

Project

Presence control using face recognition

1. Requirements

Installation requirements:

Edge device and server:

Operating system: Linux (preferably Ubuntu , tested on versions 16.04 and 22.04)

Language/Compiler: Python 3.6

Hardware requirements (server):

- Multiprocessing support
- Network connection between server and edge devices
- Fixed IP

Hardware requirements (edge device):

- Multiprocessing support
- usb camera
- <https://solectroshop.com/pl/moduly-rfid/1753-modul-rfid-pn532-nfc-uart-i2c-spi.html> with adapter for USB interface (GND, 5V, RX, TX outputs)
- Stable network connection between server and edge devices. Wired and wireless connectivity supported

Required libraries to install:

- Packages listed in the installation list below (must be run before installing OpenCV and FaceRecognition packages)
- Mosquito MQTT (sudo apt-get install mosquitto and sudo apt-get install mosquitto - clients)
- OpenCV (pip install opencv -python)
- Face-recognition(pip install face-recognition)
- Imutils (pip install imutils)
- Pickle(pip install pickle- mixin)
- Paho.mqtt.client (pip install paho-mqtt)
- Psutil (pip install psutil)
- https://github.com/carolinedunn/facial_recognition (git clone https://github.com/carolinedunn/facial_recognition)
- CLIENT ONLY: libnfc -bin libnfc -dev libnfc -examples

Installation list required for OpenCV and face - recognition :

1. sudo apt install cmake build-essential pkg-config git
2. sudo apt install libjpeg -dev libtiff -dev libjasper -dev libpng -dev libwebp -dev libopenexr -dev
3. sudo apt install libavcodec -dev libavformat -dev libswscale -dev libv4l-dev libxvidcore -dev libx264-dev libdc1394-22-dev libgstreamer-plugins-base1.0-dev libgstreamer1.0-dev

4. `sudo apt install libgtk-3-dev libqtgui4 libqtwebkit4 libqt4-test python3-pyqt5`
5. `sudo apt install libatlas-base-dev liblapacke-dev gfortran`
6. `sudo apt install libhdf5-dev libhdf5-103`
7. `sudo apt install python3-dev python3-pip python3-numpy`

2. Configuration and commissioning

2.1 Server (database preparation)

The data structure on the server is stored as follows:

Dataset `/{student index}/{student photos}`

Dataset `/{student index}/{student attendance file}`

Dataset `/{student index}/{student recognition file}`

In order to prepare the database, place verified photos of students in the above-mentioned location and then execute the command

```
" python train_model.py {student index} "
```

for each new student or to recalculate an existing student.

```
#!/usr/bin/python

# import the necessary packages
from imutils import paths
import face_recognition
#import argparse
import pickle
import cv2
import os
import sys
# our images are located in the dataset folder
print("[INFO] start processing faces...")
imagePaths = list(paths.list_images(f'dataset/{sys.argv[1]}'))

# initialize the list of known encodings and known names
knownEncodings = []
knownNames = []

# loop over the image paths
for (i, imagePath) in enumerate(imagePaths):
    # extract the person name from the image path
    print("[INFO] processing image {}/{}".format(i + 1,
        len(imagePaths)))
    name = imagePath.split(os.path.sep)[-2]

    # load the input image and convert it from RGB (OpenCV ordering)
    # to dlib ordering (RGB)
    image = cv2.imread(imagePath)
    rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
```

```

# detect the (x, y)-coordinates of the bounding boxes
# corresponding to each face in the input image
boxes = face_recognition.face_locations(rgb,
    model="hog")

# compute the facial embedding for the face
encodings = face_recognition.face_encodings(rgb, boxes)

# loop over the encodings
for encoding in encodings:
    # add each encoding + name to our set of known names and
    # encodings
    knownEncodings.append(encoding)
    knownNames.append(name)

# dump the facial encodings + names to disk
print("[INFO] serializing encodings...")
data = {"encodings": knownEncodings, "names": knownNames}
f = open(f"dataset/{sys.argv[1]}/{sys.argv[1]}.pickle", "wb")
f.write(pickle.dumps(data))
f.close()

```

The data structure on the client is stored as follows:

Dataset /{student index}/{student recognition file}

Dataset /{student index}/{student attendance file}

2.2 Server (MQTT):

After installing the mosquito mqtt and the mqtt client (ubuntu : sudo apt install -y mosquitto AND sudo apt install -y mosquitto-clients)

First, it is required to edit the configuration file: sudo nano / etc / mosquitto / mosquitto.conf

Where to add the code:

" listener 1883

allow_anonymous true "

Then start the mqtt broker (sudo / etc / init.d / mosquitto start)

Server (server.py file):

In order to start the server, it is required to define the "broker" variable and set the IP address of the server on which the MQTT broker was started

To link the server with the edge device, you must give the edge device a unique ID (a string of letters and numbers)

Permanent link:

Add the device ID to the "EDGEIDS" list

Temporary binding:

After starting the server, enter the device ID on the input

The server is started by executing the command: " python server.py"

If a new edge device has been added to the network or student data has been manually updated, the server must be restarted to upload face recognition data or provide "UPDATE" to the server input.

2.3 Customer

After installing the mosquito mqtt and the mqtt client (ubuntu : sudo apt install -y mosquitto AND sudo apt install -y mosquitto-clients)

Configure the client file:

Set the "broker" variable to the IP of the MQTT broker

Define a list of teacher/teacher/admin cards by adding their unique IDs to the " admincardID " list

Have at least 1 (there may be an example attached to the client's package) . pickle in " dataset /{ uniqueID }/"

Before starting the client, you must give the device a unique ID (a string of letters and numbers)

After unpacking the package with the client source code, follow the instructions below:

1. Go to the root folder with the source codes
2. Execute the command " make installconfig "
3. Execute the command " ls / dev / ttyUSB *"
4. Connect the NFC card reader
5. Execute the command " ls / dev / ttyUSB *"
6. Identify which reader ID has appeared and record it
7. Complete the configuration file (/etc/nfc/devices.d/pn532_via_uart2usb.conf) Reader ID
8. Execute the command " gcc -o reader.o reader.c - lnfc "

After executing the above instruction, we run the client with the command ". / reader.o | python client.py { uniqueID } "

3. Customer service

After starting, the client is in "passive mode" where it only receives packets used to update the facial recognition database.

In order to activate the presence system, the same administrator/leader card should be presented to the reader twice

After starting the attendance system, to register attendance, look at the camera and apply the ID card until a window with a photo with an index appears above the student's face. If the photo appears, the presence has been registered correctly. (Note: Only one person should be in the frame!)

If the message "UNRECOGNIZED" is displayed above the student's face in the photo, you can re-measure by selecting the "r" (reset) key and repeat the actions from the point above.

If the system is unable to recognize the person repeatedly, select the "s" key (save) to update the student recognition data.

After selecting the data update option, verification by the instructor is required. In order to do this, the teacher's card should be approached (bounced), verify that the person on the ID card is the person shown in the photo and select the "y" (yes) button if the data is correct or "n" if the data is not correct. After selecting the " yes " option, the student does not have to re-register the ID card in order to register attendance. Example data displayed on the console:

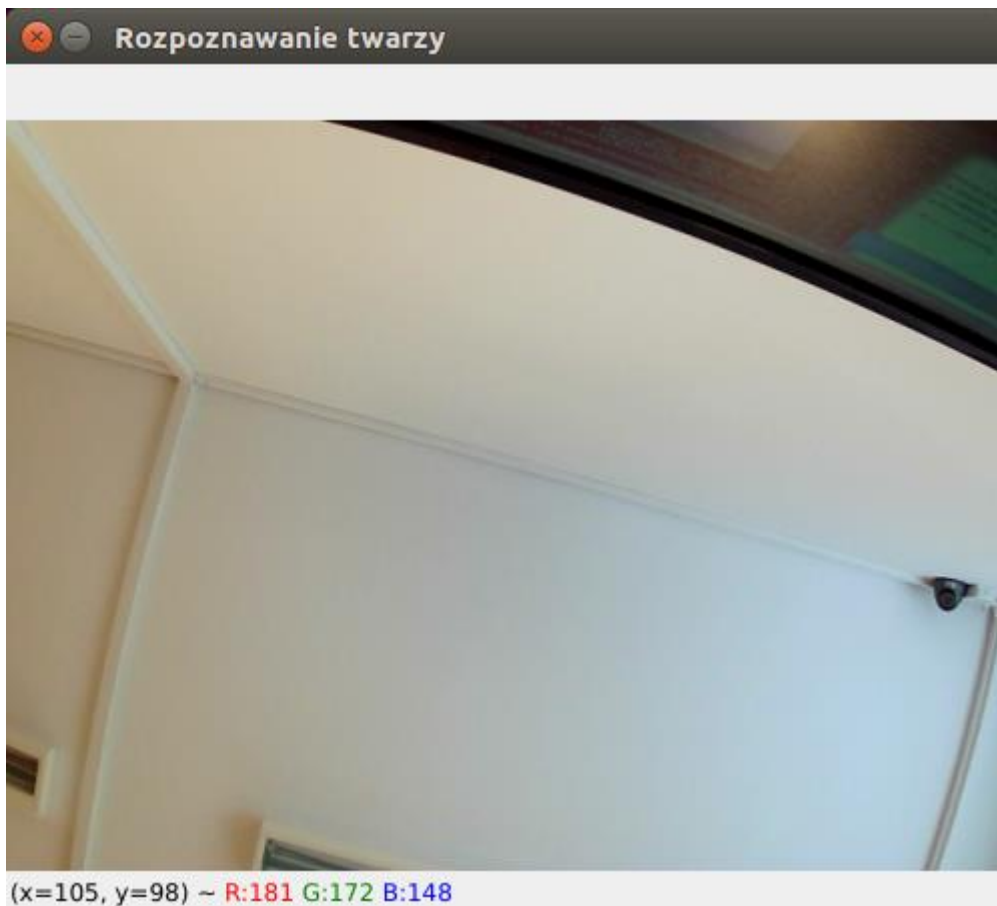
```
upsquared@upsquared-UP-APL01: ~/facial_recognition
pid, sts = os.waitpid(self.pid, flag)
KeyboardInterrupt
^^^Killed
upsquared@upsquared-UP-APL01:~/facial_recognition$ ./reader.o | python client.py
EDGE1
['dataset/222222/222222.pickle', 'dataset/111111/111111.pickle', 'dataset/265114
8469/2651148469.pickle']
/home/upsquared/facial_recognition/facial_recognition/dataset
connecting to broker 192.168.0.118
subscribing
received
found header
received
found data bytes= 1217
received
found header
Hash match - file valid 495300a25a7dc20d2e14a0bc84260921
complete file received
/home/upsquared/facial_recognition/dataset/2651148469/2651148469.pickle
/home/upsquared/facial_recognition/facial_recognition/dataset
connecting to broker 192.168.0.118
subscribing
2920761525
Prosze zbliżyc karte jeszcze raz w ciagu 10 sekund by uruchomic system obecności
na 15 minut
CARDREMOVED
2920761525
Prosze zbliżyc karte jeszcze raz w ciagu 10 sekund by uruchomic system obecności
na 15 minut
Uruchomiono system obecności na 15 minut
CARDREMOVED
2651148469
2651148469
/home/upsquared/facial_recognition/dataset/2651148469/2651148469.attpkl
Nie znaleziono
NIEROZPOZNANO
Nacisnij r by powtorzyc pomiar lub s by zaktualizowac rozpoznawanie twarzy
CARDREMOVED
1856422986
NIEROZPOZNANO
Nacisnij r by powtorzyc pomiar lub s by zaktualizowac rozpoznawanie twarzy
2651148469
NIEROZPOZNANO
Nacisnij r by powtorzyc pomiar lub s by zaktualizowac rozpoznawanie twarzy
NIEROZPOZNANO
Nacisnij r by powtorzyc pomiar lub s by zaktualizowac rozpoznawanie twarzy
W celu potwierdzenia tożsamości wezwij prowadzacego
CARDREMOVED
2920761525
y by potwierdzic tożsamość, n by odrzucic
CARDREMOVED
Potwierdzono tożsamość
/home/upsquared/facial_recognition/dataset/1856422986/1856422986.attpkl
Nie znaleziono
connecting to broker 192.168.0.118
subscribing
publishing
bytearray(b'header,,1856422986.jpg,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
```

```
upsquared@upsquared-UP-APL01: ~/facial_recognition
CARDREMOVED
2920761525
y by potwierdzic tozsamosc, n by odrzucic
CARDREMOVED
Potwierdzono tozsamosc
/home/upsquared/facial_recognition/dataset/1856422986/1856422986.attpkl
Nie znaleziono
connecting to broker 192.168.0.118
subscribing
publishing
bytearray(b'header,,1856422986.jpg,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,')
match mid 2
match mid 3
bytearray(b'end,,1856422986.jpg,,e9f7ad17977bfc2e3d734c498c7b2018,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,')
File copied OK -valid hash e9f7ad17977bfc2e3d734c498c7b2018
match mid 4
took 1.5066165924072266
received
found header
received
found data bytes= 1217
received
found header
Hash match - file valid 48385f2d95cf84a6261f081114250d6c
complete file received
/home/upsquared/facial_recognition/dataset/1856422986/1856422986.pickle
/home/upsquared/facial_recognition/facial_recognition/dataset
connecting to broker 192.168.0.118
subscribing
1856422986
CARDREMOVED
1856422986
1856422986
/home/upsquared/facial_recognition/dataset/1856422986/1856422986.attpkl
Znaleziono
[[['Monday', '8:00', '0'], ['Monday', '9:45', '0'], ['Monday', '11:45', '0'], ['Monday', '13:30', '0'], ['Monday', '15:10', '0'], ['Monday', '16:50', '0'], ['Monday', '18:30', '0'], ['Monday', '20:10', '0']], [['Tuesday', '8:00', '0'], ['Tuesday', '9:45', '0'], ['Tuesday', '11:45', '0'], ['Tuesday', '13:30', '1'], ['Tuesday', '15:10', '0'], ['Tuesday', '16:50', '0'], ['Tuesday', '18:30', '0'], ['Tuesday', '20:10', '0']], [['Wednesday', '8:00', '0'], ['Wednesday', '9:45', '0'], ['Wednesday', '11:45', '0'], ['Wednesday', '13:30', '0'], ['Wednesday', '15:10', '0'], ['Wednesday', '16:50', '0'], ['Wednesday', '18:30', '0'], ['Wednesday', '20:10', '0']], [['Thursday', '8:00', '0'], ['Thursday', '9:45', '0'], ['Thursday', '11:45', '0'], ['Thursday', '13:30', '0'], ['Thursday', '15:10', '0'], ['Thursday', '16:50', '0'], ['Thursday', '18:30', '0'], ['Thursday', '20:10', '0']], [['Friday', '8:00', '0'], ['Friday', '9:45', '0'], ['Friday', '11:45', '0'], ['Friday', '13:30', '0'], ['Friday', '15:10', '0'], ['Friday', '16:50', '0'], ['Friday', '18:30', '0'], ['Friday', '20:10', '0']], [['Saturday', '8:00', '0'], ['Saturday', '9:45', '0'], ['Saturday', '11:45', '0'], ['Saturday', '13:30', '0'], ['Saturday', '15:10', '0'], ['Saturday', '16:50', '0'], ['Saturday', '18:30', '0'], ['Saturday', '20:10', '0']], [['Sunday', '8:00', '0'], ['Sunday', '9:45', '0'], ['Sunday', '11:45', '0'], ['Sunday', '13:30', '0'], ['Sunday', '15:10', '0'], ['Sunday', '16:50', '0'], ['Sunday', '18:30', '0'], ['Sunday', '20:10', '0']]]
2651148469
1856422986
```

```
upsquared@upsquared-UP-APL01: ~/facial_recognition
1856422986
/home/upsquared/facial_recognition/dataset/1856422986/1856422986.attpkl
Znalezione
[[['Monday', '8:00', '0'], ['Monday', '9:45', '0'], ['Monday', '11:45', '0'], ['Monday', '13:30', '0'], ['Monday', '15:10', '0'], ['Monday', '16:50', '0'], ['Monday', '18:30', '0'], ['Monday', '20:10', '0']], [['Tuesday', '8:00', '0'], ['Tuesday', '9:45', '0'], ['Tuesday', '11:45', '0'], ['Tuesday', '13:30', '1'], ['Tuesday', '15:10', '0'], ['Tuesday', '16:50', '0'], ['Tuesday', '18:30', '0'], ['Tuesday', '20:10', '0']], [['Wednesday', '8:00', '0'], ['Wednesday', '9:45', '0'], ['Wednesday', '11:45', '0'], ['Wednesday', '13:30', '0'], ['Wednesday', '15:10', '0'], ['Wednesday', '16:50', '0'], ['Wednesday', '18:30', '0'], ['Wednesday', '20:10', '0']], [['Thursday', '8:00', '0'], ['Thursday', '9:45', '0'], ['Thursday', '11:45', '0'], ['Thursday', '13:30', '0'], ['Thursday', '15:10', '0'], ['Thursday', '16:50', '0'], ['Thursday', '18:30', '0'], ['Thursday', '20:10', '0']], [['Friday', '8:00', '0'], ['Friday', '9:45', '0'], ['Friday', '11:45', '0'], ['Friday', '13:30', '0'], ['Friday', '15:10', '0'], ['Friday', '16:50', '0'], ['Friday', '18:30', '0'], ['Friday', '20:10', '0']], [['Saturday', '8:00', '0'], ['Saturday', '9:45', '0'], ['Saturday', '11:45', '0'], ['Saturday', '13:30', '0'], ['Saturday', '15:10', '0'], ['Saturday', '16:50', '0'], ['Saturday', '18:30', '0'], ['Saturday', '20:10', '0']], [['Sunday', '8:00', '0'], ['Sunday', '9:45', '0'], ['Sunday', '11:45', '0'], ['Sunday', '13:30', '0'], ['Sunday', '15:10', '0'], ['Sunday', '16:50', '0'], ['Sunday', '18:30', '0'], ['Sunday', '20:10', '0']]]
2651148469
1856422986
NIEROZPOZNANO
Nacisnij r by powtorzyc pomiar lub s by zaktualizowac rozpoznawanie twarzy
1856422986
2651148469
1856422986
DR
CARDREMOVED
connecting to broker 192.168.0.118
subscribing
publishing
bytearray(b'header',dataset/222222/222222.attpkl,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
match mid 2
match mid 3
bytearray(b'end',dataset/222222/222222.attpkl,,fd3b85e994a53c049ab8c724cb02e82b,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
File copied OK -valid hash fd3b85e994a53c049ab8c724cb02e82b
match mid 4
took 1.5041241645812988
connecting to broker 192.168.0.118
subscribing
publishing
bytearray(b'header',dataset/111111/111111.attpkl,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
match mid 2
match mid 3
bytearray(b'end',dataset/111111/111111.attpkl,,fd3b85e994a53c049ab8c724cb02e82b,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
File copied OK -valid hash fd3b85e994a53c049ab8c724cb02e82b
match mid 4
took 1.503870964050293
connecting to broker 192.168.0.118
```

```
upsquared@upsquared-UP-APL01: ~/facial_recognition
CARDREMOVED
connecting to broker 192.168.0.118
subscribing
publishing
bytearray(b'header',,dataset/222222/222222.attpkl,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
match mid 2
match mid 3
bytearray(b'end',,dataset/222222/222222.attpkl,,fd3b85e994a53c049ab8c724cb02e82b,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
File copied OK -valid hash fd3b85e994a53c049ab8c724cb02e82b
match mid 4
took 1.5041241645812988
connecting to broker 192.168.0.118
subscribing
publishing
bytearray(b'header',,dataset/111111/111111.attpkl,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
match mid 2
match mid 3
bytearray(b'end',,dataset/111111/111111.attpkl,,fd3b85e994a53c049ab8c724cb02e82b,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
File copied OK -valid hash fd3b85e994a53c049ab8c724cb02e82b
match mid 4
took 1.503870964050293
connecting to broker 192.168.0.118
subscribing
publishing
bytearray(b'header',,dataset/2651148469/2651148469.attpkl,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
match mid 2
match mid 3
bytearray(b'end',,dataset/2651148469/2651148469.attpkl,,f4ebfb6aa6e36803dfc7b6b7e
08740f4,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
File copied OK -valid hash f4ebfb6aa6e36803dfc7b6b7e08740f4
match mid 4
took 1.5039336681365967
connecting to broker 192.168.0.118
subscribing
publishing
bytearray(b'header',,dataset/1856422986/1856422986.attpkl,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
match mid 2
match mid 3
bytearray(b'end',,dataset/1856422986/1856422986.attpkl,,f4ebfb6aa6e36803dfc7b6b7e
08740f4,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
File copied OK -valid hash f4ebfb6aa6e36803dfc7b6b7e08740f4
match mid 4
took 1.5036962032318115
1856422986
CARDREMOVED
1856422986
```


Example view from the camera:



5. Stopping the server and client

To stop the server or client, give KeyboardInterrupt (Ctrl+C) input to any application. Applications will first close subprocesses and then exit.

6. Input data

Face recognition photos should have a .jpg extension. The size of the photos is irrelevant if the face is recognizable "with the naked eye". Even for the size of 200x200px, the effectiveness of the model is sufficient to recognize its owner with 90% efficiency.

7. Conclusions and additional information

1. The server was tested in continuous operation for ~24 hours, no situation that would interfere with its operation was detected. The client was tested at work for ~6h, conclusions as above .
2. To read a single student's presence, run the script `presence.py student_index`
3. System resource usage on the client (UP2 Edge AI device) is as follows:

Enabled client without facial recognition enabled:

```

upsquared@upsquared-UP-APL01: ~
1 [|||||100.0%] Tasks: 126, 443 thr; 5 running
2 [|||||13.0%] Load average: 0.60 1.22 1.33
3 [|||||100.0%] Uptime: 00:43:45
4 [|||||23.4%]
Mem [|||||1.57G/7.64G]
Swp [|||||0K/4.39G]

  PID USER      PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
10045 upsquared  20   0 1234M 180M 13348 R 123. 2.3  0:07.79 python client.py
10043 upsquared  20   0  968M 170M  6216 R 102. 2.2  0:07.05 python client.py
10049 upsquared  20   0 1234M 180M 13348 S 15.4 2.3  0:00.99 python client.py
  904 root       20   0  269M 6508  5660 S  8.7 0.1  0:38.84 /usr/lib/accounts
 3600 upsquared  20   0 1115M 99084 53824 S  3.4 1.2  0:52.88 compiz
10047 upsquared  20   0 1234M 180M 13348 S  2.7 2.3  0:00.16 python client.py
10048 upsquared  20   0 1234M 180M 13348 S  2.7 2.3  0:00.16 python client.py
 9401 root       20   0 26136 3936  3176 R  2.7 0.0  0:08.67 htop
10046 upsquared  20   0 1234M 180M 13348 S  2.0 2.3  0:00.17 python client.py
 3682 upsquared  20   0  816M 45480 29132 S  1.3 0.6  0:43.05 /opt/intel/Arduin
 1270 root       20   0  437M 90792 57436 S  1.3 1.1  0:35.73 /usr/lib/xorg/Xor
 3768 upsquared  20   0  816M 45480 29132 R  0.7 0.6  0:05.40 /opt/intel/Arduin
 8446 upsquared  20   0 113G 91852 57632 S  0.7 1.1  0:06.75 unity-control-cen
 5080 upsquared  20   0  646M 36436 27684 S  0.7 0.5  0:07.99 /usr/lib/gnome-te
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice + F9Kill F10Quit

```

Enabled client with facial recognition enabled:

```

upsquared@upsquared-UP-APL01: ~
1 [|||||100.0%] Tasks: 126, 447 thr; 3 running
2 [|||||27.0%] Load average: 1.87 1.47 1.42
3 [|||||49.7%] Uptime: 00:44:13
4 [|||||78.8%]
Mem [|||||1.59G/7.64G]
Swp [|||||0K/4.39G]

  PID USER      PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
10045 upsquared  20   0 1373M 214M 36576 R 123. 2.7  0:41.74 python client.py
10043 upsquared  20   0  968M 170M  6216 R  98.7 2.2  0:34.53 python client.py
 3600 upsquared  20   0 1115M 99128 53868 S 10.3 1.2  0:54.66 compiz
10046 upsquared  20   0 1373M 214M 36576 S  8.4 2.7  0:01.69 python client.py
10047 upsquared  20   0 1373M 214M 36576 S  8.4 2.7  0:01.67 python client.py
10048 upsquared  20   0 1373M 214M 36576 S  7.7 2.7  0:01.60 python client.py
 1270 root       20   0  439M 91856 58500 S  3.9 1.1  0:36.79 /usr/lib/xorg/Xor
10049 upsquared  20   0 1373M 214M 36576 S  3.2 2.7  0:03.56 python client.py
 6095 upsquared  20   0 2988M 316M 169M S  2.6 4.0  0:41.31 /usr/lib/firefox/
 3682 upsquared  20   0  816M 45480 29132 S  1.9 0.6  0:43.46 /opt/intel/Arduin
 9401 root       20   0 26136 3936  3176 R  1.3 0.0  0:09.17 htop
 5080 upsquared  20   0  646M 36436 27684 S  1.3 0.5  0:08.19 /usr/lib/gnome-te
 4305 upsquared  20   0  816M 45480 29132 S  1.3 0.6  0:03.59 /opt/intel/Arduin
 3723 upsquared  20   0  816M 45480 29132 S  1.3 0.6  0:05.68 /opt/intel/Arduin
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice - F9Kill F10Quit

```

4. Due to the current file transfer protocol, the transfer time for facial recognition pickles is ~1 second. If the system is run with a large number of students, it may take too long to send all data to the client (30-60 minutes to send all data). The proposed solution would be to pack all data and send a single package to customers.
5. Another limitation of the current system is the fact that there should be only one person in the frame
6. A possible development of the system is the use of a touch panel for display instead of a screen + keyboard, programming a graphical interface with buttons would be required.
7. Communication between processes takes place via a queue system (Queue)
8. Optimization of resource usage can be done by removing loops listening to mqtt channels and replacing them with an appropriate algorithm of waiting for data