**2 СИСТЕМНОЕ ПРОЕКТИРОВАНИЕ**

Изучив теоретические аспекты разрабатываемого средства и выработав список требований необходимых для разработки, разбиваем программу на функциональные блоки (модули).

В разрабатываемом приложении можно выделить следующие блоки:

* блок пользовательского интерфейса;
* блок разделения видеозаписей на кадры;
* блок обработки изображения;
* блок детектирования;
* блок классификации;
* блок приведения полученных данных к одному формату;
* блок экспортирования результатов в базу данных.

Структурная схема, иллюстрирующая перечисленные блоки и связи между ними приведена на чертеже ГУИР.400201.067 C1.

Рассмотрим функциональные блоки приложения.

Блок пользовательского интерфейса представляет собой совокупность средств, при помощи которых пользователь взаимодействует с приложением. Для построения интерфейса используется технология Windows Forms - интерфейс программирования приложений, отвечающий за графический интерфейс и представление данных, получение и обработку ввода пользователя и представляет собой событийно-ориентированное приложение, поддерживаемое Microsoft .NET Framework. Здесь пользователь взаимодействует с программой с помощью различных элементов управления. Так как обработка исходной информации происходит в несколько этапов, то интерфейс изменяется по мере прохождения определенных стадий.

Блок разделения на кадрывыполняет задачу кадрирования полученных видеозаписей для последующей работы с отдельными изображениями. Он является оберткой ffmpeg – набора программ с открытым исходным кодом, которые позволяют обрабатывать, конвертировать и передавать цифровые аудио- и видеозаписи в различных форматах. В данной программе из набора используется только одноименный модуль ffmpeg – утилита командной строки для конвертирования видеофайла из одного формата в другой. Он вызывается как отдельный процесс, в который передается набор аргументов.

Блок обработки изображения необходим для улучшения количественных и качественных характеристик модулей детектирования и классификации. Здесь происходит преобразования изображений в наиболее удобные для работы форматы и цветовые форматы. Функционал основан на использовании библиотеки EmguCV.

Блок детектированияработает на основе алгоритма каскадов Хаара, используя Виолы-Джонса для нахождения знаков на изображении. В методе Виолы-Джонса основу составляют примитивы Хаара, представляющие собой разбивку заданной прямоугольной области на наборы разнотипных прямоугольных подобластей. В данном блоке используется класс CascadeClassifier библиотеки EmguCV. Для работы алгоритма необходимы файлы обученных каскадов. Для обучения необходимо не менее 1000 положительных и столько же отрицательных примеров и около суток для обучения. Результат обучения хранится в xml файле, который и используется данным блоком.

Блок классификациипредставляет собой сверточную нейронную сеть, которая после обучения на подготовленной выборке производит классификации дорожных знаков на найденных на предыдущем этапе прямоугольных фрагментах изображения.

Блок получения координатсодержит необходимые методы получения координат дорожных знаков используя исходные видеозаписи.

Блок приведения полученных данных к одному форматунеобходим для стандартизации полученных при обработке видеозаписей в ходе работы программы данных к формату, пригодному для экспортирования в базу данных. Каждый подготовленный элемент содержит название знака, его координаты и время его обнаружения на видеозаписи, что позволит контролировать изменение дорожных знаков на определенных участках дорог.

Блок экспортирования результатовв базу данныхсодержит программный код для сохранения полученных результатов в реляционную sql базу данных. Пользователь указывает место хранения данных результатов предыдущего блока.