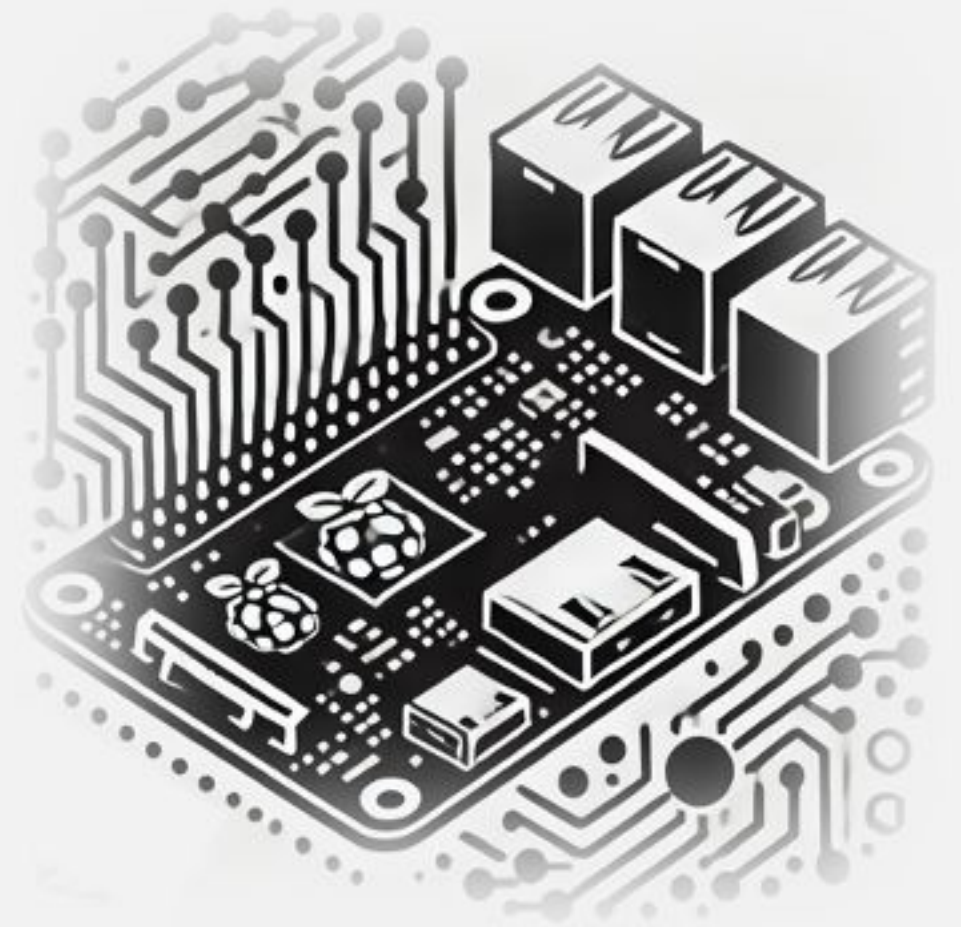


IESTI05 – Edge AI

Machine Learning System Engineering

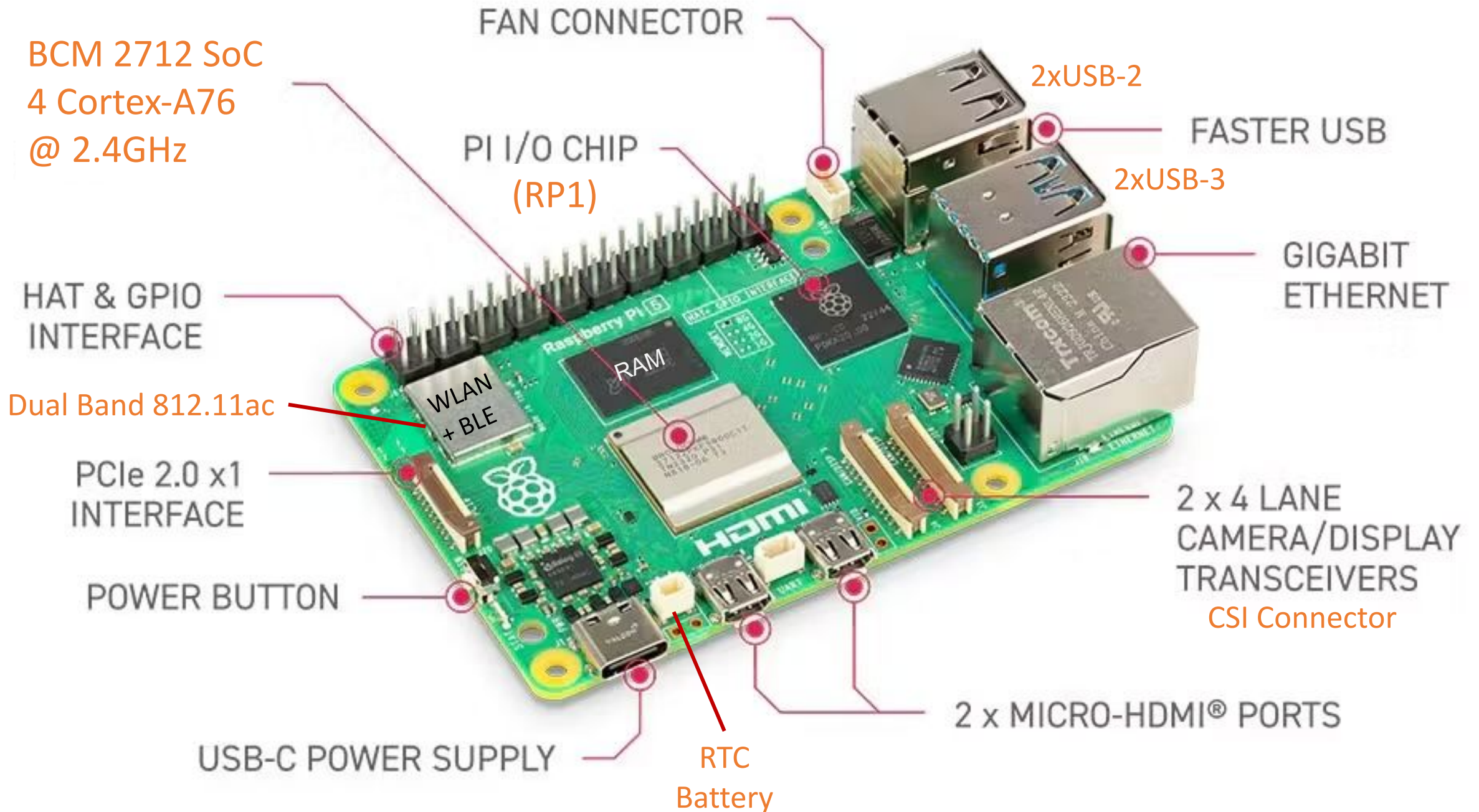
Raspberry

16.a Raspberry Pi 5 Setup

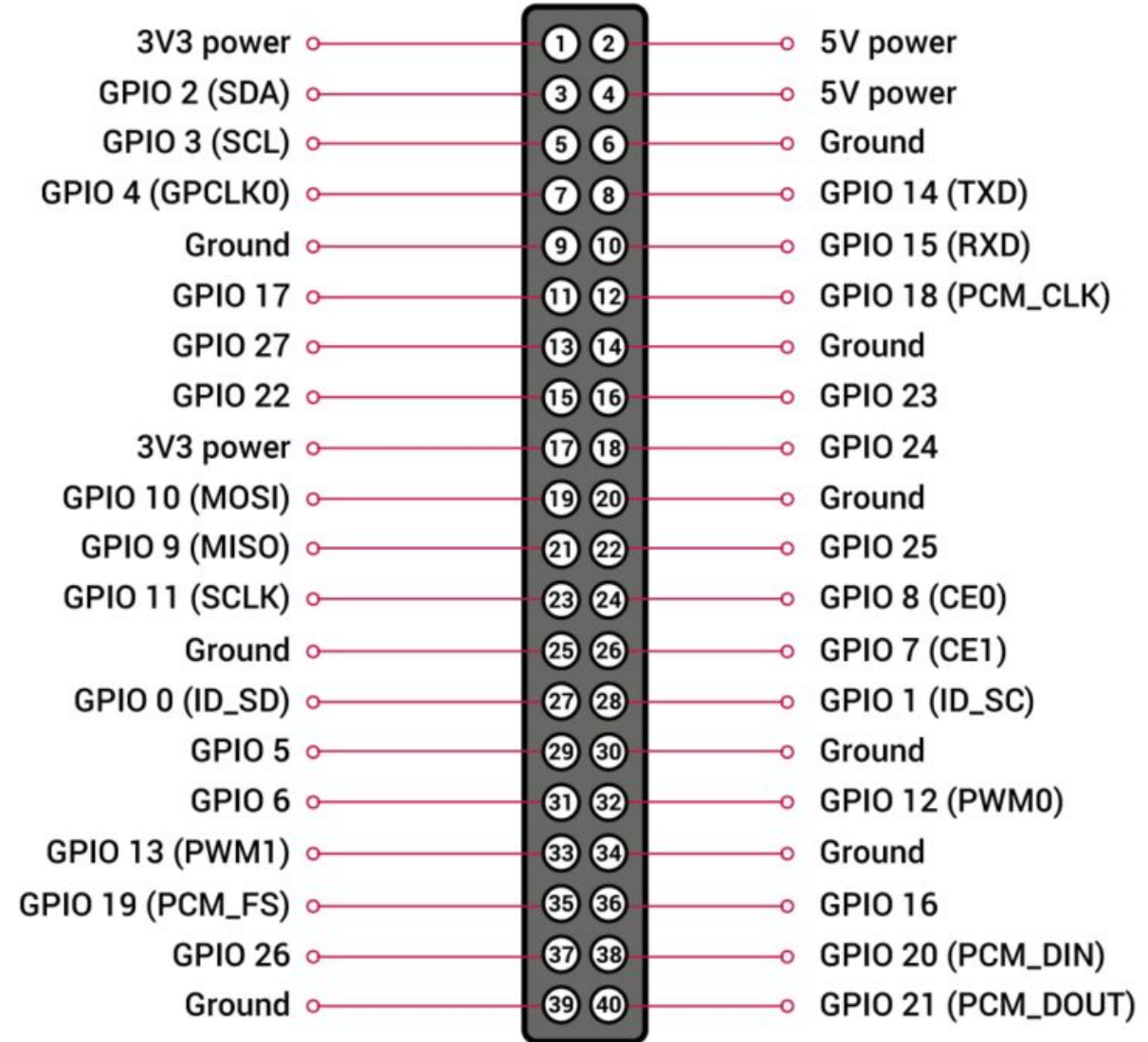
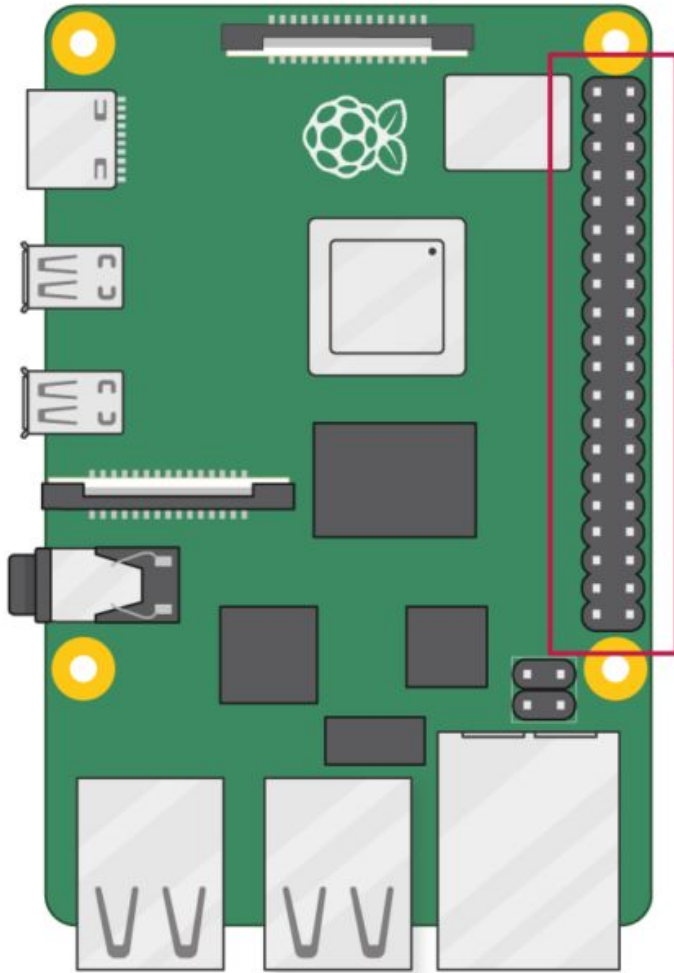


Raspberry Pi 5

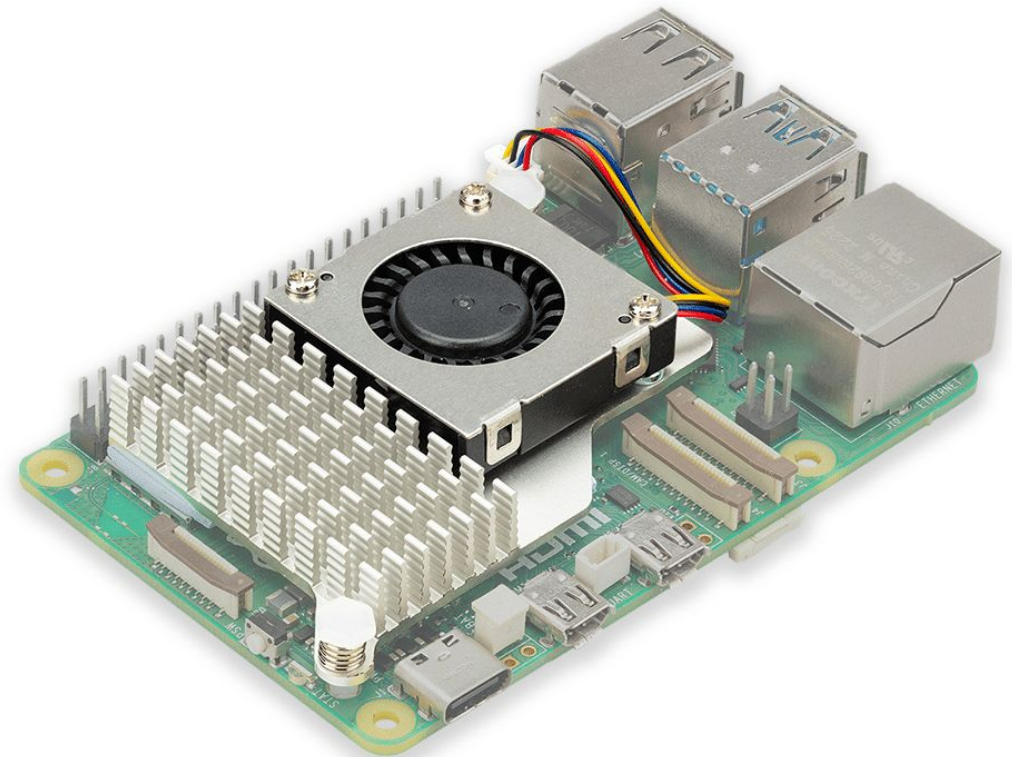
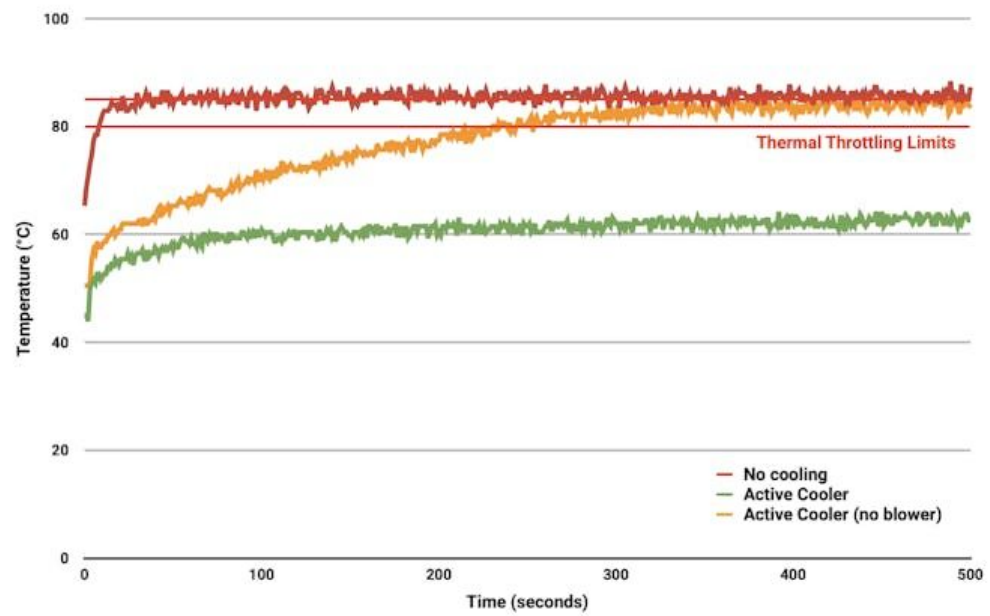
HW and Installation



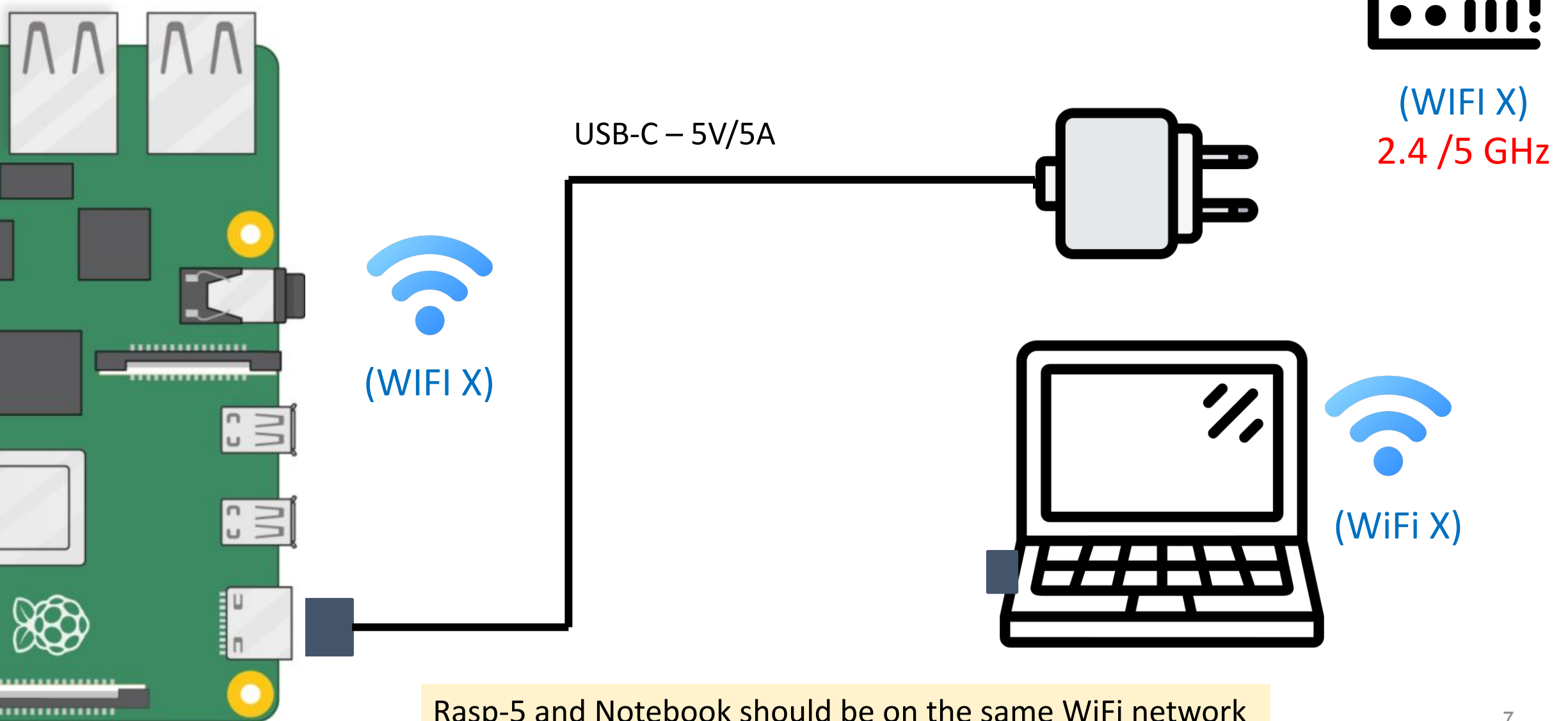
GPIO pinout





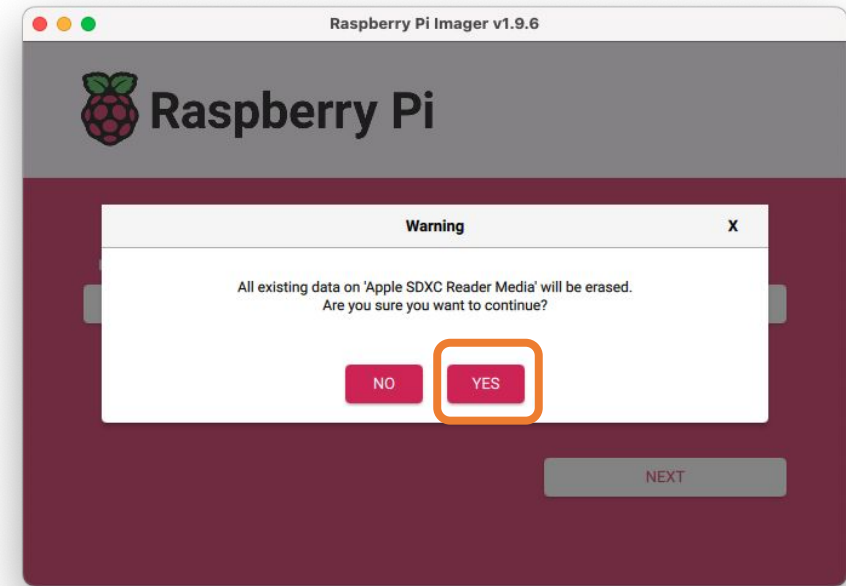
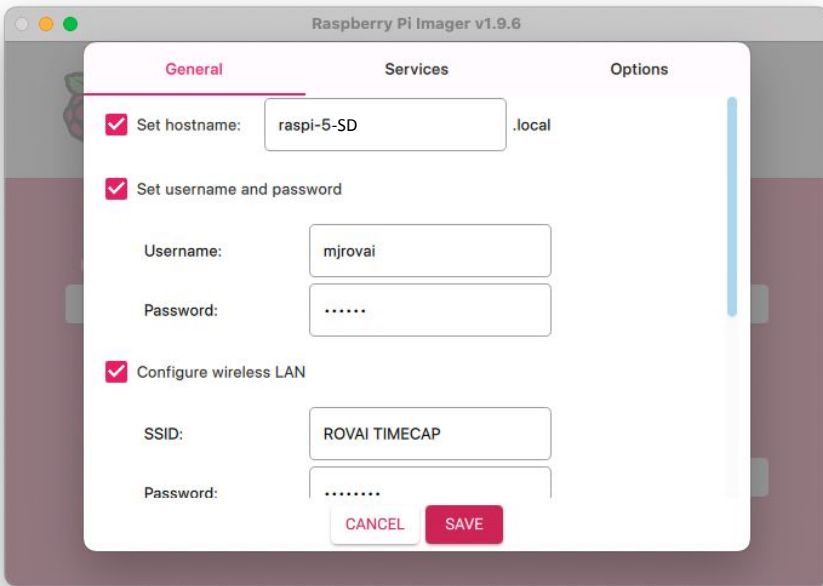
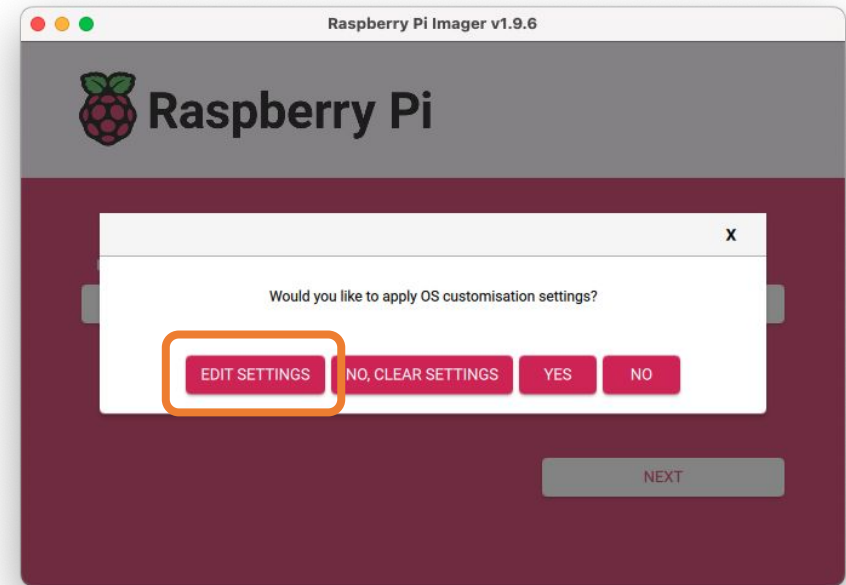
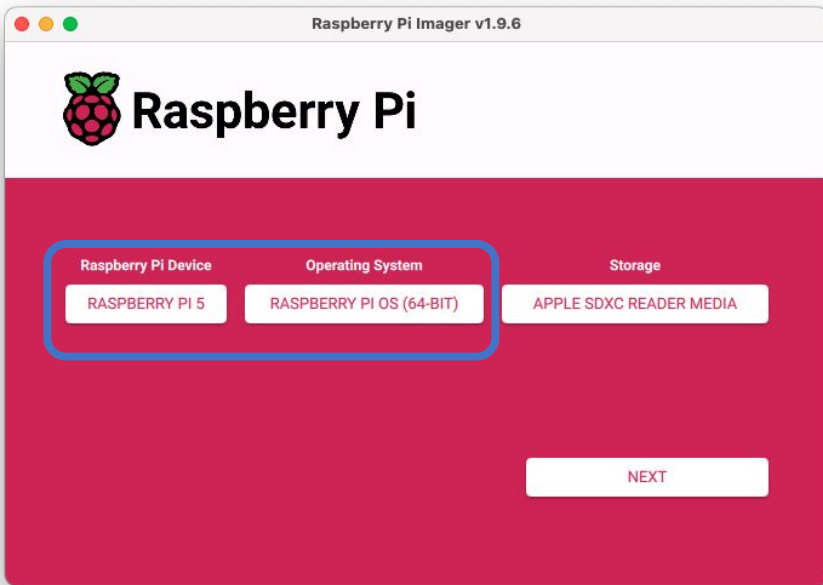


Power Supply and WIFI Network



Raspberry Pi 5

Setup



```
marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 94x24

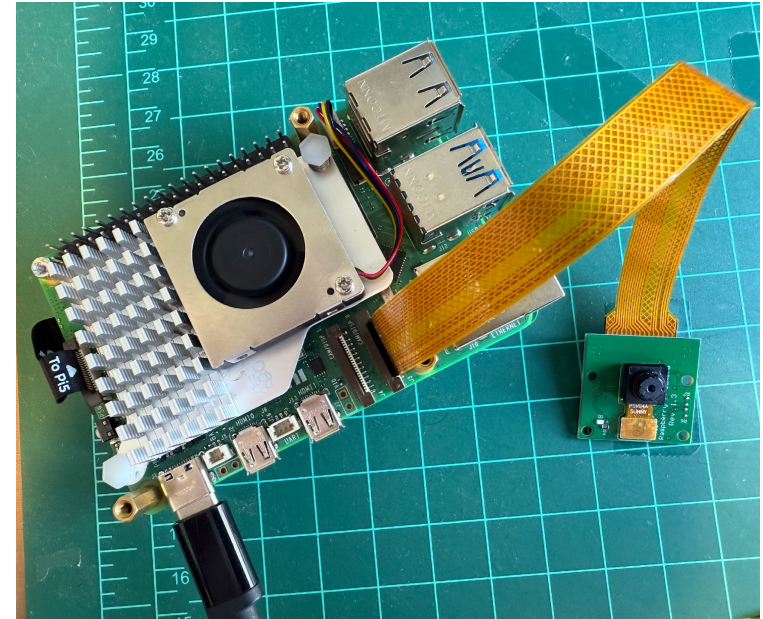
Last login: Mon Sep 15 12:06:35 on ttys000
(base) marcelo_rovai@Marcelos-MacBook-Pro ~ % ssh mjrovai@raspi-5-SD.local
The authenticity of host 'raspi-5-sd.local (192.168.4.209)' can't be established.
ED25519 key fingerprint is SHA256:ErOOSolR8uyzELk03vJQ6kQUBcjx0UrA0IW1H7GVA4Y.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'raspi-5-sd.local' (ED25519) to the list of known hosts.
mjrovai@raspi-5-sd.local's password:
Linux raspi-5-SD 6.12.25+rpt-rpi-2712 #1 SMP PREEMPT Debian 1:6.12.25-1+rpt1 (2025-04-30) aarc
h64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 12 20:17:59 2025
mjrovai@raspi-5-SD:~$
mjrovai@raspi-5-SD:~$ hostname -I
192.168.4.209 fde3:6154:bba3:1:aed5:ada5:ab3d:a37e
mjrovai@raspi-5-SD:~$
```

Update System and test the camera

1. Package mgmt:
 - `sudo apt update`
 - `sudo apt upgrade`
 - `sudo reboot`
2. List the installed cameras:
`rpicas-hello --list-cameras`
3. Capture a 640x480 JPEG image:
`rpicas-jpeg --output test_cli_camera.jpg --width 640 --height 480`
4. Use the command `ls` to check if the image was saved in the current directory and transfer it to your desktop with FileZilla.



Increasing SWAP Memory

1. First, turn off the swap-file:
`sudo dphys-swapfile swapoff`
2. Next, open and modify the file /etc/dphys-swapfile. For that, we will use the **nano** text editor:
`sudo nano /etc/dphys-swapfile`
 - a. Search for the **CONF_SWAPSIZE** variable (default is 500) and
 - b. update it to **2000**: **CONF_SWAPSIZE=2000**, and
 - c. save the file: **CTRL+X** -> **Y** => **Enter**.
3. Next, turn on the swapfile again and reboot the Raspberry Pi:
`sudo dphys-swapfile setup`
`sudo dphys-swapfile swapon`
`sudo reboot`

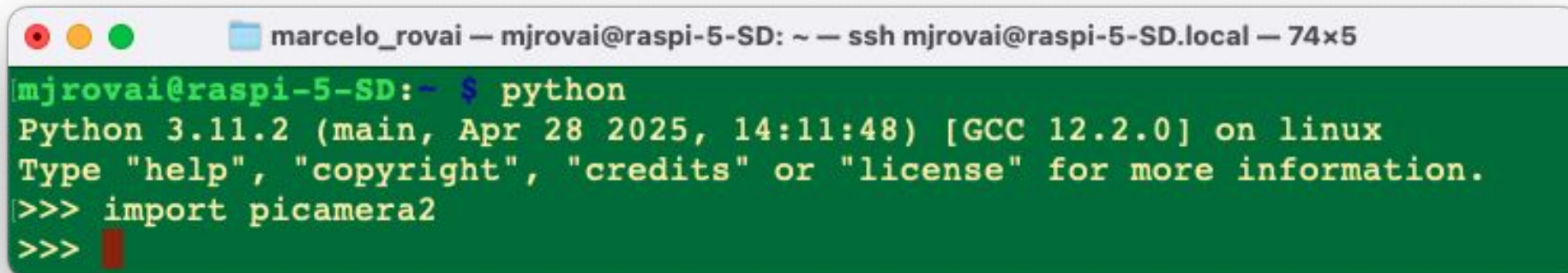
Installing Required **System Level** Libraries

Root Installation

1. Install **Python tools** and **camera libraries** (Picamera2)

```
sudo apt install -y python3-pip python3-venv python3-picamera2  
sudo apt install -y libcamera-dev libcamera-tools libcamera-apps
```

2. Testing **picamera2** installation



```
mjrovai@raspi-5-SD:~$ python  
Python 3.11.2 (main, Apr 28 2025, 14:11:48) [GCC 12.2.0] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> import picamera2  
>>>
```


Setting up a Virtual Environment

1. Create a **virtual environment** with access to system packages to manage dependencies:

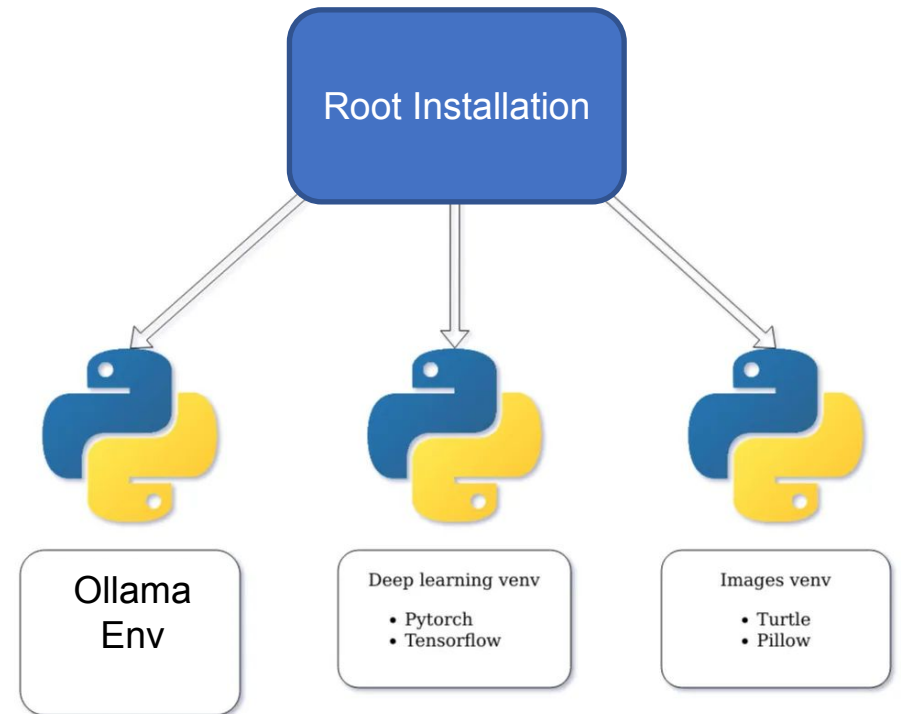
```
python3 -m venv ~/ollama --system-site-packages
```

2. **Activate** the environment:

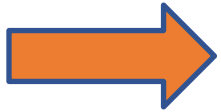
```
source ~/ollama/bin/activate
```

To **exit** the virtual environment, use:

```
deactivate
```



Install Python Packages (inside Virtual Environment)



Ensure you're in the virtual environment (venv)

1. Install Python Packages

```
pip install numpy pillow # Image processing
pip install matplotlib # For displaying images
pip install opencv-python # Computer vision
```

2. Verify installation

```
pip list | grep -E "(numpy|pillow|opencv|picamera)"
```

If the Numpy version is not 1.24.2

```
pip uninstall numpy
pip install numpy==1.24.2
```

```
marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 84x6
Successfully installed numpy-2.2.6 opencv-python-4.12.0.88
(ollama) mjrovai@raspi-5-SD:~ $ pip list | grep -E "(numpy|pillow|opencv|picamera)"
numpy                2.2.6
opencv-python        4.12.0.88
picamera2             0.3.31
(ollama) mjrovai@raspi-5-SD:~ $
```

Setting up Jupyter Notebook



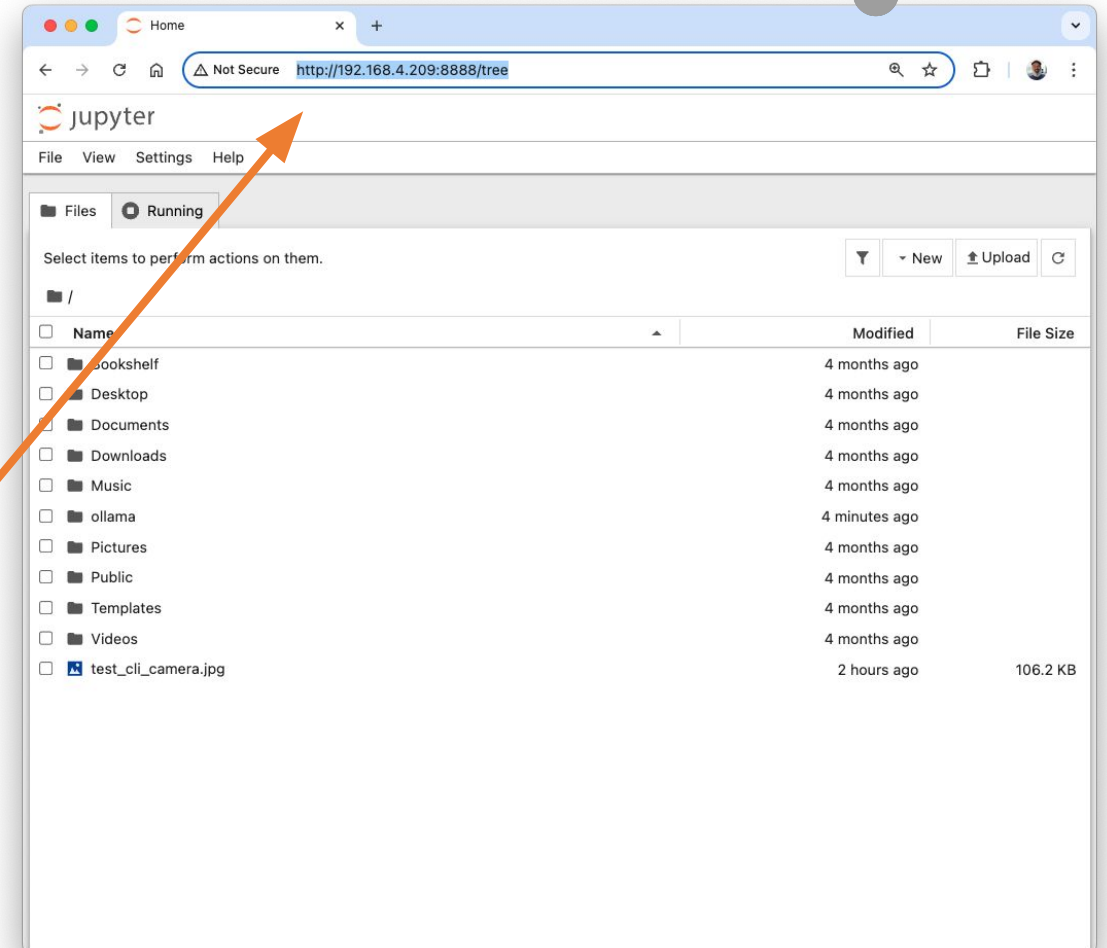
```
pip install jupyter jupyterlab notebook
```

```
jupyter notebook --ip=[YOUR IP ADDRESSES] --no-browser
```

Copy the **Raspberry Pi's IP address** and the provided **token** in a web browser.

```
marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 98x13

To access the server, open this file in a browser:
  file:///home/mjrovai/.local/share/jupyter/runtime/jpserver-2176-open.html
Or copy and paste one of these URLs:
  http://192.168.4.209:8888/tree?token=243ab7baf679502d2634fd0c27b7748bb60c2ecc236b8af7
  http://127.0.0.1:8888/tree?token=243ab7baf679502d2634fd0c27b7748bb60c2ecc236b8af7
[I 2025-09-15 14:30:17.662 ServerApp] Skipped non-installed server(s): basedpyright, bash-language
-server, dockerfile-language-server-nodejs, javascript-typescript-langserver, jedi-language-server
, julia-language-server, pyrefly, pyright, python-language-server, python-lsp-server, r-languagese
rver, sql-language-server, texlab, typescript-language-server, unified-language-server, vscode-css
-language-server-bin, vscode-html-language-server-bin, vscode-json-language-server-bin, yaml-language
-server
```



Creating a working directory

```
cd Documents
```

```
mkdir OLLAMA
```

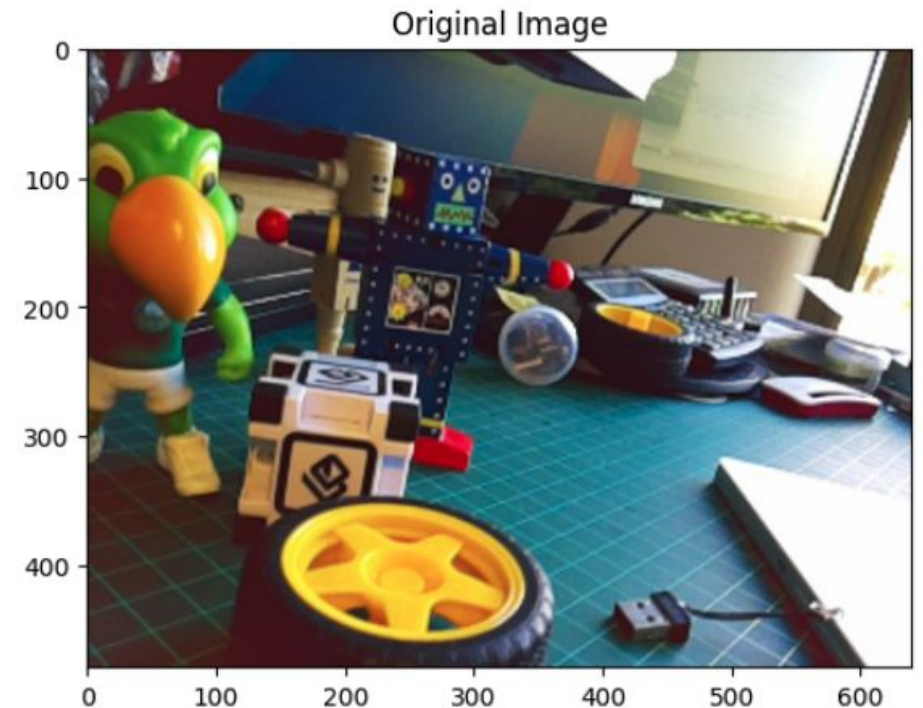
```
cd OLLAMA
```



```
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image

img_path = "../..../test_cli_camera.jpg"
img = Image.open(img_path)

# Display the image
plt.figure(figsize=(6, 6))
plt.imshow(img)
plt.title("Original Image")
plt.show()
```



Testing with Live Camera

```
from picamera2 import Picamera2
import time

# Initialize camera
picam2 = Picamera2()
picam2.start()

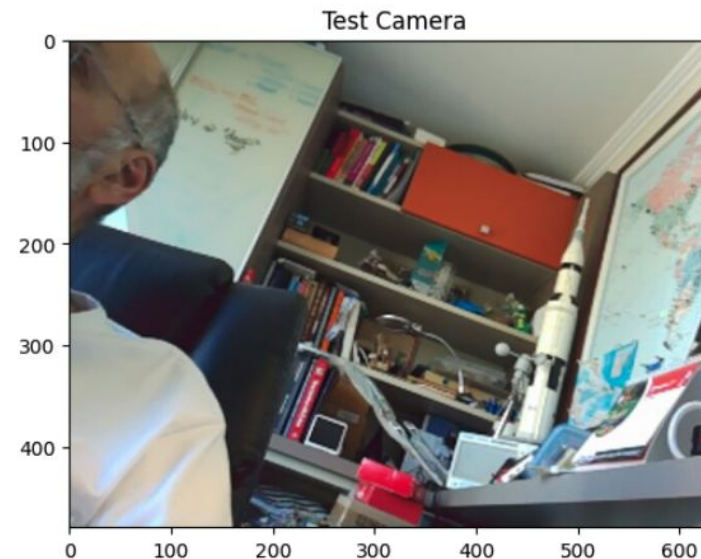
# Wait for camera to warm up
time.sleep(2)

# Capture image
picam2.capture_file("class16_test.jpg")
print("Image captured:
class16_test.jpg")

# Stop camera
picam2.stop()
picam2.close()
```

```
img_path = "class16_test.jpg"
img = Image.open(img_path)

# Display the image
plt.figure(figsize=(6, 6))
plt.imshow(img)
plt.title("Test Camera")
plt.show()
```



Questions?



Prof. Marcelo J. Rovai

rovai@unifei.edu.br

