



Q



TRANG CHỦ

Q

THƯ VIỆN

GROUP FACEBOOK

KÊNH YOUTUBE

GITHUB

NGƯỜI NẤU MÌ

① X

SSD NVMe lên đến 4TB

Tận hưởng hiệu năng NVMe trong SSD tự mã hóa hỗ trợ bảo vệ dữ liệu cuối.

[Kingston Technology](#)
[Learn More](#)
[Home](#) » [2020](#) » [August](#) » [23](#) »

Triển khai hệ thống nhận diện khuôn mặt và đo thân nhiệt trên Raspberry Pi4

Triển khai hệ thống nhận diện khuôn mặt và đo thân nhiệt trên Raspberry Pi4

By Nguyễn Chiến Thắng | 23/08/2020 | Computer Vision | 13 Comments

Lượt xem: 14,867

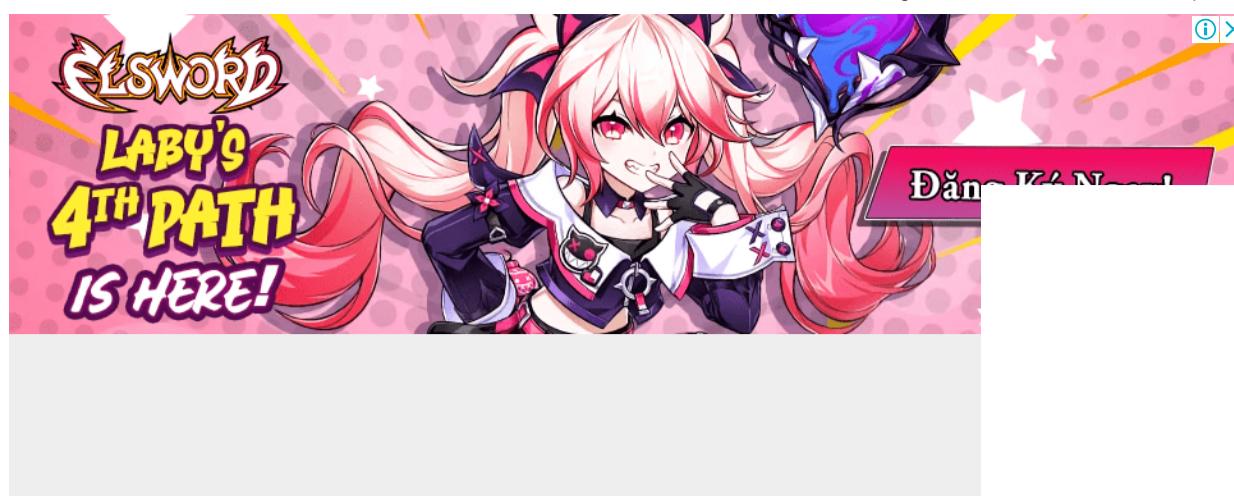
Hello tuần mới anh em Mì, hôm nay chúng ta sẽ cùng triển khai thực tế hệ thống nhận diện khuôn mặt và đo thân nhiệt trên Raspberry Pi4 nhé.

Okie, nhận diện khuôn mặt, phát hiện khuôn mặt thì mình làm nhiều rồi và trên Blog: <https://miae.vn> cũng đã có nhiều bài viết về vấn đề này, các bạn có thể tham khảo nhé.

Hôm nay chúng ta sẽ làm một món mới là kết hợp thêm camera nhiệt để đo nhiệt độ. Bài toán của chúng ta sẽ là nhận biết người ra vào và ghi lại nhiệt độ của người đó, đồng thời cảnh báo nếu như người đó bị sốt. Việc này sẽ góp phần ngăn chặn sự bùng phát của dịch COVID-19.

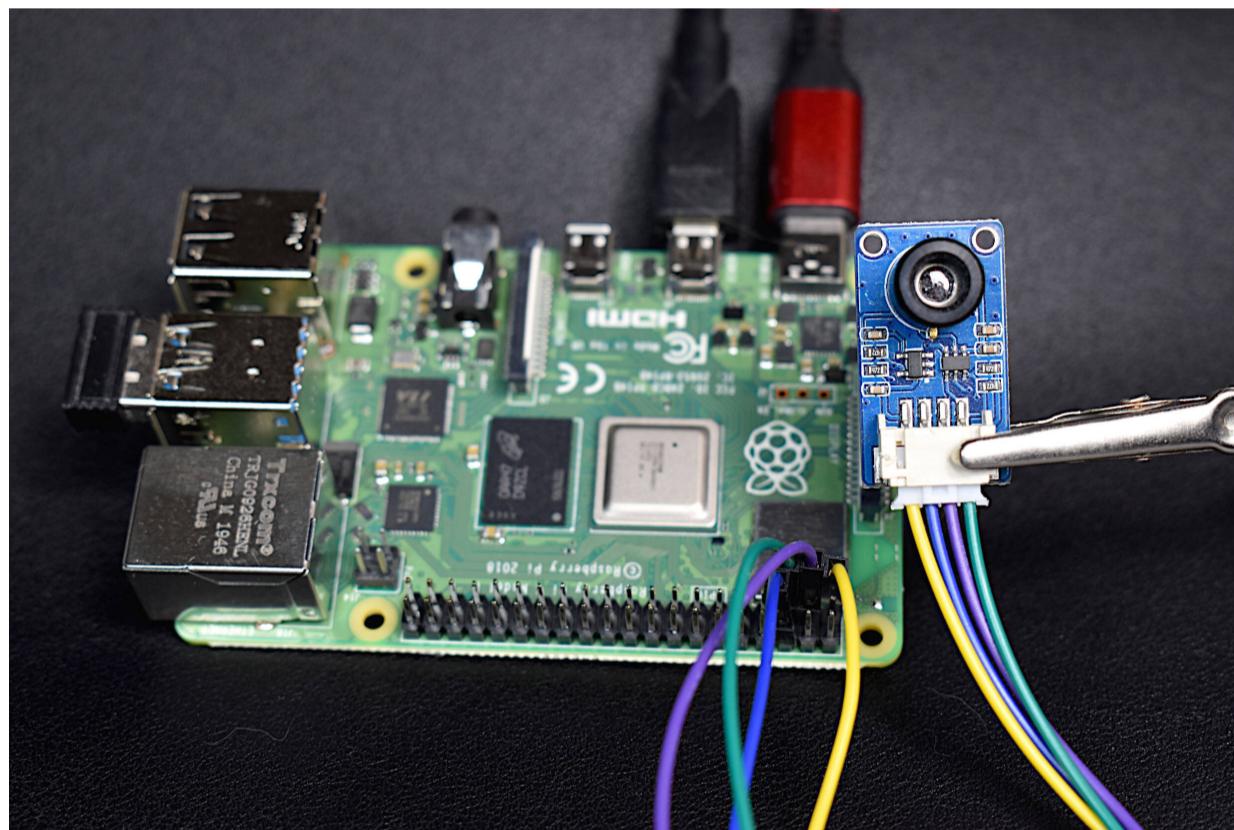
Rồi! Bắt đầu nhé các mem!

Phần 1 – Chuẩn bị các thiết bị cần thiết



Ở bài đo thân nhiệt này mình cần có các món sau:

- Một Raspberry Pi (3 hoặc 4 đều okie nhé, nhưng mình recommend 4 cho nhanh) hoặc Jetson Nano. Trong bài này mình xài Pi 4 do jetson Nano đang bận cho dự án khác
- Một camera thông thường để nhận hình ảnh, nhận khuôn mặt. Ở đây mình sử dụng Logitech Webcam.
- Một camera nhiệt, ở đây mình xài camera nhiệt MLX90640 mua tại Mlab.vn [tại đây](#). Chú ý là mình chỉ đưa link, các bạn kiểm tra kỹ hàng và bảo hành khi mua nhé.
- Một màn hình hiển thị có đầu vào HDMI hoặc VGA.
- Các bộ chuyển đổi và dây cắm cần thiết.



Camear nhiệt cắm vào Pi 4 (<https://makersportal.com/blog/2020/6/8/high-resolution-thermal-camera-with-raspberry-pi-and-mlx90640>)

Phần 2 – Cài đặt OS cho Pi

Phần này bạn này chưa cài đặt OS trên thẻ nhớ thì đọc, bạn nào cài rồi thì next sang Phần sau luôn nhé.

Trước đây ta phải cài đặt thủ công còn bây giờ thì chính chủ đã hỗ trợ khá nhiều, các bạn chỉ cần tải phần mềm **Raspberry Pi Imager** về và làm theo hướng dẫn là xong cái thẻ nhớ! Dễ vãi!

Link tải cho các bạn nhé: [Tại đây](#).



Giao diện ứng dụng Raspberry Pi

Sau khi xong quá trình này thì các bạn cắm thẻ nhớ vào Pi, bật điện, cắm màn và bàn phím, chuột vào để hoàn tất quá trình cài đặt Pi lần đầu nhé.

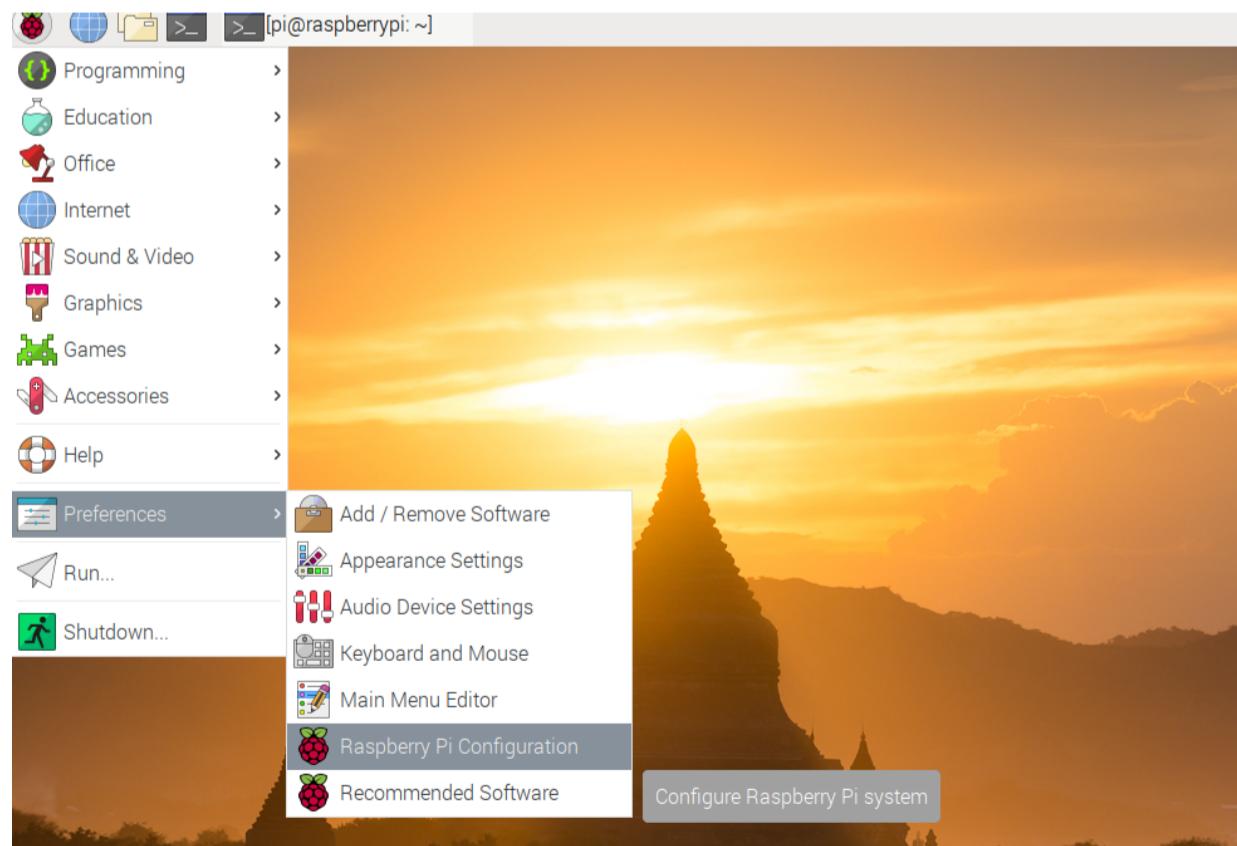
Sau khi đã hiện được giao diện Desktop, các bạn tiến hành cài đặt các gói lập trình theo đúng guide tại link này: [tại đây](#).

Tiếp đến là cài đặt các gói thư viện để có thể truy cập vào camera nhiệt:

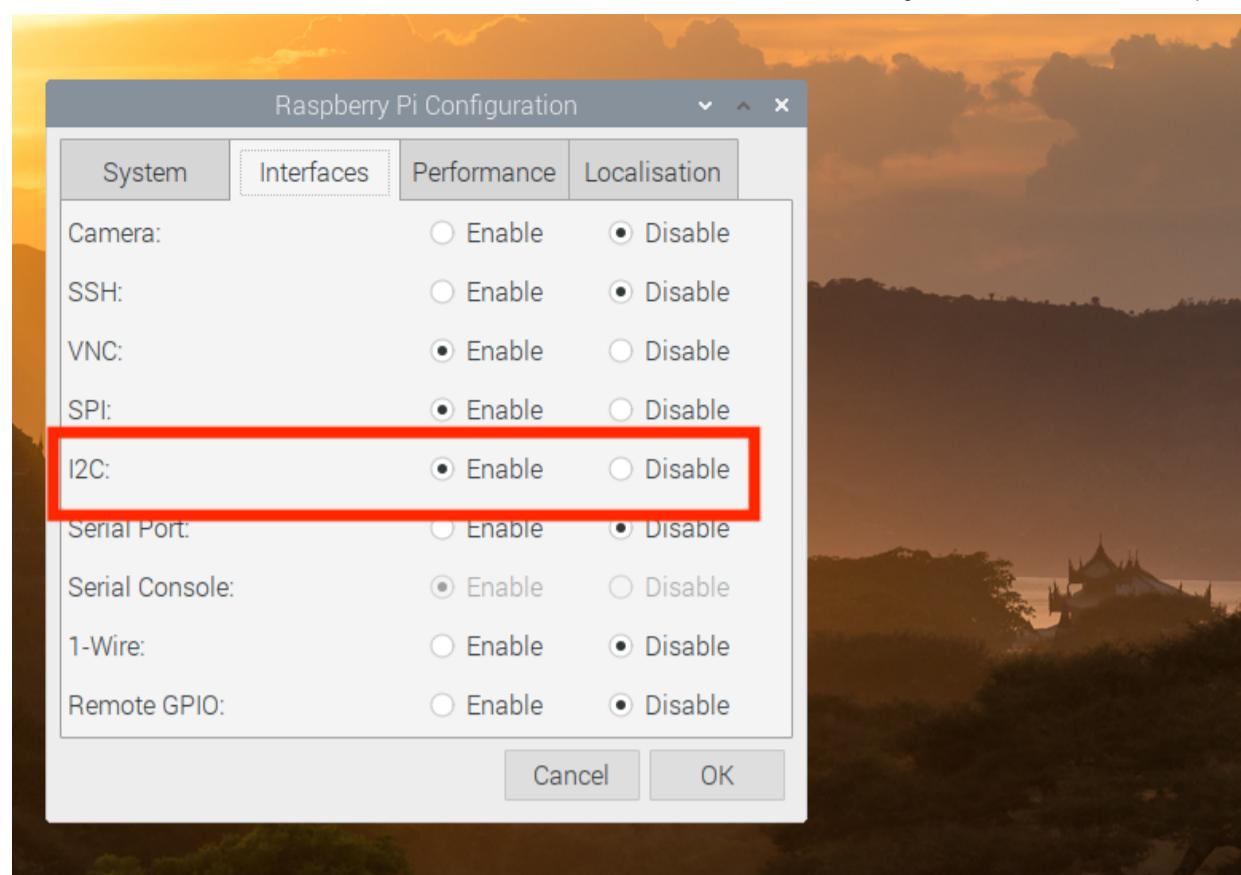
```
pip3 install matplotlib scipy numpy
sudo apt-get install -y python-smbus
sudo apt-get install -y i2c-tools
pip3 install RPI.GPIO adafruit-blinka
pip3 install adafruit-circuitpython-mlx90640
```

Rồi, vậy là xong môi trường, xong OS, tiếp đến phần cấu hình thêm cho Pi tí.

Bạn vào menu Start, chọn tiếp Preferences > Raspberry Pi Configuration

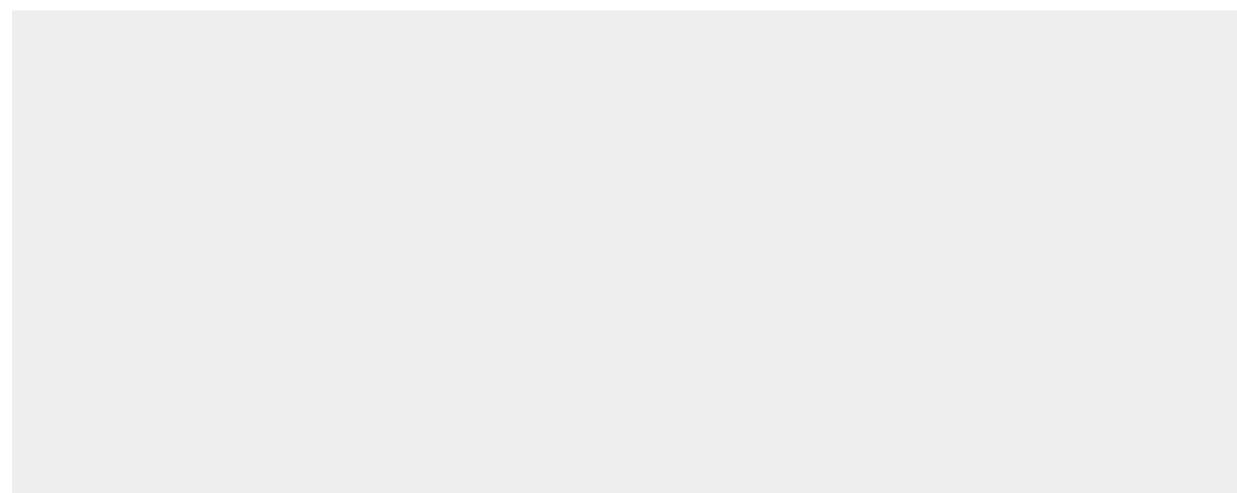


Và sau đó bật I2C lên thành Enabled như hình:



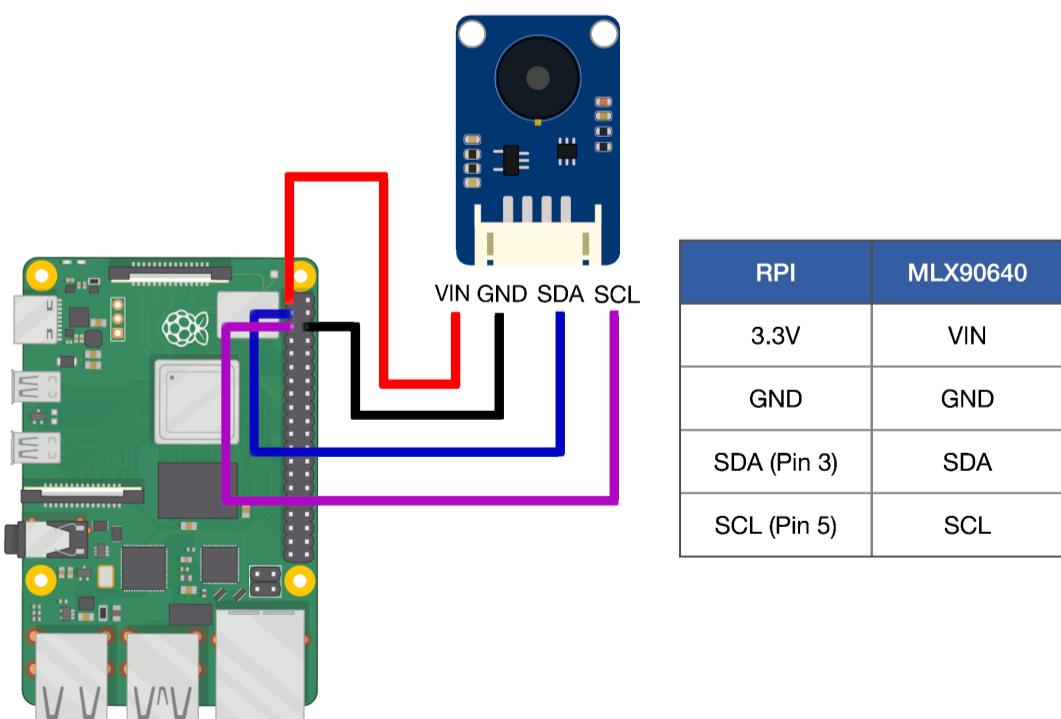
Các bạn cũng có thể tick thêm vào SSH và VNC và Camera cho sau này đỡ phải bật nhé. Chú ý bạn nào không dùng USB Camera mà dùng camera của Pi thì bắt buộc phải bật Camera.

Sau khi làm tất cả các bước trên thì các bạn reboot lại Pi 1 lần cho nó nhận các tham số nhé!



Phần 3 – Cắm camera đo thân nhiệt vào Pi

Phần camera nhiệt này các bạn cắm theo đúng hình sau vào các chân của Pi nhé:



Link. <https://makersportal.com/blog/2020/6/8/high-resolution-thermal-camera-with-raspberry-pi-and-mlx90640>

Có vài điểm cần chú ý cho các bạn:

- Chân VIN đôi khi còn viết là VCC, các bạn cắm vào chân 3.3V hoặc 5V trên Pi đều được.
- Chân GND các bạn có thể chọn chân GND bất kì trên Pi
- Riêng 2 chân còn lại thì phải cắm chuẩn.

Sau khi cắm xong các bạn thử lệnh:

```
sudo i2cdetect -y 1
```

Nếu kết quả trả ra như hình là done, xong con ong nhé!

```
pi@raspberrypi:~ $ sudo i2cdetect -y 1
 0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00: --
10: --
20: --
30: -- 33 --
40: --
50: --
60: --
70: --
pi@raspberrypi:~ $
```

Link <https://makersportal.com/blog/2020/6/8/high-resolution-thermal-camera-with-raspberry-pi-and-mlx90640>

Các bạn cũng cắm luôn webcam logitech vào một cổng USB bất kì nhé (ưu tiên chọn cổng màu xanh – USB 3.0 – nếu có).

Phần 4 – Hướng xử lý bài toán đo thân nhiệt

Để các bạn nắm được tinh thần mình cũng xin chia sẻ luôn vài ý như sau:

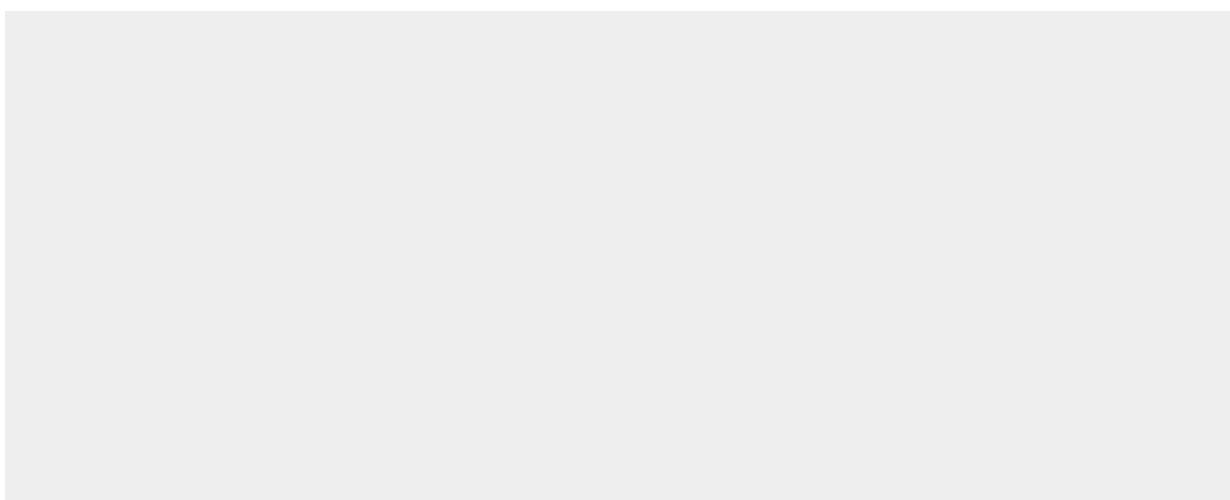
- Thứ nhất, Pi4 không quá khoẻ nên mình không áp dụng Deep Learning vào đây, chỉ sử dụng HAAR CASCADE để detect khuôn mặt mà thôi. Các bạn hoàn toàn có thể áp dụng dlib, MTCNN... nếu cần (recommend nâng lên Jetson Nano)
- Thứ hai, đó là đây là bài toán demo nên mình ưu tiên về mặt giới thiệu, chưa thực sự tối ưu nhé.

Trong bài này Pi của chúng ta sẽ đọc ảnh từ 2 camera (1 camera thường và 1 camera nhiệt).

Flow xử lý sẽ như sau:

- Bước 1: Đọc ảnh từ camera thường, phát hiện khuôn mặt trong ảnh. Ví dụ có 1 khuôn mặt tại vị trí hình chữ nhật (x1,y1)-(x2,y2).
- Bước 2: Ta tiến hành đọc ảnh từ camera nhiệt (đọc đồng thời nhé) sau đó tìm đến vị trí (x1,y1) – (x2,y2) trên camera nhiệt và tính các giá trị: nhiệt độ lớn nhất, nhiệt độ nhỏ nhất và nhiệt độ trung bình (mình ví dụ thế).
- Bước 3: Ghi nhận lại khuôn mặt và nhiệt độ đo được vào CSDL, đồng thời cảnh báo phát ra loa nếu như nhiệt độ vượt ngưỡng quy định (ví dụ là 37.5 chặng hạn).

Để làm được việc này bạn phải chạy chương trình và cân chỉnh 2 camera sao cho khung hình của 2 camera tương đồng nhau nhé!



Phần 5 – Viết code cho chương trình đo thân nhiệt

Okie việc đầu tiên là chúng ta sẽ khai báo các thư viện cần thiết. Bình thường mình sẽ không nói đến phần này do nó không có gì đặc biệt. Tuy nhiên hôm nay chúng ta sẽ dùng một số thư viện khá lạ nên mình sẽ note thêm:

```
import time, board, busio
import numpy as np
import adafruit_mlx90640
import matplotlib.pyplot as plt
import matplotlib.patches as patches
from scipy import ndimage
import cv2
import matplotlib as mpl
mpl.rcParams['toolbar'] = 'None'
```

Các bạn để ý sẽ có thư viện board, busio, ada... là các thư viện để làm việc với bo mạch và các cổng vào ra trên Raspberry Pi nhé. Bạn nào chạy trên máy tính lõi mình không chịu trách nhiệm nha, haha!

Tiếp theo là chúng ta sẽ khai báo phần nhận diện khuôn mặt cũng như kết nối với webcam logitech cắm USB. Cái này như mình đã nói, mình dùng tạm HAAR CASCADE để tăng tốc độ xử lý:

```
# Load phan detect khuong mat
face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

# Ket noi camera Thuong
cap = cv2.VideoCapture(0)
```

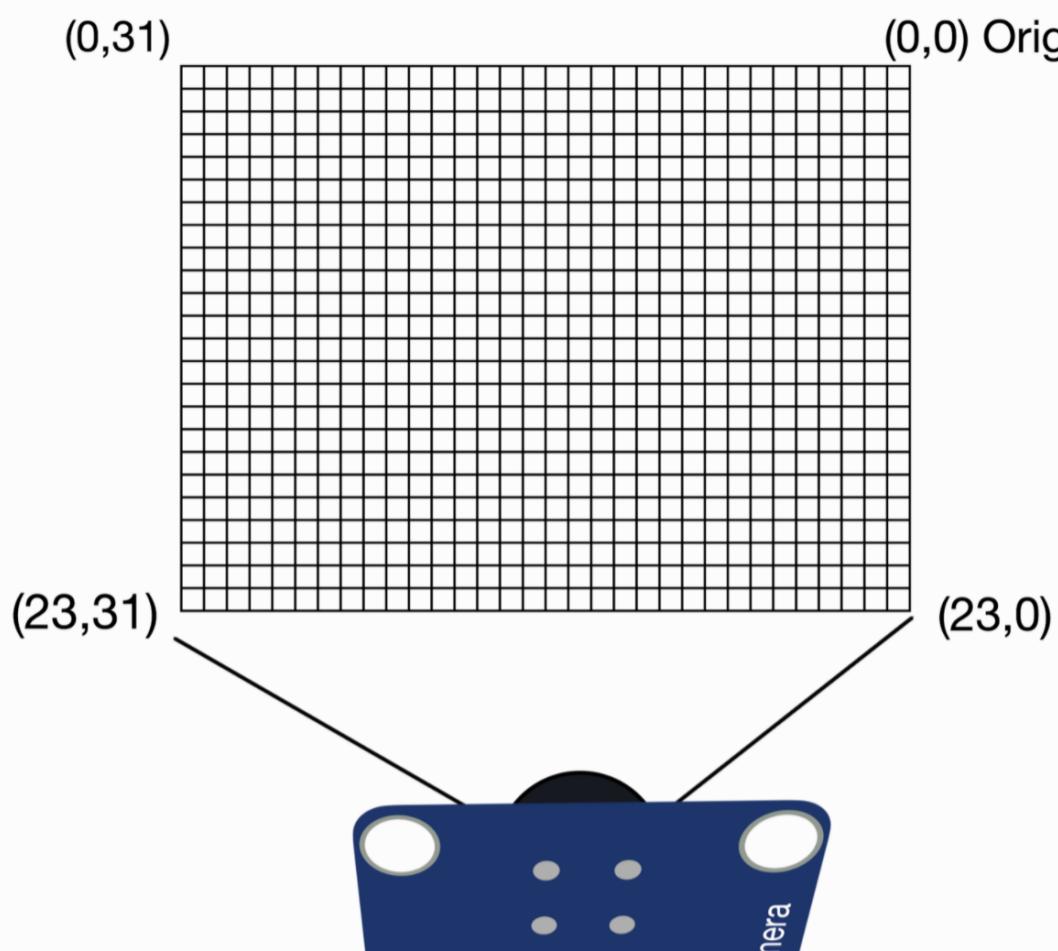
Rồi, phần này thì bình thường như cân đường hộp sữa, lâu nay chúng ta đã viết code này rồi.

Bây giờ đến phần kết nối camera nhiệt, chúng ta sẽ dùng đoạn chương trình sau:

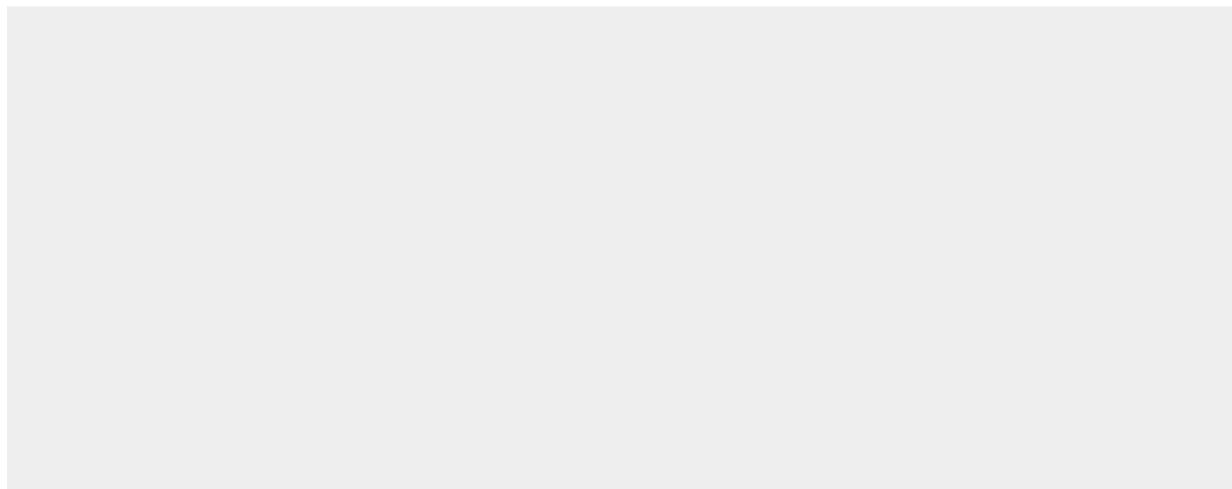
```
# Kết nối camera Nhiệt
i2c = busio.I2C(board.SCL, board.SDA, frequency=100000)
mlx = adafruit_mlx90640.MLX90640(i2c)
mlx.refresh_rate = adafruit_mlx90640.RefreshRate.REFRESH_8_HZ
mlx_shape = (24, 32)

# Phóng to kích thước ma trận nhiệt
mlx_interp_val = 10
mlx_interp_shape = (mlx_shape[0] * mlx_interp_val, mlx_shape[1] * mlx_interp_v
```

Ở đây các bạn chú ý một chút ở biến mlx_shape = (24,32). Nghĩa là kích thước sensor của cam nhiệt chỉ là 24×32 thôi, sau đó nó được nén suy lên 240×320 nhé.



Link: <https://makersportal.com/blog/2020/6/8/high-resolution-thermal-camera-with-raspberry-pi-and-mlx90640>



Rồi bây giờ là đoạn vẽ giao diện chính, gồm 3 thành phần là : 1 ô để hiện ảnh nhiệt (therm1), 1 ô hiển thị ảnh thường (therm2) và 1 dòng text hiển thị nhiệt độ của mặt người do được.

```
# Vẽ giao diện
fig = plt.figure(figsize=(8, 6))
fig.canvas.set_window_title('Hệ thống giám sát nhiệt độ ra/vào')
fig.canvas.toolbar_visible = False
ax = fig.add_subplot(1, 2, 1)
ax2 = fig.add_subplot(1, 2, 2)
fig.subplots_adjust(0.05, 0.05, 0.95, 0.95)

# Hiển thị ảnh nhiệt
therm1 = ax.imshow(np.zeros(mlx_interp_shape), interpolation='none',
                   cmap=plt.cm.bwr, vmin=25, vmax=45) # preemptive image

# Hiển thị ảnh thường
therm2 = ax2.imshow(np.zeros((240, 320)), interpolation='none',
                    cmap=plt.cm.bwr, vmin=25, vmax=45)

fig.canvas.draw()
ax_background = fig.canvas.copy_from_bbox(ax.bbox)
fig.show()
```

Okie, màu mè hoa lá đã xong, giờ đến phần chính của chương trình, chúng ta sẽ lắp liên tục và đọc ảnh đồng thời thì cam chính, cam nhiệt... và xử lý theo đúng flow ở mục trên:

```

while True:
    ret, img = cap.read()

    if ret:
        count = count + 1
        img = cv2.resize(img, dsize=(320, 240))
        img2 = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

        if count%5==0:
            faces = face_cascade.detectMultiScale(gray, scaleFactor=1.3, minNe
            if len(faces)==1:
                for (x, y, w, h) in faces:
                    cv2.rectangle(img2, (x, y), (x + w, y + h), (0, 255, 0), 2
            try:
                plot_update(x,y,w,h) # update plot
            except:
                continue
            therm2.set_data(img2)

```

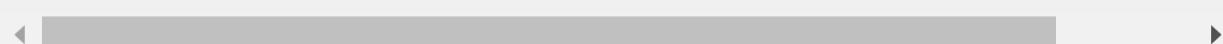


Ở đây các bạn chú ý 1 hàm `plot_update`, hàm này sẽ làm nhiệm vụ vẽ lại ảnh nhiệt, ảnh thường và đặc biệt là kiểm tra xem nhiệt độ hiện tại như nào? Nếu < 38 độ thì là bình thường, hiện màu vàng và ngược lại sẽ cảnh báo và hiện màu đỏ rực rỡ:

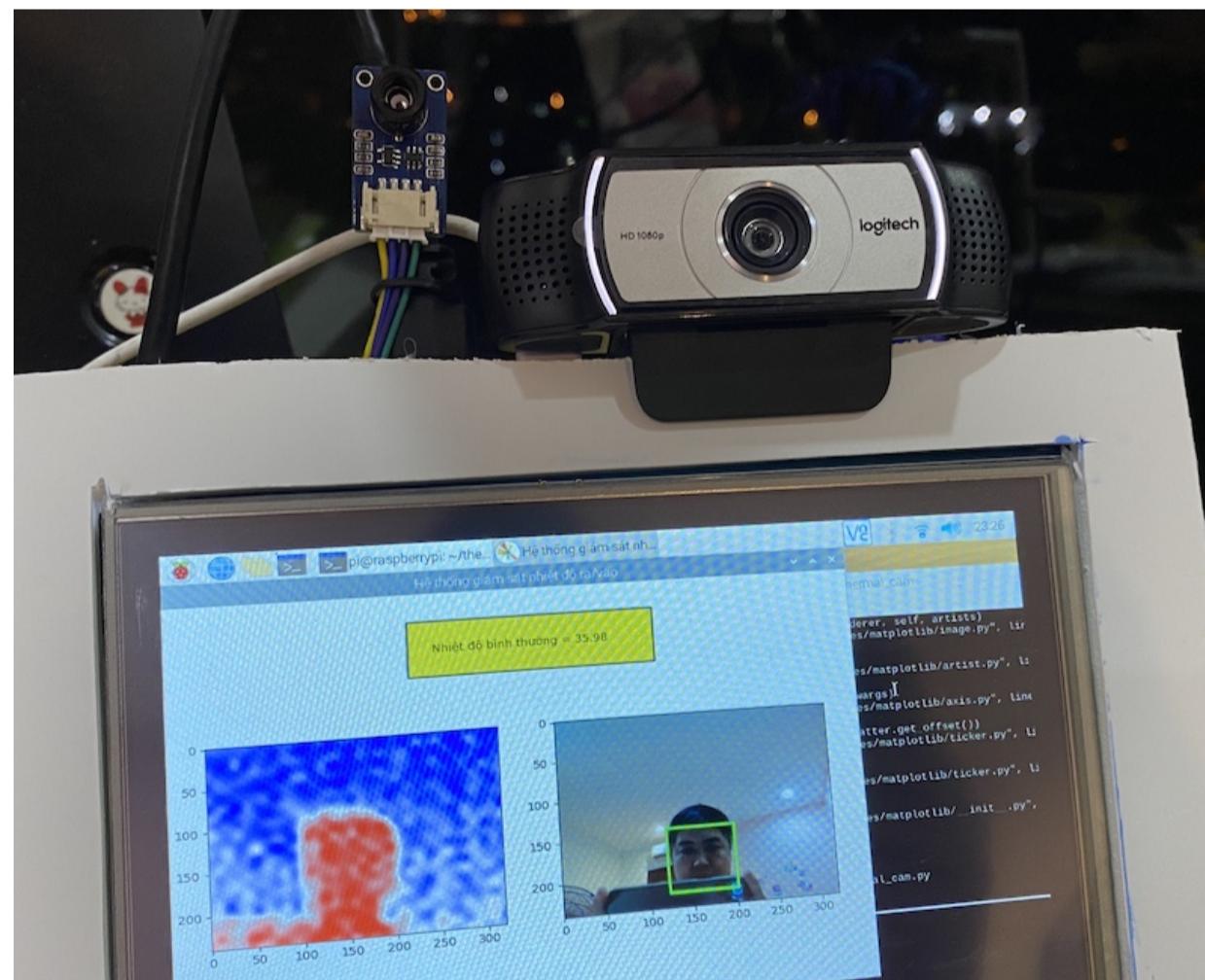
```

# Tính nhiệt face
vface_temp = round(np.max(data_array[y:y+h,x:w+x]), 2)
if vface_temp<38:
    ax.text(250, -100, 'Nhiệt độ bình thường = ' + str(vface_temp) +
           bbox={'facecolor': 'yellow', 'alpha': 1, 'pad': 20})
else:
    ax.text(250, -100, 'Có biểu hiện sốt. Nhiệt độ = ' + str(vface_temp) +
           bbox={'facecolor': 'red', 'alpha': 1, 'pad': 20})

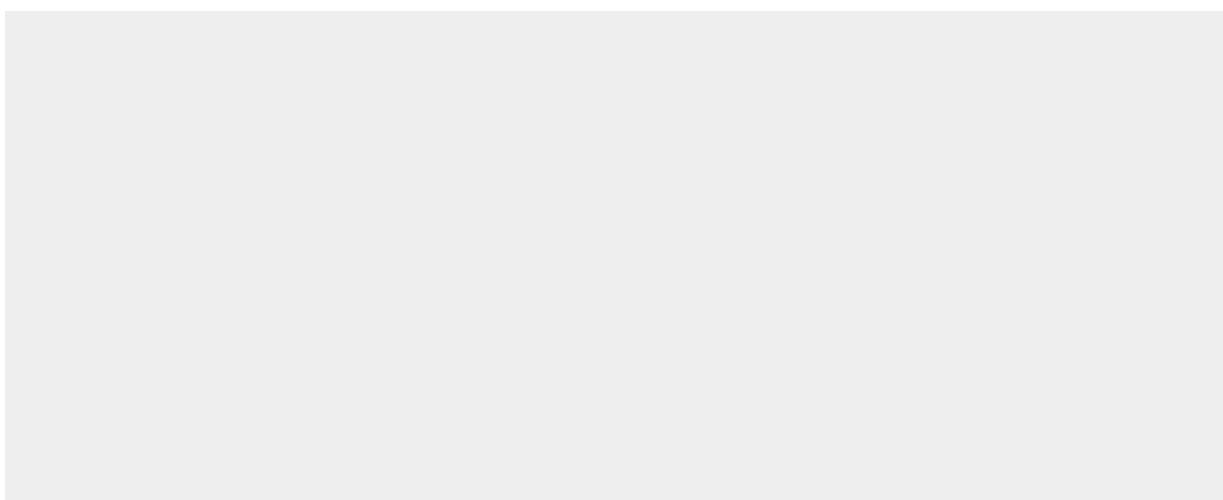
```



Okie và khi chạy thử chương trình thì kết quả nhận diện mặt và đo nhiệt khá tốt:



Các bạn có thể tìm thấy source của mình tại [repo này trên github](#) nhé!



Mình có quay thêm clip ở trên kênh Youtube của Mì AI. Anh em có thể bấm vào bên dưới để xem thêm nhé!



Phần 6 – Kết luận và lời cảm ơn

Chắc chắn khi các bạn đọc bài này sẽ thắc mắc các loại câu hỏi trên trời dưới biển nên mình xin trả lời luôn:

- Hỏi: Thế người ta đi quay mặt chỗ khác thì sao? Trả lời: Bài này mình làm sample với một hệ thống chấm công phải đưa mặt vào để nhận diện. Các bạn có thể cải tiến cho case quay mặt đi bằng cách detect người thay cho detect mặt.
- Hỏi: Thế nhiều người lấp nhau thì sao? Cái đó thì các bạn detect đầu, hoặc detect max nhiệt trong ảnh để xử lý.

Ngoài ra, nhân đây mình xin cảm ơn anh **Hồ Viết Hà, Trường Cao đẳng Nghề Đà Nẵng** (tác giả của hệ thống kiểm soát ra vào chợ ở Đà Nẵng tại đây <https://baomoi.com/danang-thu-nghiem-cong-kiem-soat-luot-nguoi-thong-minh/c/36126508.epi>) đã cho mình mượn thiết bị cam nhiệt để viết bài cho các bạn. Cảm ơn anh em Mlab.vn đã hỗ trợ không kể ngày đêm!

Chúc các bạn thành công!

#MìAI

Fanpage: <http://facebook.com/miaiblog>

Group trao đổi, chia sẻ: <https://www.facebook.com/groups/miaigroup>

Website: <http://miai.vn>

Youtube: <http://bit.ly/miaiyoutube>

Bài tham khảo từ hãng sản xuất tại đây <https://makersportal.com/blog/2020/6/8/high-resolution-thermal-camera-with-raspberry-pi-and-mlx90640>

auto thermal check , covid-19 , đo thân nhiệt ,
 đo thân nhiệt tự động , face detect , haar cascade ,
 kiểm tra nhiệt độ tự động , raspberry Pi , thermal
 cam

Related Post

[Re-implement nhận diện đeo khẩu trang \(Face Mask Detection\) bằng YOLO](#)

27/04/2020

[Train,test và triển khai YOLOv6 với custom data \(Phần 1/2\) – Mì AI](#)

31/08/2022

[Train model SSD Mobilenet với Tensorflow 2 trên Colab – Mì AI](#)

03/01/2022

13 Replies to “Triển khai hệ thống nhận diện khuôn mặt và đo thân nhiệt trên Raspberry Pi4”



BLan

[06/09/2020 AT 4:58 AM](#)

Dạ anh ơi,
 Anh cho em hỏi ảnh nhiệt mà camera nhiệt trả về, các pixels của nó đã là mang giá trị nhiệt độ
 luôn rồi hả anh?



Nguyễn Chiến Thắng

[06/09/2020 AT 9:19 AM](#)

Đúng rồi em. Nó trả về ma trận nhiệt.
 Cần thêm gì em cứ lên Group trao đổi, chia sẻ: <https://facebook.com/groups/miaigroup> nhé!



Trần Đình Cung

[14/01/2021 AT 10:03 AM](#)

em chào anh



Lê Văn Phương

[10/09/2020 AT 4:54 AM](#)

Độ chính xác thế nào?, khoảng cách đo? mình đang xài con amg8833 kết hợp với hệ thống face
 của mình mà thấy độ chính xác kém quá đang tính mua con MLX90640 nhưng không biết độ
 chính xác của nó thế nào?



Lê Văn Phương

[10/09/2020 AT 6:51 AM](#)

Độ chính xác nó bao nhiêu? Khoảng cách đo? Hiện tại mình đang dùng con AMG8833 độ chính
 xác tệ quá tính mua con MLX90640 nhưng không biết chất lượng thế nào?

Pingback: [Các phương pháp Liveness Detection trong nhận diện khuôn mặt \(eKYC, Chấm công...\) - Mì AI](#)



quanvq

02/03/2021 AT 5:27 AM

Bác Thắng NC cho bài về Jetson Nano đi ạ. Thanks!

Nguyễn Chiến Thắng02/03/2021 AT 5:57 AM

Hihi triển khai trên Jetson khá giống Pi. Chỉ khác tí code nên bạn cần thêm gì cứ post lên <https://www.facebook.com/groups/miaigroup> nhé!



Tu Anh

24/04/2021 AT 10:35 AM

Cho em hỏi có thể nhận dạng nhiều người cùng lúc được không anh? Và phải đứng lại nhìn vào camera hay di chuyển nhanh qua vậy anh? Em cảm ơn

Nguyễn Chiến Thắng26/04/2021 AT 3:08 PM

Cái này do mình code thôi em. Cái này anh đang code là 1 người thì phải. Lâu quá a ko nhớ nữa 😊



Thảo

01/04/2022 AT 10:16 AM

Em chào anh, anh cho em hỏi là máy tính đang dùng window 10 thì có thể cài win cho Raspberry Pi không ạ?

Nguyễn Chiến Thắng04/04/2022 AT 9:10 AM

Máy tính em dùng OS nào không liên quan đến Pi nhé. Pi dùng Window nặng lắm. Dùng hệ điều hành riêng cho Pi nhé. Cần thêm gì em post lên <https://facebook.com/groups/miaigroup> nhé!

Pingback: [Tổng hợp 10+ thông tin \[Nhận dạng khuôn mặt raspberry pi\], hướng dẫn chi tiết từ A-Z – VPN Miễn Phí](#)

