

РАЗРАБОТКА СИСТЕМЫ РАСПОЗНАВАНИЯ И ДОСТУПА ПО ФИЗИОГНОМИЧЕСКИМ ПРИЗНАКАМ НА ОСНОВЕ ОДНОПЛАТНОГО КОМПЬЮТЕРА RASPBERRY PI

Муравьев Даниил Олегович

магистрант, Алматинский Университет Энергетики и Связи имени Гумарбека Даукеева, Казахстан, г. Алматы

Жуматаев Ильяс

магистрант, Алматинский Университет Энергетики и Связи имени Гумарбека Даукеева, Казахстан, г. Алматы

Нурбай Дарын

магистрант, Алматинский Университет Энергетики и Связи имени Гумарбека Даукеева, Казахстан, г. Алматы

Байкенов Бахытжан Сергеевич

научный руководитель, канд. техн. наук, доцент Алматинский Университет Энергетики и Связи имени Гумарбека Даукеева, Казахстан, г. Алматы

Видеонаблюдение является основным типом контроля для любого процесса. Отрасль транспортных перевозок не является исключением. Контроль необходим на любом этапе, без видеонаблюдения требуются огромные человеческие ресурсы и не всегда наличие таких ресурсов гарантирует качество. На смену людям, в задаче мониторинга, пришли компьютерные системы. Они смогли обеспечить надёжность и автономность, как при грузоперевозках, так и в любой промышленности. Свести человеческий фактор к минимуму и исключить вредоносное внешнее влияние, также, является ключевым преимуществом цифровых систем. СКУД системы, помимо непосредственно работы по выдаче допуска на промышленные объекты, также активно начали внедряться в структуры транспортных перевозок. Данная тенденция обусловлена необходимостью в повышении безопасности для грузов. В данной работе, рассмотрим вариант реализации экономичной системы контроля и управления доступом, которая обладает всем необходимым функционалом.

Основой системы будет служить одноплатный компьютер Raspberry Pi 4. Raspberry отлично подходит для создания систем удалённого контроля, имея ряд преимуществ в создании небольших проектов. Для простоты взаимодействия компьютера и камеры используется Raspberry Pi CameraV2. Камеру можно подключить к последовательному интерфейсу MIPI в Raspberry Pi (рисунок 1).[1]

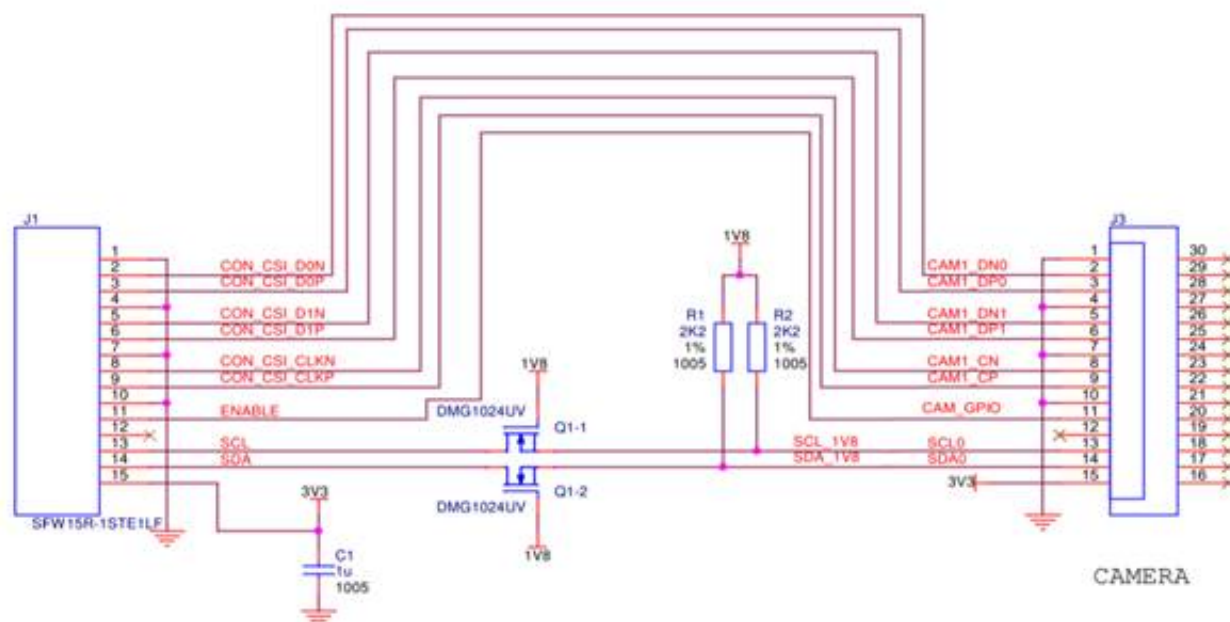


Рисунок 1. Электрическая схема подключения камеры

В данной работе будет использоваться метод Виолы-Джонса для захвата объектов, данный выбор обусловлен по соотношению показателей эффективность распознавания/скорость работы. Среда обнаружения объектов Виолы-Джонса - это подход машинного обучения для обнаружения объектов, предложенный Полом Виолой и Майклом Джонсом в 2001 году. Эту структуру можно обучить обнаруживать практически любой объект, но это в первую очередь решает проблему обнаружения лиц в режиме реального времени.

Объекты классифицируются по очень простым характеристикам как функция кодирования специальных знаний предметной области и работают намного быстрее, чем пиксельная система. Эта функция похожа на фильтры Хаара, отсюда и название «Хаар». Примером этих функций является функция с двумя прямоугольниками, определяемая как разность суммы пикселей площади внутри прямоугольника, которая может иметь любое положение и масштаб в пределах исходного изображения.[2]

В качестве алгоритма распознавания будет использоваться LBPH. Данный алгоритм интерпретирован в библиотеке OpenCV для python. OpenCV — это библиотека компьютерного зрения с открытым исходным кодом, включающая в себя различные алгоритмы компьютерного зрения, распознавания изображений и многое другое, работающее в реальном режиме времени. Структуру модулей библиотеки изображена на рисунке 2.

Используя LBP в сочетании с гистограммами, мы можем представить изображения лиц с помощью простого вектора данных. Поскольку LBP является визуальным дескриптором, его также можно использовать для задач распознавания лиц, чем и воспользуемся. [3]

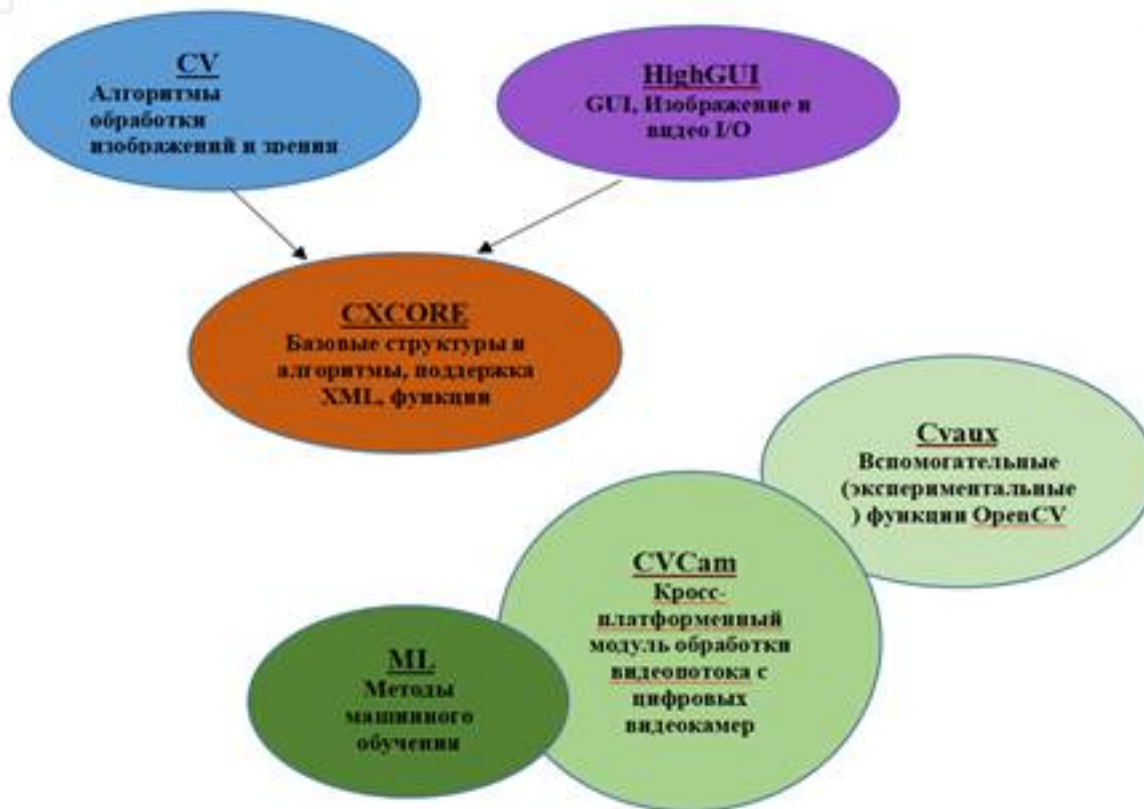


Рисунок 2. Структура модулей OpenCV

Алгоритм работы системы можно поделить на 3 этапа: Сбор данных, обучение распознаванию и само распознавание (рисунок 1).

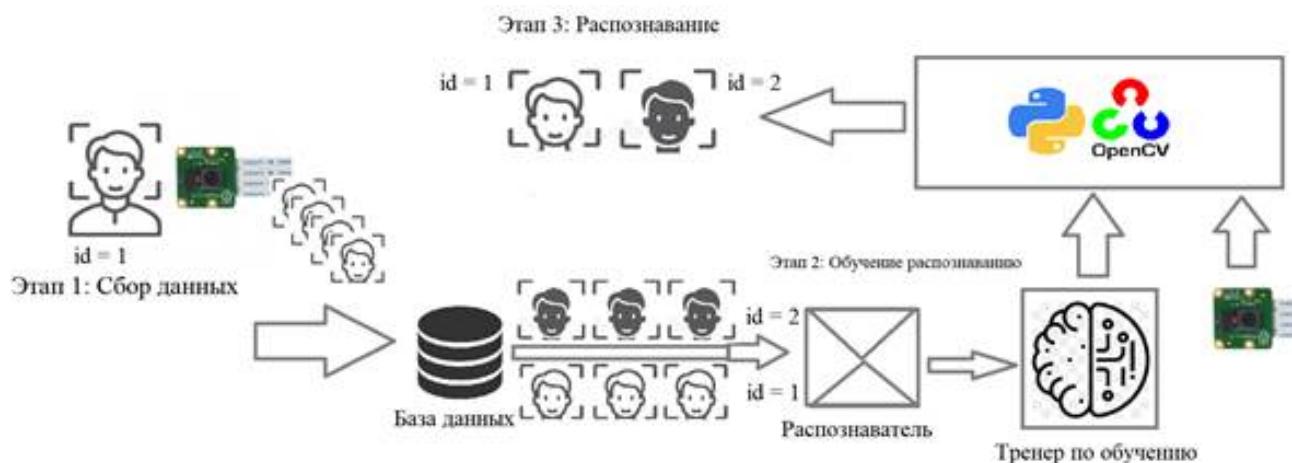


Рисунок 3. Алгоритм системы

В данной работе для обнаружения объекта будет использован «Классификатор Хаара». Классификатор обучается на базе алгоритма, которому требуется множество изображений, среди них должны быть как с лицом, так и без. Необходимо извлечь из классификатора особенности. Инструктор и детектор уже находятся внутри библиотеки OpenCV. Есть возможность создать новый классификатор на базе библиотеки, но в данном проекте будет использоваться уже созданный. Установим камеру и внутри цикла и загрузим входное видео в

режиме градаций серого. Далее мы «помечаем» грани на изображении, используя, например, синий прямоугольник, если грани найдены, он возвращает позиции обнаруженных граней в виде прямоугольника с левым верхним углом (x, y) и имеет «w» в качестве своей ширины и «h» в качестве высоты (рисунок 4).

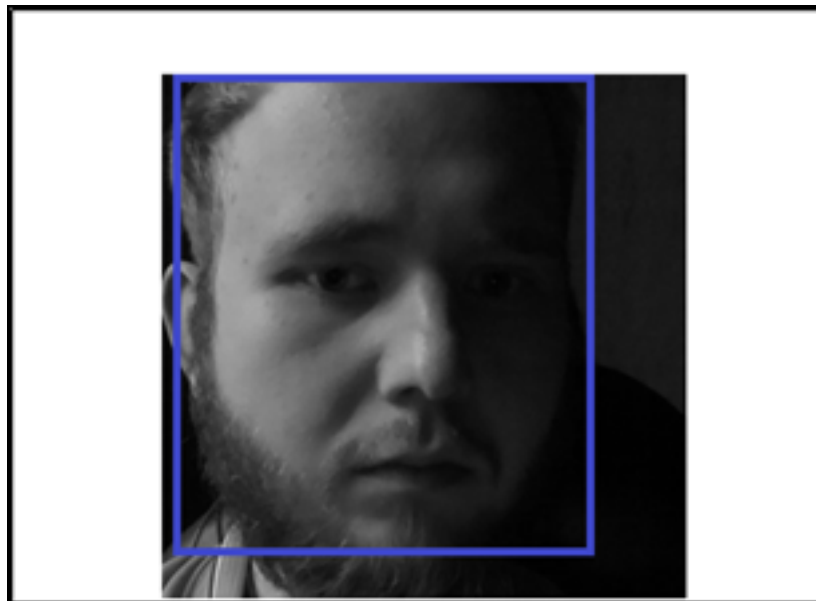


Рисунок 4. Обнаружение лица

На этапе «сбора данных», просто создаём набор данных, в котором будем хранить для каждого идентификатора группу фотографий в сером цвете с той частью, которая использовалась для обнаружения лица. Для реализации этой задачи удобней всего написать скрипт, который будет отгружать фото в определённую папку (рисунок 5).

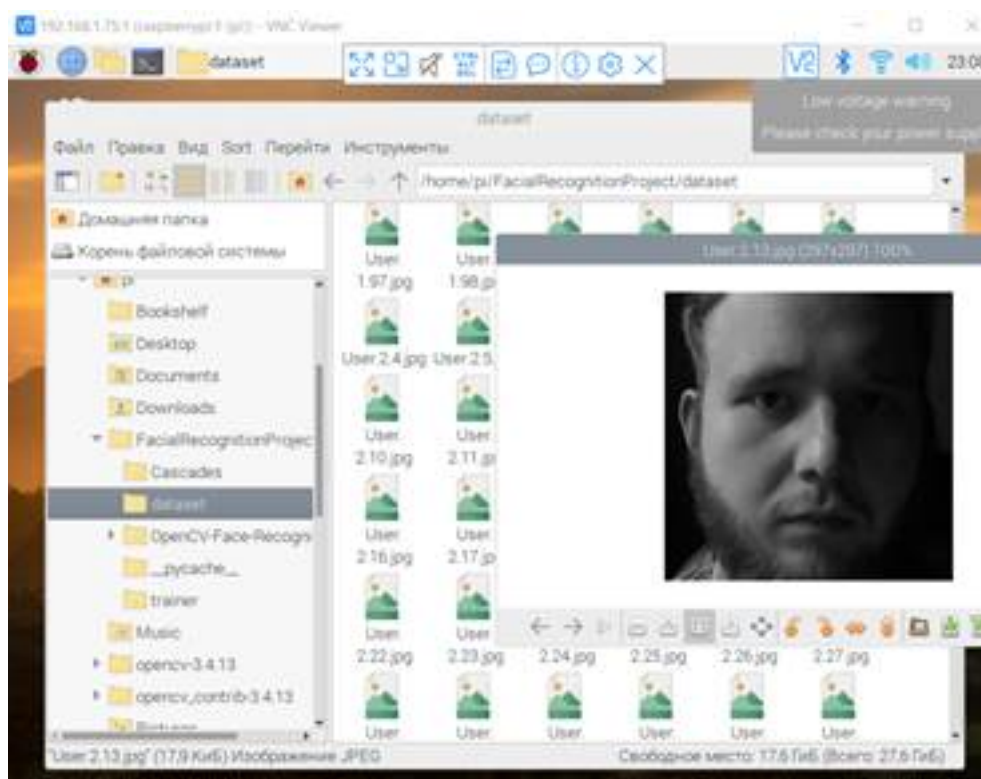


Рисунок 5. Сбор данных

На этапе «обучение» возьмём все пользовательские данные из нашего набора данных "trainer" OpenCV Recognizer. Это делается напрямую с помощью определенной функции OpenCV. Результатом будет файл. yml, который будет сохранен в каталоге «trainer /». Распознавателем будет выступать LBPН. Происходит это командой: recognizer = cv2.face.LBPHFaceRecognizer_create(). Функция «getImagesAndLabels (path)» будет принимать все фотографии в каталоге: "dataset/" , возвращая 2 массива: "Ids" и "faces"(идентификаторы и лица).

И финальным этапом будет реализация распознавания нового лица на камере. При отображении на данной камере, и, если у этого человека было запечатлено и обучено его лицо, распознаватель сделает прогноз, возвращающий его идентификатор и индекс, показывающий, насколько оно вероятно верно.

Далее реализуем имитацию системы к которой требуется допуск. В качестве объекта допуска возьмём сервопривод SG90. Дополняем последний скрипт кодом движения на определённый угол сервопривода, а в качестве параметра используем индекс вероятного распознавания. Можно изменять параметр индекса в зависимости от окружающего освещения, для достижения оптимальной работы (рисунок 6).

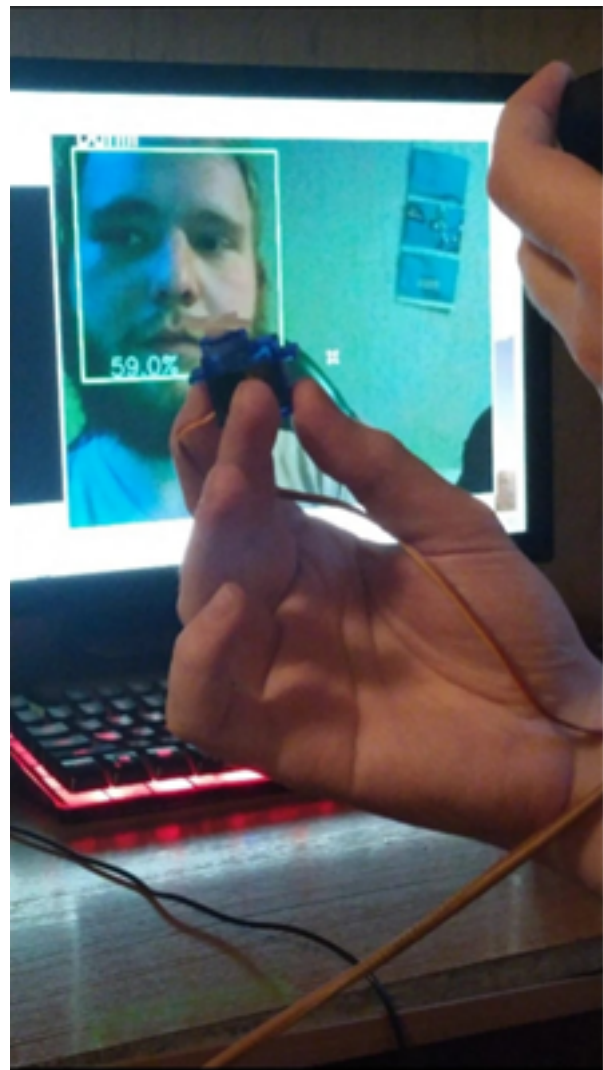


Рисунок 6. Демонстрация допуска

Список литературы:

1. Raspberry Pi Documentation [Электронный ресурс]. - 2016. - URL: <https://www.raspberrypi.com/documentation/accessories/camera.html> (дата обращения: 26.04.2021).
2. P. Viola and M.J. Jones, «Robust real-time face detection», International Journal of Computer Vision, vol. 57, no. 2, 2004., pp.137-154
3. Trefný, Jiri, and Jiri Matas. "Extended set of local binary patterns for rapid object detection." Proceedings of the Computer Vision Winter Workshop. Vol. 2010. 2010.

