**Пояснительная записка к проекту Computer Vision Face Recognition. Выполнил студент 2 курса УрФУ департамента математики и компьютерных наук, практикант на комбинате “ЭХП”.**

Была поставлена задача распознавания лиц сотрудников, входящих в здание, для проставления отметок в базе данных об их присутствии или отсутствии на рабочем месте. В ходе решения поставленной задачи был разработан проект в области Computer Vision Face Recognition, пригодный для работы с любыми Web или IP-камерами.

**Для пользователя:** для того, чтобы полноценно начать пользоваться программой, достаточно нажать левой кнопкой мыши 2 раза на иконку “Начало работы”. Появится графический интерфейс пользователя с 3 кнопками:

“**Начать фотографирование**” – эта кнопка призвана помочь Вам в заполнении базы данных для конкретного рабочего. При нажатии этой кнопки вам будет предложен ввести табельный номер и ФИО рабочего. При нажатии кнопки подтверждения данных, смотрите в камеру, появится окно и поясняющая надпись - работнику надо будет сидеть перед камерой и медленно поворачивать лицо под как можно большим количеством углов – для лучшего распознавания в дальнейшем. В тот момент, когда необходимое количество фотографий будет взято, приложение автоматически завершит свою работу. Примечание: количество фотографий определено в тексте программы и в данном случае принято на уровне 150 штук.

“**Тренировка распознавателя**” – при нажатии этой кнопки, Вы обучаете нейросеть новому лицу. Рекомендуется сначала взять фотографии у как можно большего числа рабочих, и лишь потом начинать обучение сети, так как оно займет время и в это время приложение будет недоступно во избежание прерываний в обучении.

“**Распознавание**” – эта кнопка передает управление в приложение по распознаванию лиц. Приложение автоматически заносит информацию о присутствии человека на работе в базу данных, которые могут быть извлечены и проверены контроллерами, мастерами и другими уполномоченными людьми. Для того, чтобы завершить работу распознавателя нажмите кнопку “Завершить”, которая появится после нажатия кнопки “Распознавание”. Перед тем, как начать непосредственно распознание образов должно пройти некоторое время для того, чтобы распознаватель считал результаты обучения, поэтому запускать его надо заранее. Время до непосредственного распознавания будет различаться в зависимости от размера базы данных.

**Для последующих разработчиков:** в файле “**Начало работы.ру**” – описана графическая оболочка пользователя – все взаимодействие пользователя с программой происходит посредством этого графического интерфейса. Класс Main – точка входа в программу. \_\_init\_\_() – вызывает метод, ответственный за инициализацию графических компонент, в том числе и кнопок, к которым по параметру command=передается метод, который будет выполнен при нажатии этой кнопки – ее обработчик. Обработчики вызывают скрипты, которые лежат в папке scripts. Их мы импортируем в файл “Начало работы.py”

**database\_connection.ру** – облегчает подключение к базе данных, если изменится база данных – просто поменяйте параметры в этом скрипте, и во всей остальной программе подключение будет происходить к нужной вам базе данных.

**frontal\_face\_dataset\_with\_database.py** – скрипт, который вызывается по нажатию кнопки “Начать фотографирование” пользовательского интерфейса, описанного в пояснительной записке для пользователя. Этот скрипт захватывает видеопоток с камеры методом cv2.VideoCapture(n) – где n – номер вашего оборудования (одна веб камера – 0, две – 0 и 1, IP-камера – IP адрес этой камеры).

Работа происходит с кадрами из видеопотока – изображение переводится в градации серого, на них находится лицо с помощью каскадов Хаара, которые хранятся в директории Cascades/, и фотография только лица сохраняется в каталоге “dataset/”. Так же ФИО и табельный номер сотрудника вносятся в базу данных. Более подробное описание методов, использованных в этой программе, приведено в описании библиотеки openCV.

**frontal\_faces\_training\_with\_reading\_and\_cleanup.py** - скрипт, который вызывается нажатием кнопки “Тренировка распознавателя” в графическом интерфейсе пользователя. Этот скрипт учит нейронную сеть изображениями, которые находятся в папке dataset/, а затем, после обучения стирает все фотографии из dataset/ для того, чтобы освободить ресурсы, так как в дальнейшем эти фотографии не понадобятся для переобучения, так как результат обучения ими нейронной сети уже отражен в файле trainer.yml, который в процессе работы генерирует в каталоге trainer/ скрипт.

Имеется 2 скрипта для распознавания образов. Один с графическим интерфейсом пользователя - **frontal\_face\_recognition\_with\_GUI\_and\_database.py**, и без него - **frontal\_face\_recognition\_with\_only\_database.py**. Подобно скрипту для фотографирования находится лицо на изображении, которое так же выхватывается из видеопотока камеры (то есть распознавание происходит в реальном времени). Далее распознаватель пытается определить какому человеку из базы данных принадлежит это лицо. Он определяет это с определенной степенью уверенности, варьируя степень уверенности, можно получать более точные или более обширные данные (параметр confidence). Рекомендованный диапазон значений – 40-70 (50-60). Этот скрипт проставляет в базу данных отметки о присутствии и отсутствии работника.

**Requirements или немного об установке:**

Python==3.7.0 – бесплатно скачивается с официального сайта – [www.python.org](http://www.python.org).

На сайте в разделе downloads надо скачать последнюю стабильную версию языка, на текущий момент это 3.7.0, скачается .ехе файл, который необходимо будет установить.

Дальнейшие модули ставятся на этот интерпретатор python бесплатно все сразу, командой **pip install requirements** в командной строке cmd в каталоге с программой, при подключенном интернете (в папке с программой имеется .txt файл, с прописанными внутри библиотеками, и, выполняя команду pip install requirements – будут поставлены и установлены все библиотеки из списка в этом файле). Библиотеки скачиваются из общего репозитория в интернете, следовательно, должно быть подключение к сети. Библиотеки скачиваются архивом и распаковываются в папку интерпретатора python, в папку Lib/. Если на данной машине нету подключения к сети Интернет, то на другой машине, где есть подключение, надо поставить интерпретатор со всеми модулями, а потом на диске или Flash накопителе перенести на нужную машину без интернета. Можно переносить всю папку с интерпретатором, так как в ней уже будут и все модули.

Так же можно поставить все необходимые модули по отдельности вручную, командами, написанными ниже для каждой конкретной библиотеки (команда pip install находит .whl файл в общем репозитории в интернете, скачивает его и распаковывает в папку с интерпретатором в поддиректорию Lib/)

**numpy** **- pip install numpy**

(библиотека для работы с массивами – преобразование изображений в массив, нужна для установки многих других библиотек)

**matplotlib - pip install matplotlib**

(библиотека для построения графиков и математических расчетов, нужна для того, чтобы поставилась библиотека opencv-contrib-python3.7.0)

**PIL - pip install pillow**

(библиотека для работы с изображениями)

**pypyodbc - pip install pypyodbc**

(библиотека для работы с базами данных)

**opencv-contrib-python-3.7.0 - pip install opencv-contrib**

(библиотека для распознавания образов и обучения нейронной сети)

Так же можно скачать .whl файлы из интернета, найденные при поисковых запросах о загрузке того или иного модуля и в cmd, командой pip install (путь до .whl файла).whl поставить эту библиотеку.