

План выступления на НТК

**Стек обработки входных данных видеопотока для системы контроля
управления доступом.**

Сатаев Д. А.

Научный руководитель: Борисов Н. К.

РТУ МИРЭА, Институт Искусственного интеллекта

***Аннотация:** в докладе будет рассматриваться стек обработки изображений или видеозаписей системой распознавания образов, а также разработка графического интерфейса системы.*

***Ключевые слова:** распознавание, СКУД, изображения, видеозаписи*

Введение

Защита любого объекта включает в себя несколько аспектов, число которых зависит от уровня режимности объекта. При этом во всех случаях важным аспектом будет система управления контроля доступом (СКУД) - это совокупность программно - аппаратных технических средств контроля и средств управления, имеющих целью ограничение и регистрацию входа-выхода объектов (людей, транспорта) на заданной территории. Хорошо организованная с использованием современных технических средств СКУД позволит решать целый ряд задач. К числу наиболее важным можно отнести следующие:

- учет рабочего времени;
- контроль своевременности прихода и ухода сотрудников;
- защита конфиденциальности информации;
- регулирование потока посетителей;
- контроль въезда и выезда транспорта.

Распознавание образов в системе происходит с помощью искусственных нейронных сетей, которые прочно вошли в нашу жизнь и в настоящее время широко используются при решении самых разных задач: распознавание речи, звука, диагностика в медицине, анализ и прогнозирование экономических показателей. Искусственный интеллект активно применяется там, где обычные алгоритмические решения оказываются неэффективными или вовсе невозможными.

В данной практической работе будет исследована проблема распознавания образов с помощью использования искусственных нейронных сетей. Практической частью данной работы является разработка программного обеспечения с использованием новейших фреймворков для распознавания образов.

Средства разработки.

Программное обеспечение реализовано на языке Python версии 3.7.6. в текстовом редакторе Visual Studio Code, с использованием графического фреймворка PyQt5, библиотеки алгоритмов компьютерного зрения OpenCV и библиотек машинного обучения - TensorFlow и PyTorch. Разработанное ПО удовлетворяет всем функциональным требованиям, поставленным на этапе постановки задачи.

Основные функциональные требования

1) Графический интерфейс.

Работа с графическим интерфейсом должна быть интуитивно понятна любому пользователю. К тому же, пользователь должен иметь возможность просматривать изображения и видеозаписи внутри графического интерфейса разрабатываемой программы.

Система должна предоставить возможность пользователю загружать изображения и видеозаписи из файловой системы с помощью интуитивно понятного интерфейса.

2) Распознавание объектов.

При обнаружении объекта на изображении или видеозаписи система должна отрисовывать рамку (бокс) на выходном изображении или видеозаписи с точностью определения.

Пользователь должен иметь возможность настраивать порог обнаружения объектов и менять предобученную модель нейронной сети.

3) Распознавание лиц.

В случае обнаружения человека система должна определить его лицо, снять facemask и попытаться найти совпадение в базе данных и затем сохранить лицо в файловую директорию и, вдобавок происходит сохранение в базу данных в blob формате. Для того, чтобы система искала совпадения в своей базе данных необходимо предоставить пользователю

возможность добавлять людей для сравнения с обнаруженным человеком. К тому же, лицо обнаруженного человека должно корректно отображаться.

Разработка графического интерфейса.

Разработка системы распознавания образов началась с создания главного окна, которое будет открываться при каждом запуске приложения и предоставлять пользователям доступ к функциям приложения.

1. Главное окно приложения содержит следующие элементы:

В левой верхней части экрана отображаются входные файлы для системы распознавания, с помощью двойного клика можно запустить процесс распознавания.

В левой нижней части экрана отображаются файлы, полученные после обработки нейронной сетью, с выделенными объектами. Программа сама распределяет объекты по папкам и вырезает объекты со входного изображения. С помощью двойного клика можно вызвать отдельное окно для просмотра изображения.

По центру располагается два виджета, отображающих файл до обработки искусственным интеллектом и после.

В нижнюю консоль выводится информация о детекциях.

С помощью вкладок, пользователь может открыть журнал детекций и настройки программного обеспечения.

2. Окно настроек приложения содержит следующие элементы:

В верхней части окна располагается слайдер, с помощью которого пользователь может отредактировать значение порога обнаружения. Если у объекта будет вероятность ниже чем порог обнаружения, то он не будет выделяться на выходном изображении.

Ниже находятся поля для ввода пути до выходных и входных файлов и модели нейронной сети.

Диаграмма классов

- **MainWindow** – основной класс, который выступает в качестве связывающего звена между остальными компонентами, а также является ответственным за отрисовку графического интерфейса.
- **Facerecognition** – класс, отвечающий за обработку лица персоны, с помощью которого можно получить facemask, если лицо обнаружено.
- **YoloV5** – класс, отвечающий за конфигурацию и запуск нейронной сети YoloV5.
- **XML** – класс, отвечающий за хранение, сохранение и получение конфигурации пользователя из файла Settings.xml.
- **SqliteManager** – класс, отвечающий за взаимодействие системы с базой данных, которая хранится локально в файловой директории.

Распознавание объектов

С самого начала обработки изображение поступает в нейронную сеть YoloV5, где происходит обнаружение и классификация объектов на основе предобученной модели. Затем происходит сохранение полученного изображения, с нанесенными боксами и точностью определения. После этого каждый объект вырезается отдельно и сохраняется в файловой системе, затем добавляется в базу данных в формате blob. Если обнаруженный объект — это персона (человек), то модуль FaceRecognition пытается обнаружить лицо на полученном объекте и в случае успеха получает facemask и тоже добавляет в базу данных в формате blob, в соответствующую таблицу.

YoloV5

Yolo - это аббревиатура расшифровывается как “You look only once” (Стоит только раз взглянуть). Yolo - современный алгоритм глубокого обучения, который широко используется для обнаружения объектов. Он был разработан Джозефом Редмоном и Али Фархади в 2016 году.

YoloV5 - усовершенствованная пятая версия Yolo, реализованная на фреймворке PyTorch.

Face Recognition

Face recognition – это библиотека Python, которая использует dlib – современный фреймворк C++, который содержит алгоритмы машинного обучения.

Первая задача модуля FaceRecognition – это обнаружение лиц на изображении или в видеозаписи. Первоначально этот модуль получает координаты лица и берет его для дальнейшей обработки. Затем происходит процедура под названием эмбединг: нейронная сеть принимает на вход

изображение, а на выходе возвращает числовой вектор, характеризующий основные признаки данного лица (facemask). Данный вектор как раз и называется эмбедингом.

Заключение

В работе было рассмотрено решение поставленной задачи - разработка графического интерфейса и модуля распознавания системы контроля и управления доступом. За отведенное время удалось поработать с нейронными сетями, изучить высокоуровневый язык программирования Python, а также фреймворк PyQt5.

Список литературы

1. <https://face-recognition.readthedocs.io/en/latest/readme.html>
2. <https://doc.qt.io/qtforpython/>