

# MULTIMEDIA SOFTWARE ENGINEERING

MS2: Term Project

Gerasimos Palaiopanos

Professor S.K. Chang

Link of demo, 1st version:

[https://drive.google.com/file/d/169zkSc5dp9MMMtlcMUb0Dcr\\_1bVDoqVZ/view?usp=sharing](https://drive.google.com/file/d/169zkSc5dp9MMMtlcMUb0Dcr_1bVDoqVZ/view?usp=sharing)

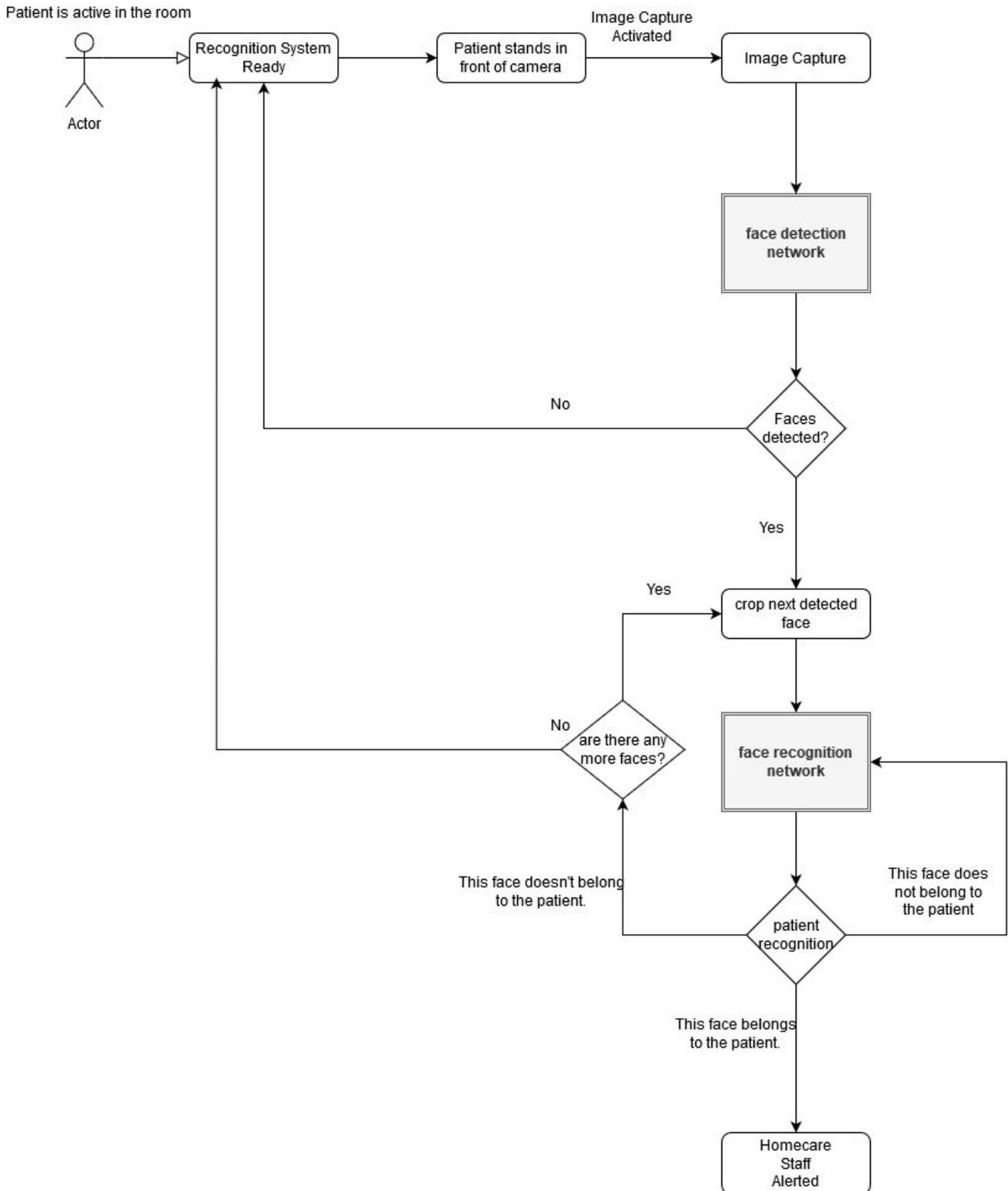
The following scenario is implemented:

- ❑ The developed system is in the 'Recognition System Ready' state and accepts input from a monitoring device ('web camera' located in a steady, fixed position in the patient's room).
- ❑ An actor ('patient') is located in his/her room and whenever he/she needs help, they need to address to the camera (Capture device).
- ❑ The capture device is constantly taking photos ('frames').
- ❑ When the patient sits in front of the camera (i.e. places themselves properly in front of it), the face detection component gets activated. It processes the current frame and isolates the face of the foreground ('cropping' function of 'Deep Learning component 1').
  - If a face is detected, the system proceeds to the next step below (dispatches the frame to the next function: 'embedding' of the frame).
  - If no other face is detected, the system loops back to the start: 'Recognition System Ready' state.
- ❑ The frame belongs to a person and is sent to the next component: preparation of the frame for the face recognition. This preparatory component is called 'Deep Learning component 2' (not shown in the flowchart below).
- ❑ The processed frame is fed into the 'Face recognition' component (called here 'Deep Learning component 3'), as shown in the flowchart below. At this point the face is classified into one of the following 6 classes:
  - If it belongs to the patient (this person is named 'patient' in our implementation), it is classified as 'patient'. This is the positive classification part of this system. Then the software proceeds to the next step below of notifying the homecare cell staff (sends email to the staff).
  - If it belongs to any other face (in our implementation we used 5 different faces from celebrity people, i.e. Ben Affleck, Elton John, Jerry Seinfeld, Madonna, Mindy Kaling), it loops back to the waiting state ('Recognition system ready')
- ❑ The patient needs to make a move of their face and have a second picture taken by the camera. Until then, the system awaits.
  - ★ If the patient stands up and moves away of the camera, the system awaits for 1 minute. If patient does not return, the systems loops back to its initial state and the procedure restarts.

- ★ If someone else (other resident) takes the patient's place in front of the camera, the system recognizes this picture and awaits for the patient until that 1 minute is over. If the patient does not show up, the system restarts.
- ★ If the patient stands again in front of the camera, the system recognizes him/her by the new picture shot and proceeds (2nd notification).
- The final step is a component which emails properly the homecare cell staff (called 'Homecare Staff Alerted'). This allows for an action from their part (like 'call the patient at home', 'send a team at the patient's house', 'send an emergency ambulance at the patient's house').

In total we used 4 components: 3 Deep Learning ones and 1 for the remote notification of the Homecare cell.

The following flowchart was implemented in this vs1 version, where the patient is represented as an actor. The possibilities are presented in the boxes and according to the recorded input (images captured from the web camera) there are consequent actions taken by the system:



We provide screenshots of the above steps, accompanied by parts of the code.

There are 4 major components for this are distinguished into the following:

1. 'Capture component': Image capturing to create a dataset with the patient.

- Continuous capture

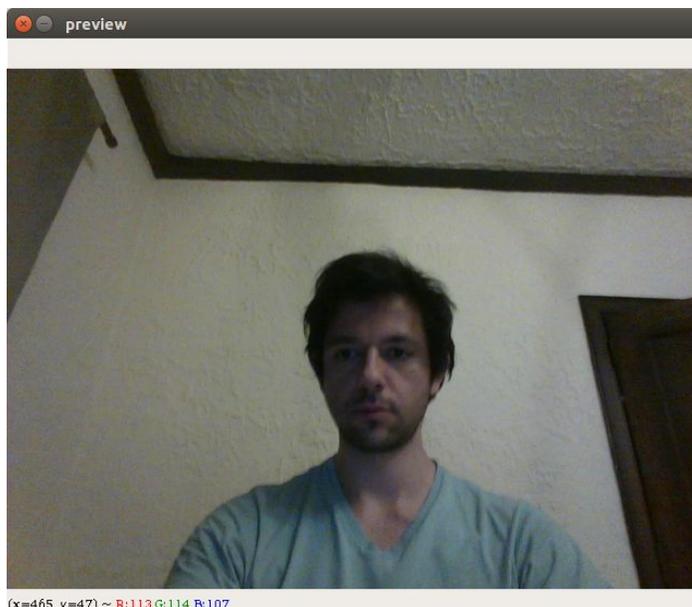
```
import cv2 # Open the device at the ID 0

cap = cv2.VideoCapture(0) #Check whether user selected camera is opened
successfully.

if not (cap.isOpened()):
    print('Could not open video device')
#To set the resolution
#cap.set(cv2.cv.CV_CAP_PROP_FRAME_WIDTH, 640)
#cap.set(cv2.cv.CV_CAP_PROP_FRAME_HEIGHT, 480)
while(True):
```

```
jeronimo@dynkin-laptop: ~/Deep_Learning/MLM_fr/model
>loaded 5 examples for class: jerry_seinfeld
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/model$
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/model$
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/model$ python camera_stream.py
```

The uncropped captured images look like this:



2. 'Deep Learning component 1': Face detection.

- Case of one face detected. The following function crops one face and appends it in the corresponding list.

```

from numpy import asarray
from mtcnn.mtcnn import MTCNN
# extract a single face from a given photograph
def extract_face(filename, required_size=(160, 160)):
    # load image from file
    image = Image.open(filename)
    # convert to RGB, if needed
    image = image.convert('RGB')
    # convert to array
    pixels = asarray(image)
    # create the detector, using default weights
    detector = MTCNN()
    results = detector.detect_faces(pixels)

```

The purpose is to perform 6-way classification in order to predict the identity of a given face:  
The 6th class is the patient.

This is the result of the face detection (the recorded faces are cropped and converted to the suitable format for the next phase, that of image recognition):

```

jeronimo@dynkin-laptop: ~/Deep_Learning/MLM_fr/model
(venv4) jeronimo@dynkin-laptop:~$ cd /home/jeronimo/Deep_Learning/MLM_fr/model
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/model$ python Crop_and_show_Vs1.py
Using TensorFlow backend.
2020-04-06 19:39:21.418809: W tensorflow/stream_executor/platform/default/dso_loader.cc:5
5] Could not load dynamic library 'libnvinfer.so.6'; dlerror: libnvinfer.so.6: cannot ope
n shared object file: No such file or directory
2020-04-06 19:39:21.418955: W tensorflow/stream_executor/platform/default/dso_loader.cc:5
5] Could not load dynamic library 'libnvinfer_plugin.so.6'; dlerror: libnvinfer_plugin.so
.6: cannot open shared object file: No such file or directory
2020-04-06 19:39:21.418976: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:30] Canno
t dlopen some TensorRT libraries. If you would like to use Nvidia GPU with TensorRT, plea
se make sure the missing libraries mentioned above are installed properly.
2020-04-06 19:39:22.405579: W tensorflow/stream_executor/platform/default/dso_loader.cc:5
5] Could not load dynamic library 'libcuda.so.1'; dlerror: libcuda.so.1: cannot open shar
ed object file: No such file or directory
2020-04-06 19:39:22.405621: E tensorflow/stream_executor/cuda/cuda_driver.cc:351] failed
call to cuInit: UNKNOWN ERROR (303)
2020-04-06 19:39:22.405653: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:156] ke
rnel driver does not appear to be running on this host (dynkin-laptop): /proc/driver/nvid
ia/version does not exist
2020-04-06 19:39:22.405930: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU
supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
2020-04-06 19:39:22.414376: I tensorflow/core/platform/profile_utils/cpu_utils.cc:94] CPU
Frequency: 2394385000 Hz
2020-04-06 19:39:22.414575: I tensorflow/compiler/xla/service/service.cc:168] XLA service
0x57e1c00 initialized for platform Host (this does not guarantee that XLA will be used).
Devices:
2020-04-06 19:39:22.414602: I tensorflow/compiler/xla/service/service.cc:176] StreamExe
cutor device (0): Host, Default Version
1 (160, 160, 3)
2 (160, 160, 3)
3 (160, 160, 3)
4 (160, 160, 3)
5 (160, 160, 3)
6 (160, 160, 3)
7 (160, 160, 3)
8 (160, 160, 3)
9 (160, 160, 3)
10 (160, 160, 3)
11 (160, 160, 3)

```

These are the cropped images.

They will be sent to the classification components.

The following pictures are the patient's cropped images from the 'Deep Learning component 1'.



3. 'Deep Learning component 2': The above images, along with any other face picture taken in the room, get processed using the following code (face embeddings) in order to be in form for SVM classification.

Here we present the dimensions of the cropped images ('embedded image' dimension):

```
jeronimo@dynkin-laptop: ~/Deep_Learning/MLM_fr/model
Loaded Model
(93, 128)
(25, 128)
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/model$
```

4. Face recognition through face classification (Deep Learning component 3, references [3], [4]).

- Case of patient's face recognized.

Then classification is performed. It is successful as the accuracy results show below. This is a separate running of this component, after the successful detection of a face in the previous steps. In this step the classification of the face is carried out into one the 6 possible classes.

```
jeronimo@dynkin-laptop: ~/Deep_Learning/MLM_fr/model
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/model$ python -W ignore 3.Face_classification_Vs
2.py
Dataset: train=168, test=48
Accuracy: train=100.000, test=100.000
[[14  0  0  0  0  0]
 [ 0 17  0  0  0  0]
 [ 0  0 21  0  0  0]
 [ 0  0  0 19  0  0]
 [ 0  0  0  0 22  0]
 [ 0  0  0  0  0 75]]
[[ 5  0  0  0  0  0]
 [ 0  5  0  0  0  0]
 [ 0  0  5  0  0  0]
 [ 0  0  0  5  0  0]
 [ 0  0  0  0  5  0]
 [ 0  0  0  0  0 23]]
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/model$
```

We also include a well known classification metric in machine learning problems: the confusion matrix, which is another way to show the correct classifications. Here all 6 classes were correctly classified.

The first confusion matrix is the training part and the second matrix is the testing part. The second measures the success of the developed system.

Here all pictures were successfully recognized (100% success) for the implemented 'Deep Learning component 3'.

All possible residents of the room:

Numbers corresponding to the detected faces:

[0] Resident 1: Ben Affleck

[1] Resident 2: Elton John

[2] Resident 3: Jerry Seinfeld

[3] Resident 4: Madonna

[4] Resident 5: Mindy Kaling

[5] Resident: Patient (this is the successful classification).

When the system recognizes the patient's face (number [5] in the output below), it proceeds to the Homecare notification step.

A real time running of the system, also presented in the live demo (see link on top of this report) is presented below. In this case there is **only one** image fed into the network, i.e. the

currently shot picture. There are not a number of pictures at the same time, as shown above, because this is a live recording of the face (one picture at a time):

```
ect file: No such file or directory
2020-04-07 05:41:47.676555: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:30] Cannot dlopen some TensorRT libraries. If you would like to use Nvidia GPU with TensorRT, please make sure the missing l
ibraries mentioned above are installed properly.
/home/jeronimo/venv4/lib/python3.6/site-packages/sklearn/externals/joblib/_init_.py:15: FutureWarning: sklearn.externals.joblib is deprecated in 0.21 and will be removed in 0.23. Please import this func
tionality directly from joblib, which can be installed with: pip install joblib. If this warning is raised when loading pickled models, you may need to re-serialize those models with scikit-learn 0.21+.
warnings.warn(msg, category=FutureWarning)
2020-04-07 05:41:52.577461: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcuda.so.1'; dLError: libcuda.so.1: cannot open shared object file: No such fi
le or directory
2020-04-07 05:41:52.577591: E tensorflow/stream_executor/cuda/cuda_driver.cc:351] failed call to cuInit: UNKNOWN ERROR (303)
2020-04-07 05:41:52.577626: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:156] kernel driver does not appear to be running on this host (dynkin-laptop): /proc/driver/nvidia/version does not exist
2020-04-07 05:41:52.578077: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
2020-04-07 05:41:52.595894: I tensorflow/core/platform/profile_utils/cpu_utils.cc:94] CPU Frequency: 2394385000 Hz
2020-04-07 05:41:52.596506: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x5d652e9 initialized for platform Host (this does not guarantee that XLA will be used). Devices:
2020-04-07 05:41:52.596883: I tensorflow/compiler/xla/service/service.cc:170] StreamExecutor device (0): Host, Default Version
/home/jeronimo/venv4/lib/python3.6/site-packages/keras/engine/saving.py:341: UserWarning: No training configuration found in save file: the model was *not* compiled. Compile it manually.
warnings.warn("No training configuration found in save file: ")
Loaded Model
The patient is not recognized yet and the system is retaking photos...
[4]
The patient is not recognized yet and the system is retaking photos...
[4]
The patient is not recognized yet and the system is retaking photos...
[3]
The patient is not recognized yet and the system is retaking photos...
[2]
The patient is not recognized yet and the system is retaking photos...
[0]
The patient is not recognized yet and the system is retaking photos...
[4]
The patient is not recognized yet and the system is retaking photos...
[0]
The patient is not recognized yet and the system is retaking photos...
[4]
The patient is not recognized yet and the system is retaking photos...
[4]
The patient is not recognized yet and the system is retaking photos...
[1]
Patient recognized once only, so system is waiting for a second recognition
/home/jeronimo/venv4/lib/python3.6/site-packages/keras/engine/saving.py:341: UserWarning: No training configuration found in save file: the model was *not* compiled. Compile it manually.
warnings.warn("No training configuration found in save file: ")
Loaded Model
The patient is not recognized yet and the system is retaking photos...
[2]
The patient is not recognized yet and the system is retaking photos...
[3]
The patient is not recognized yet and the system is retaking photos...
[1]
The patient is not recognized yet and the system is retaking photos...
[1]
The patient is not recognized yet and the system is retaking photos...
[1]
Patient recognized once only, so system is waiting for a second recognition
Patient recognized twice, so alert is emailed to Homecare now...
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/models$
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/models$
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/models$
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/models$
```

```
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/models$ python -W ignore 0.Demo_vs2.py
Using TensorFlow backend.
2020-04-07 05:45:11.066247: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libnvinfer.so.6'; dLError: libnvinfer.so.6: cannot open shared object file: No s
uch file or directory
2020-04-07 05:45:11.066433: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libnvinfer_plugin.so.6'; dLError: libnvinfer_plugin.so.6: cannot open shared obj
ect file: No such file or directory
2020-04-07 05:45:11.066457: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:30] Cannot dlopen some TensorRT libraries. If you would like to use Nvidia GPU with TensorRT, please make sure the missing l
ibraries mentioned above are installed properly.
2020-04-07 05:45:15.757238: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcuda.so.1'; dLError: libcuda.so.1: cannot open shared object file: No such fi
le or directory
2020-04-07 05:45:15.757312: E tensorflow/stream_executor/cuda/cuda_driver.cc:351] failed call to cuInit: UNKNOWN ERROR (303)
2020-04-07 05:45:15.757353: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:156] kernel driver does not appear to be running on this host (dynkin-laptop): /proc/driver/nvidia/version does not exist
2020-04-07 05:45:15.757777: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
2020-04-07 05:45:15.776166: I tensorflow/core/platform/profile_utils/cpu_utils.cc:94] CPU Frequency: 2394385000 Hz
2020-04-07 05:45:15.776790: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x5a39720 initialized for platform Host (this does not guarantee that XLA will be used). Devices:
2020-04-07 05:45:15.776961: I tensorflow/compiler/xla/service/service.cc:170] StreamExecutor device (0): Host, Default Version
Loaded Model
The patient is not recognized yet and the system is retaking photos...
[3]
The patient is not recognized yet and the system is retaking photos...
[3]
[5]
Patient recognized once only, so system is waiting for a second recognition
Loaded Model
The patient is not recognized yet and the system is retaking photos...
[4]
The patient is not recognized yet and the system is retaking photos...
[2]
The patient is not recognized yet and the system is retaking photos...
[3]
The patient is not recognized yet and the system is retaking photos...
[0]
The patient is not recognized yet and the system is retaking photos...
[2]
The patient is not recognized yet and the system is retaking photos...
[1]
The patient is not recognized yet and the system is retaking photos...
[1]
Patient recognized once only, so system is waiting for a second recognition
Patient recognized twice, so alert is emailed to Homecare now...
(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/models$
```

```
ernals.joblib is deprecated in 0.21 and will be removed in 0.23. Please import this functionality directly from joblib, which can be installed with: pip install joblib. If this warning is raised when loading pickled models, you may need to re-serialize those models with scikit-learn 0.21+.
  warnings.warn(msg, category=FutureWarning)
2020-04-07 05:34:10.615187: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcuda.so.1'; dLError: libcuda.so.1: cannot open shared object file: No such file or directory
2020-04-07 05:34:10.615231: E tensorflow/stream_executor/cuda/cuda_driver.cc:351] failed call to cuInit: UNKNOWN ERROR (303)
2020-04-07 05:34:10.615261: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:156] kernel driver does not appear to be running on this host (dynkin-laptop): /proc/driver/nvidia/version does not exist
2020-04-07 05:34:10.615635: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
2020-04-07 05:34:10.641989: I tensorflow/core/platform/profile_utils/cpu_utils.cc:94] CPU Frequency: 2394385000 Hz
2020-04-07 05:34:10.645730: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x668b0c0 initialized for platform Host (this does not guarantee that XLA will be used). Devices:
2020-04-07 05:34:10.645800: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor device (0): Host, Default Version
/home/jeronimo/venv4/lib/python3.6/site-packages/keras/engine/saving.py:341: UserWarning: No training configuration found in save file: the model was *not* compiled. Compile it manually.
  warnings.warn('No training configuration found in save file: '
Loaded Model
The patient is not recognized yet and the system is retaking photos...
[2]
The patient is not recognized yet and the system is retaking photos...
[0]
The patient is not recognized yet and the system is retaking photos...
[4]
The patient is not recognized yet and the system is retaking photos...
[0]
The patient is not recognized yet and the system is retaking photos...
[3]
[5]
Patient recognized once only, so system is waiting for a second recognition
/home/jeronimo/venv4/lib/python3.6/site-packages/keras/engine/saving.py:341: UserWarning: No training configuration found in save file: the model was *not* compiled. Compile it manually.
  warnings.warn('No training configuration found in save file: '
Loaded Model
[5]
Patient recognized once only, so system is waiting for a second recognition
Patient recognized twice, so alert is emailed to Homecare now...

(venv4) jeronimo@dynkin-laptop:~/Deep_Learning/MLM_fr/model$
```

- Case of irrelevant face recognized.

If the system does not recognize the patient's face either for the first or for the second time, and thus finds any other person's face (in our implementation that means any of other 5 residents of the room, assigned to numbers [0]-[4]), it awaits for the patient's face.

```
Loaded Model
The patient is not recognized yet and the system is retaking photos...
[2]
The patient is not recognized yet and the system is retaking photos...
[0]
The patient is not recognized yet and the system is retaking photos...
[4]
The patient is not recognized yet and the system is retaking photos...
[0]
The patient is not recognized yet and the system is retaking photos...
[3]
[5]
Patient recognized once only, so system is waiting for a second recognition
/home/jeronimo/venv4/lib/python3.6/site-packages/keras/engine/saving.py:341: UserWarning: No training configuration
```

In the next version of this system we will add a facial expression component (5th component).

References: [1] FaceNet: A Unified Embedding for Face Recognition and Clustering  
<https://arxiv.org/abs/1503.03832> .

[2] <https://github.com/davidsandberg/facenet>

[3] <https://github.com/iwantoxxoox/Keras-OpenFace>

[4] Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks,  
<https://arxiv.org/abs/1604.02878> .

[5] DATASET: <https://www.kaggle.com/dansbecker/5-celebrity-faces-dataset> .

[6] <https://github.com/davidsandberg/facenet/tree/master/src/align>

[7] Python api's:

<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.Normalizer.html> and  
<https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html> .