**1 ОБЗОР ЛИТЕРАТУРЫ**

## **Обзор существующих аналогов**

На этапе проектирования системы было проведено исследование существующих аналогов. Среди обнаруженных аналогов преобладают программные средства, которые распознают дорожные знаки с помощью видеорегистратора или смартфона. Они выводят на экран обнаруженные дорожные знаки и предупреждают пользователя о превышении скорости.

Наиболее приближенным среди найденных приложений является программное средство Roadly для ОС Android.

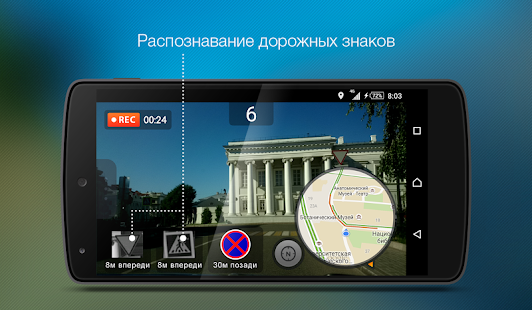


Рисунок 1.1 – Приложение Roadly [2]

Данное приложение является видеорегистратором с возможностью обнаружения на видео дорожных знаков, а также отображения предупреждений пользователю об их присутствии на пути. Обнаруженные на устройстве знаки сохраняются в общую базу, откуда потом они загружаются к другим пользователей. Чем больше пользователей приложения, тем чаще и полнее обновляется база. Кроме того, оно автоматически подстраивает экспозицию, чтобы не допускать переосвещенных областей, а также умеет самостоятельно запускаться при обнаружении движения или при повороте устройства в горизонтальное положение. Приложение может работать в фоне как авторегистратор и выдавать предупреждения поверх любых навигационных приложений, например: Waze, Навител Навигатор (Navitel), Яндекс.Навигатор и других [2].

На данный момент приложение загрузило более 500 000 человек (Google play).

В текущей версии приложение способно распознавать следующие дорожные знаки (см. таблицу 1.1):

* Уступи дорогу;
* Ограничения скорости;
* Обгон запрещен;
* Остановка запрещена;
* Стоянка запрещена;
* Пешеходный переход;
* Знаки, предписывающие проезд перекрестков;
* Предупреждающие знаки.

Таблица 1.1 – Изображения знаков

|  |  |
| --- | --- |
| Уступи дорогу | C:\Users\user\Desktop\2.4_Russian_road_sign.svg.png |
| Ограничения скорости | C:\Users\user\Desktop\40.jpgC:\Users\user\Desktop\60.jpgC:\Users\user\Desktop\90.jpg |
| Обгон запрещен | C:\Users\user\Desktop\2cbe7a1s-960.jpg |
| Остановка запрещена | C:\Users\user\Desktop\загружено.jpg |
| Стоянка запрещена | C:\Users\user\Desktop\3.28 Стоянка запрещена1.jpg |
| Пешеходный переход | C:\Users\user\Desktop\pesheh_perehod.jpg |
|  | |

*Продолжение таблицы 1.1*

|  |  |
| --- | --- |
| Знаки, предписывающие проезд перекрестков | C:\Users\user\Desktop\cfcd1d7afdf91a6424f762b82072ad68.jpg |
| Предупреждающие знаки | C:\Users\user\Desktop\1253_preeduprez.jpg |

К минусам данного аналога можно отнести высокое энергопотребление, нестабильную работу и низкое качество видеозаписи в связи с высоким потреблением ресурсов мобильного устройства.

Также в ходе исследования было обнаружено, что многие известные автопроизводители разработали встроенные приложения либо ведут работы в данном направлении. Как пример можно привести Opel Eye, Speed Limit Assist (Mercedes-Benz), Road Sigh Information (Volvo). Применяемые на автомобилях системы распознавания дорожных знаков имеют типовую конструкцию, которая включает в себя видеокамеру, блок управления и средство вывода информации (экран). Камера снимает пространство перед автомобилем в зоне расположения дорожных знаков (сверху и справа по ходу движения) и передает изображение в электронный блок управления. Тот же в свою очередь по характерным признакам распознает знаки и выводит их на дисплей комбинации приборов или дисплей информационной системы и остается видимым, пока ограничение не закончится или не будет изменено.

Системы с похожим принципом работы встроены в самоуправляемые автомобили. Однако эти системы являются более сложными, так как во время поездки они должны сканировать окружающее машину пространство, распознавая сотни элементов: другие автомобили (едущие и припаркованные на периферии), пешеходов, дорожную разметку и знаки, плотность дорожного потока и свободное пространство на парковке.

* 1. **Аналитический обзор**

Разрабатываемое программное средство будет являться .NET Desktop приложением для OC Windows, не требующим подключения к сети Интернет при наличии на ПК пользователя всех необходимых библиотек (ffmpeg, EmguCV).

Реализация в виде приложения Windows Forms позволит предоставить пользователю удобный интерфейс. Это графические приложения, которые легко развертывать и обновлять, которые могут работать, при подключении к сети Интернет или без него, и могут получать доступ к ресурсам на локальном компьютере более безопасным способом, чем традиционные приложения на основе Windows. В Windows Forms форма – это визуальный компонент, на которой пользователю выводится вся необходимая информация. Обычно приложение Windows Forms строится путем помещения элементов управления на форму и написания кода для ответной реакции на действия пользователя, такие как щелчки мыши или нажатия клавиш. Элемент управления – это отдельный элемент пользовательского интерфейса, предназначенный для хранения, отображения или ввода данных [3].

FFmpeg – бесплатный набор кроссплатформенных библиотек (лицензии GPL 2.0 и LGPL 2.1) который можно использовать в своих проектах, как в коммерческих, так и в свободно-распространяемых. С помощью программы ffmpeg, которую можно запускать из командной строки Windows с необходимыми аргументами, можно конвертировать видео из одного формата в другой, склеивать несколько видеороликов в один, разбирать видео на отдельные кадры (изображения) с определенной частотой, а потом склеивать обратно, ускорять-замедлять, масштабировать, добавлять субтитры и несколько звуковых дорожек [4].

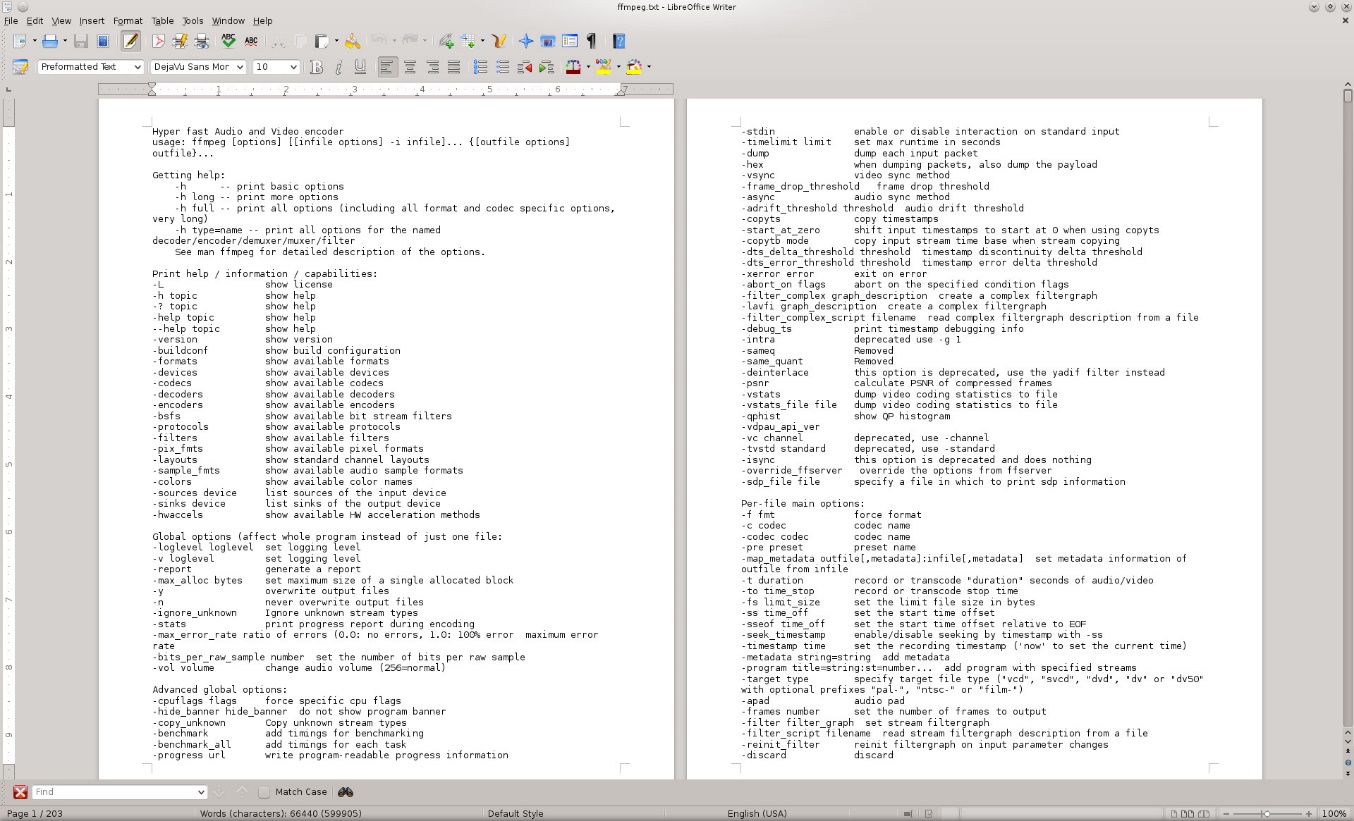
Список доступных возможностей данного набора библиотек приведен на рисунке 1.2.

Рисунок. 1.2 - Список возможностей команды ffmpeg [3]

Emgu CV – это кроссплатформенная оболочка для библиотеки обработки изображений OpenCV, которая позволяет использовать ее функции в фреймворке .Net и создавая приложения на языке с#.

OpenCV (англ. Open Source Computer Vision Library, библиотека компьютерного зрения с открытым исходным кодом) – библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом. Она реализована на C/C++, также разрабатывается для Python, Java, Ruby и других языков. Может свободно использоваться в академических и коммерческих целях – распространяется в условиях лицензии BSD. Фактически, OpenCV – это набор типов данных, функций и классов для обработки изображений алгоритмами компьютерного зрения. В библиотеке присутствует более 2500 оптимизированных алгоритмов, которые включают в себя полный набор как классических, так и современных алгоритмов компьютерного зрения и машинного обучения. Данная библиотека ориентируется в основном на приложения в режиме реального времени и использует преимущества команд MMX и SSE. Кроме того, В настоящее время активно развиваются полнофункциональные интерфейсы CUDA и OpenCL.

Основной идеей создания приложений на платформе .NET Framework является удобство и гибкость разработки за счет предоставления разработчику возможности создавать приложения различных типов, способных запускаться на различных типах устройств и в различных средах. Вторым принципом стала ориентация на семейство систем Microsoft Windows (XP, 7, 8, 10) [5].

* 1. **Событийно-ориентированное программирование**

Важным вопросом при разработке приложения является его архитектура. В данном проекте при разработке пользовательского интерфейса был сделан выбор в сторону событийно-ориентированного программирования (СОП) как наиболее подходящего для поставленных задач. СОП можно также определить как способ построения компьютерной программы, при котором в коде явным образом выделяется главный цикл приложения, тело которого состоит из двух частей: выборки события и обработки события. Событие в данном контексте можно определить как изменение состояния определенного компонента.

В языке C# события реализованы как элемент языка и являются членами классов. Механизм событий здесь реализует шаблон проектирования Publisher-Subscriber. Данный шаблон является вариацией шаблона Наблюдатель. Каждый объект может публиковать сообщения без адресата, а получатели сами подписываются на определенные виды сообщений [5].

* 1. **База данных Oracle**

На данный момент Oracle является одной из наиболее развитых, мощных и производительных баз данных. База Oracle – реляционная база данных, со сложной структурой, но в то же время и с огромными возможностями. По своей функциональности, она не имеет аналогов. Одной из характеристик СУБД Oracle является функционирование системы на большинстве платформ. В том числе на больших ЭВМ, UNIX-серверах, персональных компьютерах и так далее. Одна из отличительных особенностей сервера Oracle — возможность хранения и обработки различных типов данных. Данная функциональность интегрирована в ядро СУБД и поддерживается модулем interMedia в составе Oracle Database. Он обеспечивает работу с текстовыми документами, включая различные виды поиска, в том числе контекстного; работу с графическими образами более двадцати форматов, а также работу с аудио и видеоинформацией. На сегодняшний день она стала обязательной частью практически любой серьезной информационной системы.

Механизмы масштабирования последней версии позволяют безгранично увеличивать мощность и скорость работы сервера Oracle и своих приложений, добавляя новые и новые узлы кластера. Это не требует остановки работающих приложений, не требует переписывания старых приложений, разработанных для обычной архитектуры, состоящей из одного ПК. Кроме того, выход из строя отдельных узлов кластера также не приводит к остановке приложения.

Встраивание в СУБД Oracle JavaVM, полномасштабная поддержка современных серверных технологий (Java Server Pages, Java-сервлеты, модули Enterprise JavaBeans, интерфейсы прикладного программирования CORBA), привело к тому, что Oracle на сегодняшний день фактически является стандартом СУБД для Internet.

Последние версии стали значительно проще в установке и первоначальной настройке. Также возросли возможности по специализированной настройке работы СУБД под конкретную задачу. В результате, и при работе с OLTP-системой, и с хранилищем данных, используя эти возможности по настройке СУБД Oracle, можно достичь поистине впечатляющих результатов.

Также стоит рассказать о грамотной политике Oracle по отношению к обновлению СУБД. Понимая то, что переход с более старой версии СУБД на новую довольно трудоемкая процедура, которая обязательно будет включать в себя тестирование работы существующих приложений в новом окружении, Oracle, при выпуске новых продуктов уделяет большое внимание совместимости в новых версиях с более старыми, делая этот переход максимально удобным для разработчика.

Помимо этого, для перехода и переноса данных из СУБД других фирм (IBM DB2, MySQL, Microsoft SQL Server, Microsoft Access, Sybase, Teradata) в СУБД Oracle, Oracle бесплатно предлагает специальный инструментарий. Обладая удобным графическим интерфейсом, Oracle Migration Workbench в пошаговом режиме, полуавтоматически, поможет выполнить довольно непростую процедуру миграции [6].

* 1. **Entity Framework**

Entity Framework предоставляет разработчику специальную объектно-ориентированную технологию, основанную на базе фреймворка .NET, для работы с данными. Если традиционные средства .NET позволяют создавать подключения, команды и прочие объекты для взаимодействия с базами данных, то Entity Framework представляет функционал для более высокого уровня абстракции, который позволяет абстрагироваться от самой базы данных и работать с данными независимо от типа хранилища. Если на физическом уровне мы оперируем таблицами, индексами, первичными и внешними ключами, но на концептуальном уровне, который нам предлагает Entity Framework, мы уже можем работать с объектами.

Центральной концепцией здесь является понятие сущности или *entity*. Сущность представляет набор данных, ассоциированных с определенным объектом. Поэтому данная технология предполагает работу не с таблицами, а с объектами и их наборами.

Отличительной чертой Entity Framework является использование запросов LINQ для выборки данных из БД. С помощью LINQ мы можем не только извлекать определенные строки, хранящие объекты, из баз данных, но и получать объекты, связанные различными ассоциативными связями.

Другим ключевым понятием является *Entity Data Model* – набор концепций, которые описывают структуру данных независимо от формы, в которой они хранятся в базе данных. Эта модель сопоставляет классы сущностей с реальными таблицами в базе данных. Также она поддерживает набор примитивных типов данных, которые определяют свойства в концептуальной модели.

Ее архитектуру можно разделить на три уровня:

1. Концептуального уровня, где происходит определение классов сущностей, используемых в приложении;
2. Уровня хранилища, который определяет таблицы, столбцы, отношения между таблицами и типы данных, с которыми сопоставляется используемая база данных;
3. Уровня сопоставления (маппинга), который служит посредником между предыдущими двумя, определяя сопоставление между свойствами класса сущности и столбцами таблиц.

Таким образом, мы можем через классы, определенные в приложении, взаимодействовать с таблицами из базы данных.

* 1. **Постановка задачи**

Разрабатываемое программное средство должно отвечать следующим требованиям:

1. Иметь удобный и интуитивно-понятный интерфейс, который в полной мере дает пользователю контроль над работой приложения.
2. В полной мере выполнять необходимый функционал, определенный в поставленных задачах;
3. Не допускать ошибок, приводящих к остановке работы приложения.
   1. **Анализ источников**

Исчерпывающие инструкции по созданию приложения Windows Forms и существующие возможности подобных приложений описаны в [7]. Так же там показаны примеры использования доступных элементов управления и их возможности.

Существующие алгоритмы детектирования и распознавания дорожных знаков, такие как каскады Хаара и нейронные сети, их достигнутая точность и сравнительная характеристика были рассмотрены в [8]. Здесь же сказано, что признаки Хаара состоят из смежных прямоугольных областей. Они позиционируются на изображении, далее суммируются интенсивности пикселей в областях, после чего вычисляется разность между суммами. Эта разность и будет значением определенного признака, определенного размера, определенным образом расположенного на изображении.

Для использования ffmpeg изучена документация и инструкции по использованию, которые подробно расписаны в [9]. Были отобраны необходимые аргументы и их диапазон для лучшего и наиболее полного кадрирования для того, чтобы сохранить баланс между количеством кадров и занимаемым пространством, а также инструменты для экспорта в текстовый файл субтитров.

Для работы с библиотекой EmguCV был изучен [10]. Здесь была обнаружена вся необходимая информация для разработки программ с помощью библиотеки EmguCV. В данном источнике содержится описание всех классов, методов и полей присутствующих в данной библиотеке. Были получены необходимые знания для работы с этой библиотекой в рамках обработки изображений и компьютерного зрения.