Princess Sumaya University for Technology
جامعة الأميرة سميّة للتكنولوجيا

King Hussein School Of Computing Sciences

# Robotics Programming Project

# Face Recognition Report


**Prepared by:**

Raghad Joudeh (20200777)

Sondos Ali (20200438)

**Supervised by:**

Dr. Remah Bani Younisse

# Table of Contents

# Introduction

The main objective of this project is to merge Artificial Intelligence techniques with the Robotics Programming tools.

We have seen wide applications of facial recognition throughout emerging technologies such as in houses, in companies and in modern phones, where people are being recognized and identified. Face recognition helps to secure systems during logins, helps in verifying a person's identity.

This project mainly focuses on how to capture a photo using Raspberry Pi 4 Camera Module, train the model on the taken images and try to identify the given image.

# Requirements

In order for this project to be successful, there are some steps you have to make sure are done, listing the steps bellow:

1. Get a Raspberry Pi 4
2. Raspberry Pi Camera Module
3. Get an SD Card
4. Download Raspberry Pi Installer
5. Power supply (for your Raspberry Pi)
6. Connect your Raspberry Pi with the same network as your device
7. Download puTTY
8. Connect your device to the Raspberry Pi through puTTY
9. Download VNC Viewer

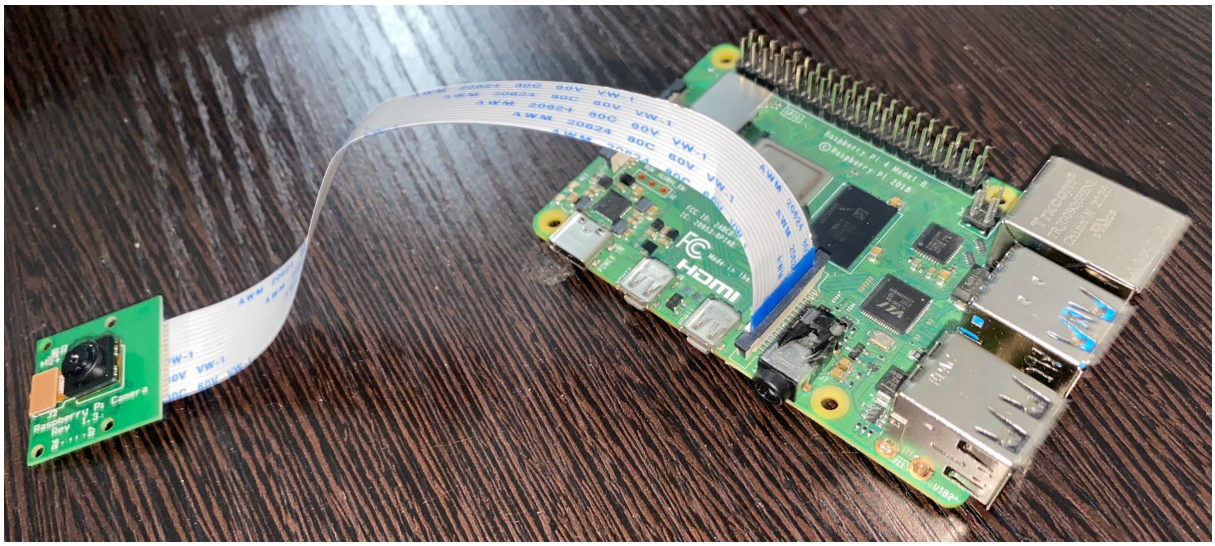
# How does the Raspberry Pi Facial Recognition project work?

For Raspberry Pi facial recognition, there are three packages to download to train our Raspberry Pi based on a set of images that we collect and provide as our dataset.

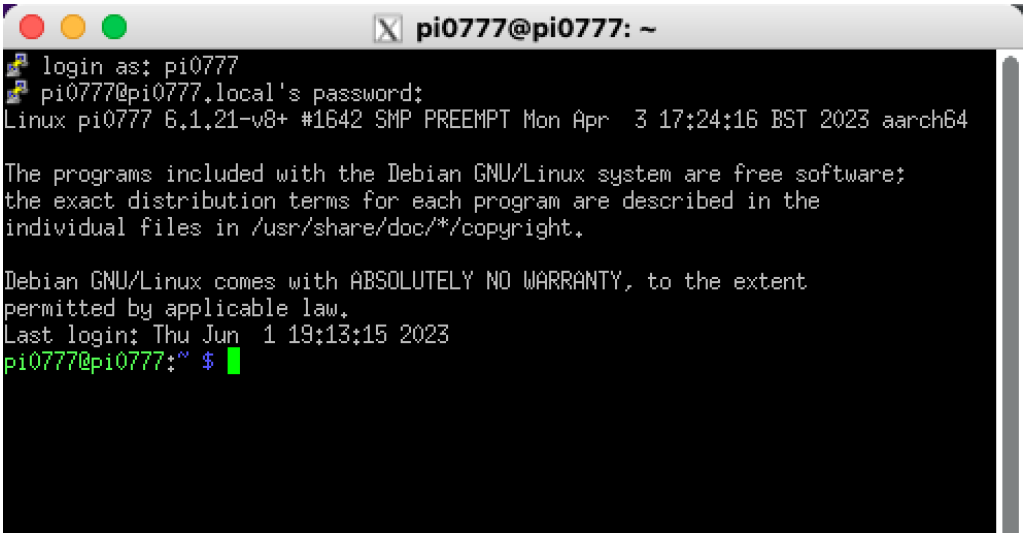
## Part 1: Install Dependencies for Raspberry Pi Facial Recognition

We will first install the three packages; OpenCV, face_recognition, imutils, and temporarily modify our swapfile to prepare our Raspberry Pi for machine learning and facial recognition. This step might take 1-2 hours to complete.

1. Insert the Raspberry Pi camera to your Raspberry Pi Board.



2. Boot your Raspberry Pi and open the Terminal




3. Install OpenCV by running the following commands in your Terminal.

```
pi0777@pi0777:~ $ sudo apt install cmake build-essential pkg-config git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
build-essential is already the newest version (12.9).
git is already the newest version (1:2.30.2-1+deb11u2).
pkg-config is already the newest version (0.29.2-1).
cmake is already the newest version (3.18.4-2+rpt1+rpi1+deb11u1).
The following package was automatically installed and is no longer required:
  libfuse2
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
pi0777@pi0777:~ $
```

Following some more command to complete this step:

- sudo apt install libjpeg-dev libtiff-dev libjasper-dev libpng-dev libwebp-dev libopenexr-dev
- 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
- sudo apt install libgtk-3-dev libqtgui4 libqtwebkit4 libqt4-test python3-pyqt5
- sudo apt install libatlas-base-dev liblapacke-dev gfortran
- sudo apt install libhdf5-dev libhdf5-103
- sudo apt install python3-dev python3-pip python3-numpy

4. Install face_recognition package

```
pi0777@pi0777:~ $ pip install face-recognition
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Requirement already satisfied: face-recognition in ./.local/lib/python3.9/site-p
ackages (1.3.0)
Requirement already satisfied: face-recognition-models>=0.3.0 in ./.local/lib/py
thon3.9/site-packages (from face-recognition) (0.3.0)
Requirement already satisfied: dlib>=19.7 in ./.local/lib/python3.9/site-package
s (from face-recognition) (19.24.1)
Requirement already satisfied: Click>=6.0 in /usr/lib/python3/dist-packages (fro
m face-recognition) (7.1.2)
Requirement already satisfied: Pillow in /usr/lib/python3/dist-packages (from fa
ce-recognition) (8.1.2)
Requirement already satisfied: numpy in /usr/lib/python3/dist-packages (from fac
e-recognition) (1.19.5)
pi0777@pi0777:~ $
```

5. Install imutils

```
e-recognition) (1.19.5)
pi0777@pi0777:~ $ pip install imutils
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Requirement already satisfied: imutils in ./.local/lib/python3.9/site-packages (
0.5.4)
pi0777@pi0777:~ $
```

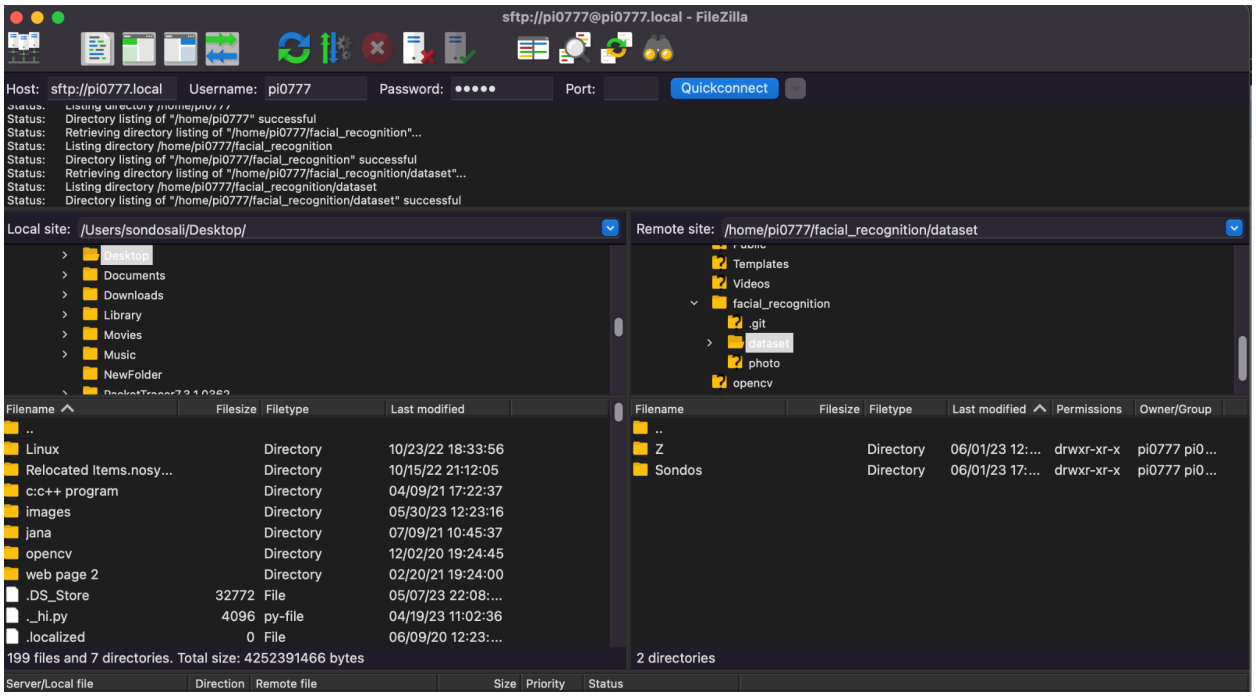
## Part 2: Train the Model for Raspberry Pi Facial Recognition

In this section, we will focus on training our Pi for the faces we want it to recognize.

1. Let's start by downloading the Python code for facial recognition.

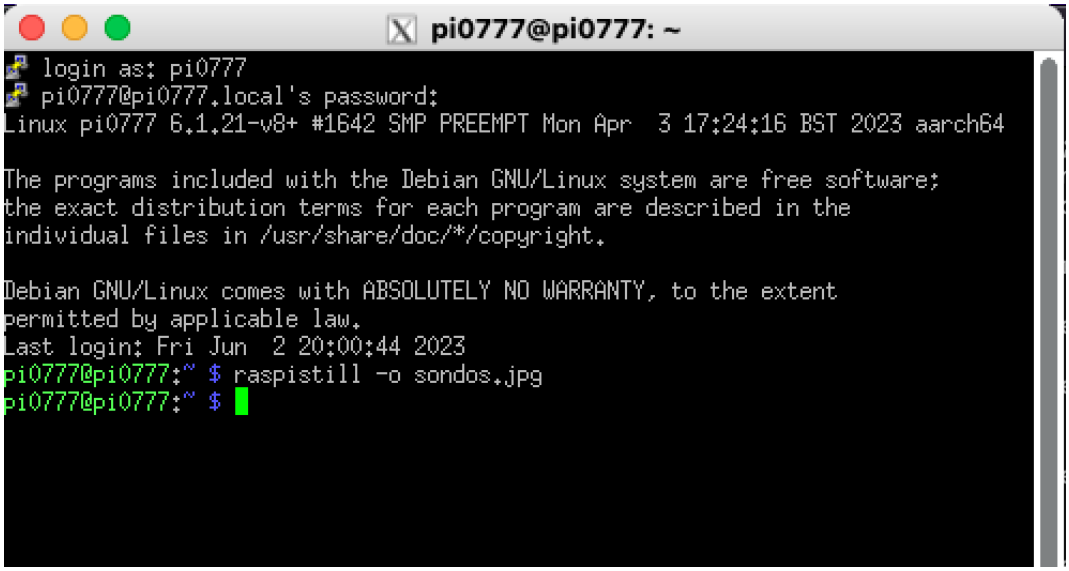
```
pi0777@pi0777: ~
pi0777@pi0777:~ $ git clone https://github.com/carolinedunn/facial_recognition
```

2. Open FileZilla to connect to your Raspberry Pi
   - Insert a new folder in the dataset folder found in the facial_recognition file. To upload your photos inside.

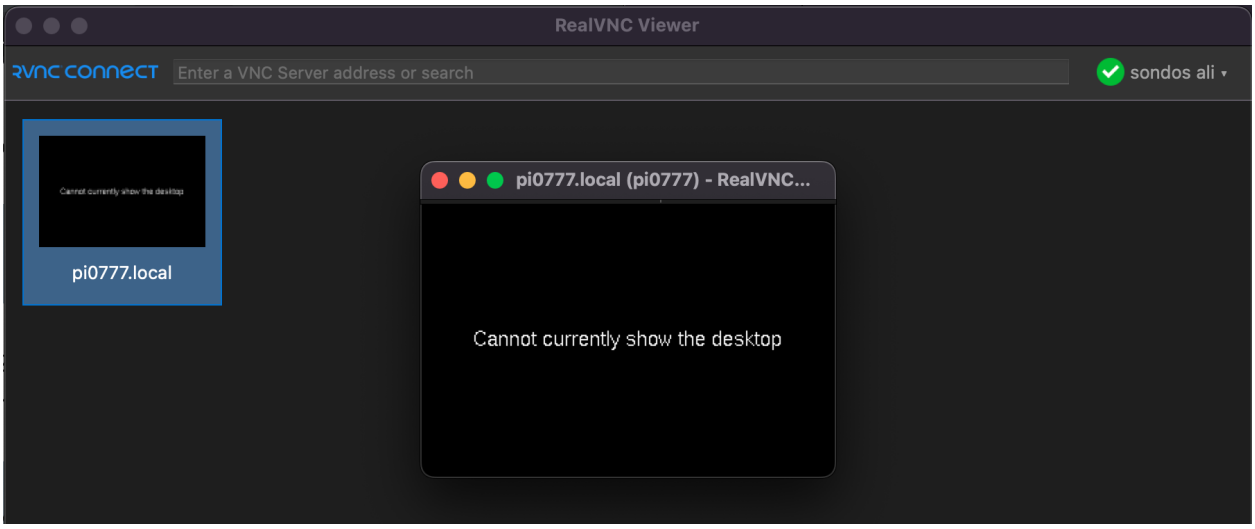


3. Capture images from the Raspberry Pi Camera Module

We tried running *python headshots_picam.py* but an error occurred, which says that cv2 library has an error running *imshow* function. As a consequence, the code we used to capture our images was *raspistill -o image.jpg,* we ran it more than once to capture more than one image.




The reason for the error occurred was because the desktop of the raspberry pi didn't show when we tried downloading VNC Viewer. The following appeared when trying to connect to the desktop of the raspberry pi.

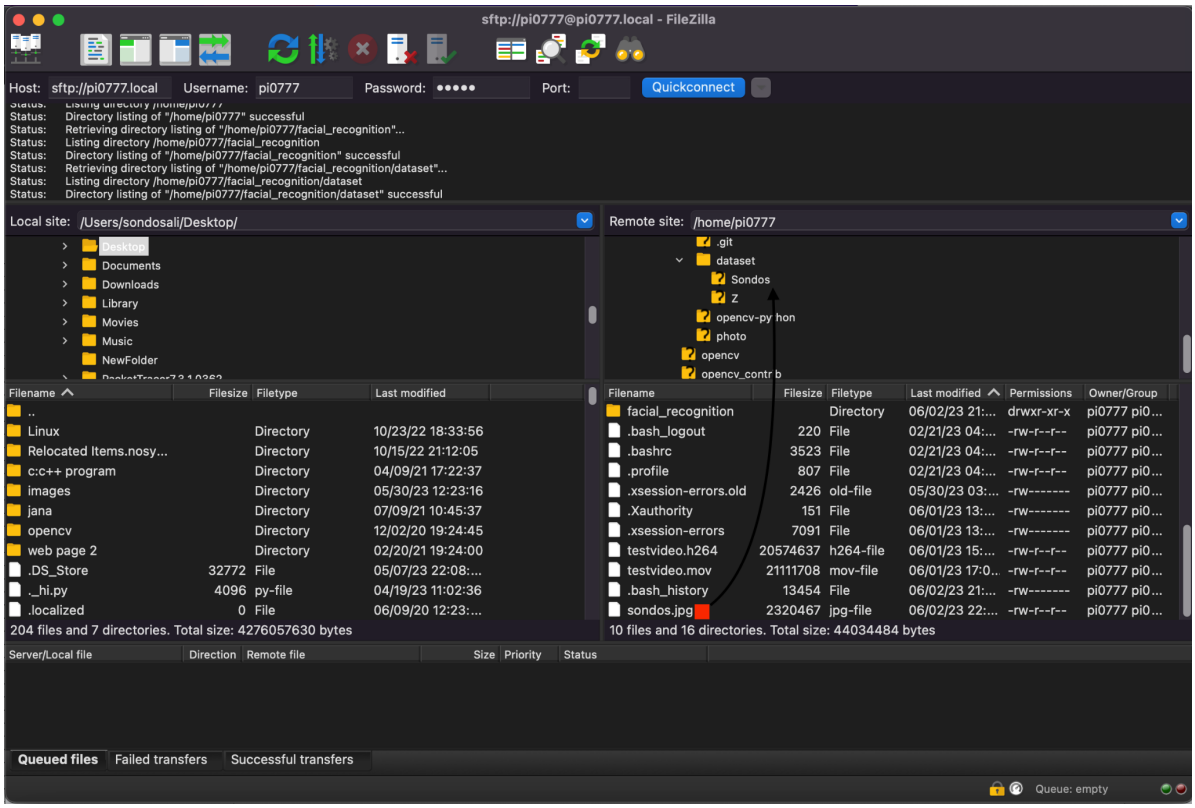


```
pi0777@pi0777:~ $ cd facial_recognition
pi0777@pi0777:~/facial_recognition $ python headshots_picam.py
Traceback (most recent call last):
  File "/home/pi0777/facial_recognition/headshots_picam.py", line 17, in <module
>
    cv2.imshow("Press Space to take a photo", image)
cv2.error: OpenCV(4.7.0) /tmp/pip-install-7e4481n9/opencv-python_6d04a616b62e470
7a61baac61cccb251/opencv/modules/highgui/src/window.cpp:1272: error: (-2:Unspeci
fied error) The function is not implemented. Rebuild the library with Windows, G
TK+ 2.x or Cocoa support. If you are on Ubuntu or Debian, install libgtk2.0-dev
and pkg-config, then re-run cmake or configure script in function 'cvShowImage'

pi0777@pi0777:~/facial_recognition $
```

The figure above shows the error.

4. Move the images from home directory to facial_recognition/datasets/YOUR_NAME directory using FileZilla



5. Train the model using the following command

```
pi0777@pi0777:~/facial_recognition $ python train_model.py
[INFO] start processing faces...
[INFO] processing image 1/3
[INFO] processing image 2/3
[INFO] processing image 3/3
[INFO] serializing encodings...
pi0777@pi0777:~/facial_recognition $
```

6. Test the model using the following command

While running the *python facial_req.py* command in the terminal, an error appeared similar to the one in Step 3. Which states that there is a problem identifying *cv2.imshow()* function due to the VNC not being able to connect to the Raspberry Pi desktop.

```
pi0777@pi0777:~/facial_recognition $ python facial_req.py
[INFO] loading encodings + face detector...
Traceback (most recent call last):
  File "/home/pi0777/facial_recognition/facial_req.py", line 90, in <module>
    cv2.imshow("Facial Recognition is Running", frame)
cv2.error: OpenCV(4.7.0) /tmp/pip-install-7e4481n9/opencv-python_6d04a616b62e470
7a61baac61cccb251/opencv/modules/highgui/src/window.cpp:1272: error: (-2:Unspeci
fied error) The function is not implemented. Rebuild the library with Windows, G
TK+ 2.x or Cocoa support. If you are on Ubuntu or Debian, install libgtk2.0-dev
and pkg-config, then re-run cmake or configure script in function 'cvShowImage'

pi0777@pi0777:~/facial_recognition $
```

From this command we expect a window to show which opens a camera, and the function will look for a face to detect and if the detected face was the same as the one the model trained on, then the name of the person will appear right above the person's face frame.

# Can Raspberry Pi Also Capture Videos?

Raspberry Pi Camera can definitely video record too. In order to perform this task, we used the following command to run.


As shown in the figure above, the video format is *.h264*. We used a converter to convert *.h246* to *.MP4* video.

# Conclusion

Although our project was not quite successful in showing if the face has been recognized, we have made sure everything was done correctly.

Majorly, in this project our focus was on learning how to combine Robotics Programming, where we use the hardware, and Artificial Intelligence, in training the model with the images pictured, to perform a commonly known topic, face recognition.

# References


- https://deepfakechallenge.com/gb/2021/03/26/8213/
- https://www.tomshardware.com/reviews/raspberry-pi-headless-setup-how-to,6028.html
- https://www.youtube.com/watch?v=T8T6S5eFpqE
- https://www.realvnc.com/en/blog/how-to-use-a-headless-raspberry-pi-with-vnc-connect/#:~:text=In%20your%20Raspberry%20Pi%20menu,and%20then%20reboot%20as%20prompted.
- https://bobbyhadz.com/blog/python-no-module-named-cv2
- https://anyconv.com/h264-to-mp4-converter/