRectNumber'ов	Распознанные с кадра номерные знаки, достоверности распознавания и координаты
8	Цели с радара	массив umrr_target'ов	Tracked цели с радара UMRR
14	Ширина исходного изображения	int32_t	Ширина исходного изображения
15	Высота исходного изображения	int32_t	Высота исходного изображения
19	Ширина картинки	int32_t	Ширина картинки
20	Высота картинки	int32_t	Высота картинки

21	Размера изображение	int32_t	Размер изображения в байтах (должен быть равен размеру поля 23)
22	Формат картинки	string	JPEG, GRayscale, YUV422P и т.д.
23	Картина	данные	непосредственно сама картина
24	GPS + данные о рейке	MapString	Данные о GPS (COGDEG=342.190002 DDLAT=59.92576700 DDLON=30.40147000 DDRADDR=559751168 HDOP=0.620000 SOGMPS=0.000000 SRC=ACTIVE SRC_TIME=FPGA) и данные рейки (REIKADISTANCE=60 REIKALEN=1.79 REIKALENINPIXELS=74 REIKAPOINTS=0,0,0,0)
51	Количество карт глубины	uint32_t	Количество карт глубины на один фрейм (Nmaps)
51	Матрица 3x4	array<double, 12>	Матрица 3x4
52	Размер одной карты глубины	uint32_t	Размер одной карты глубины в байтах (Dmap)
53	Карты глубины	uint16_t [ ]	Массив последовательно идущих данных карт глубины
54	Преобразование	uint16_t [ ]	Карта соответствия координат карты глубины координатам изображения фрейма
55	Данные камеры	struct sCameraData	Данные о настройках камеры в момент формирования кадра

65	frame.radar_model	enum        class RadarModel: uint32_t	Модель радара
----	-------------------	--	---------------

## ПРИЛОЖЕНИЕ Б. РЕЗУЛЬТАТЫ

Таблица Б.1 – Рассчитанная для записей скорость.

<b>RC-файл</b>	<b>Рас- стояние между фрей- мами</b>	<b>Кол-во целей</b>	<b>Фрей- мов на цель (1/best)</b>	<b>Погрешность (м/с)</b>	<b>Погрешность (км/ч)</b>
1580735812 2020.02.03	30	124	2.666667	1.20622	4.342394
1597823270 2020.08.19	19	67	2.75	0.476229	1.714426
1597823299 2020.08.19	18	57	3.0	1.067738	3.843856
1580735555 2020.02.03	32	352	2.125	0.993482	3.576534
1596722130 2020.08.06	19	62	3.966667	1.36869	4.927286
1597767405 2020.08.18	17	61	3.6	0.938156	3.377363