**BÁO CÁO**

**KẾT THÚC MÔN HỌC**

**MACHINE-LEARNING**

**---------------------**

**ĐỀ TÀI**: [Face Recog 2.0] Nhận diện khuôn mặt bằng MTCNN Và Facenet

Mục lục

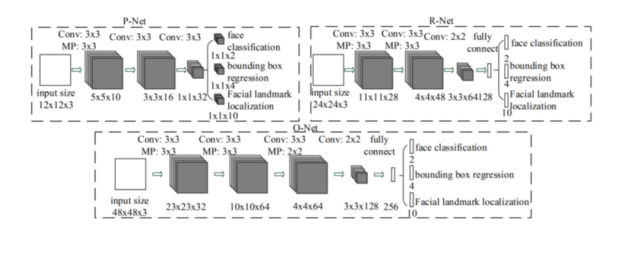
1. Tìm hiểu khái niệm
2. Khởi tạo project
3. Tiến hành train model để nhận diện khuôn mặt
4. kết quả nhận được

**Phần 1 : Tìm Hiểu Khái Niệm**

1. MTCNN Là gì ?

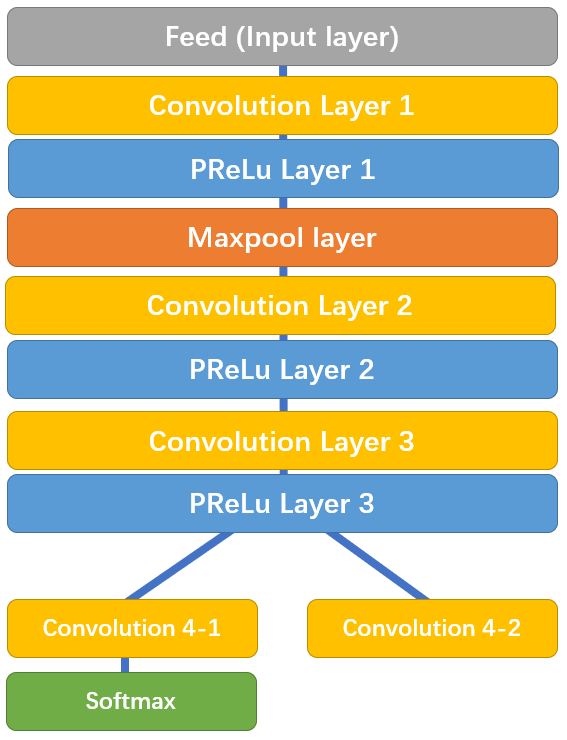
MTCNN là viết tắt của Multi-task Cascaded Convolutional Networks. Nó là bao gồm 3 mạng CNN xếp chồng và đồng thời hoạt động khi detect khuôn mặt. Mỗi mạng có cấu trúc khác nhau và đảm nhiệm vai trò khác nhau trong task. Đầu ra của MTCNN là vị trí khuôn mặt và các điểm trên mặt như: mắt, mũi, miệng…

MTCNN hoạt động theo 3 bước, mỗi bước có một mạng neural riêng lần lượt là: P-Net, R-Net và O-net…

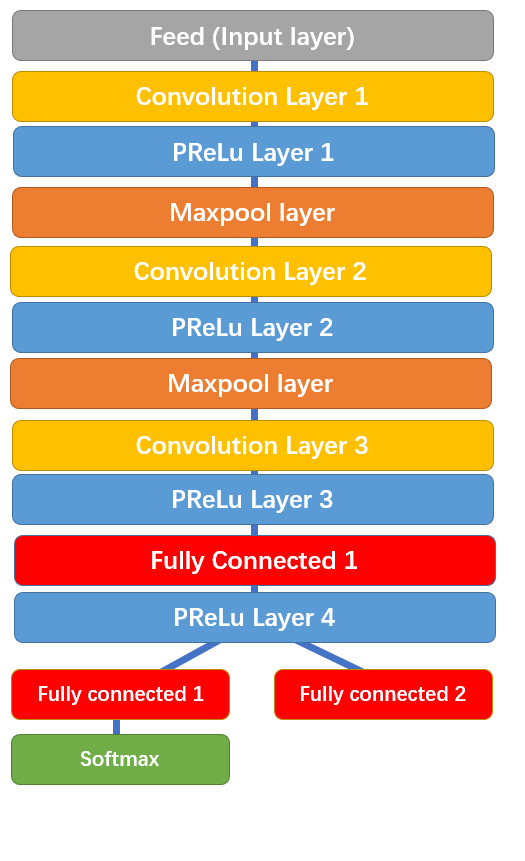


bức ảnh đầu vào, nó sẽ tạo ra nhiều bản sao của hình ảnh đó với các kích thước khác nhau.

Tại P-Net, thuật toán sử dụng 1 kernel 12x12 chạy qua mỗi bức hình để tìm kiếm khuôn mặt. Sau lớp convolution thứ 3, mạng chia thành 2 lớp. Convolution 4-1 đưa ra xác suất của một khuôn mặt nằm trong mỗi bounding boxes, và Convolution 4-2 cung cấp tọa độ của các bounding boxes.

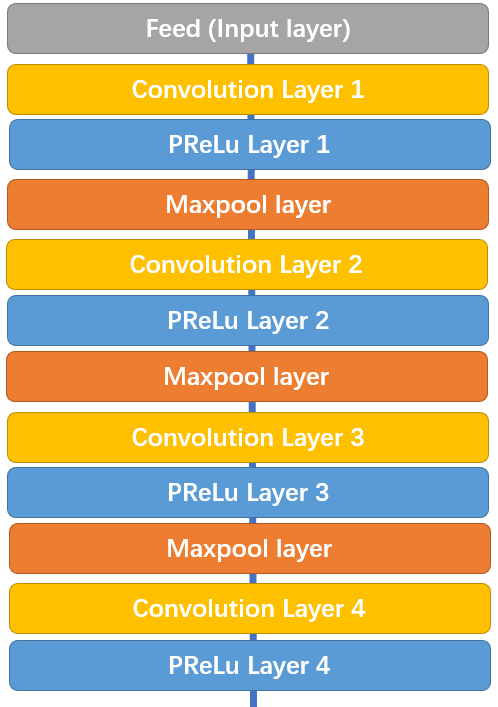


R-Net có cấu trúc tương tự vói P-Net. Tuy nhiên sử dụng nhiều layer hơn. Tại đây, network sẽ sử dụng các bounding boxes đc cung cấp từ P-Net và tinh chỉnh là tọa độ

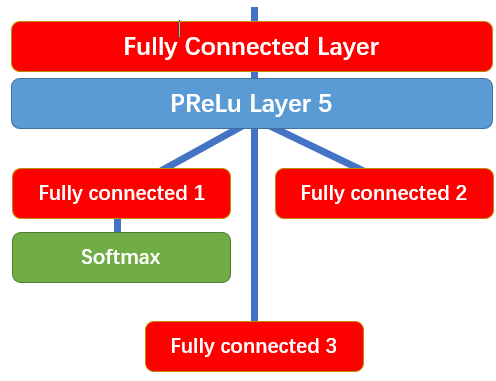


Tương tự R-Net chia ra làm 2 layers ở bước cuối,cung cấp 2 đầu ra đó là tọa độ mới của các bounding boxes, cùng độ tin tưởng của nó.

O-Net lấy các bounding boxes từ R-Net làm đầu vào và đánh dấu các tọa độ của các mốc trên khuôn mặt.

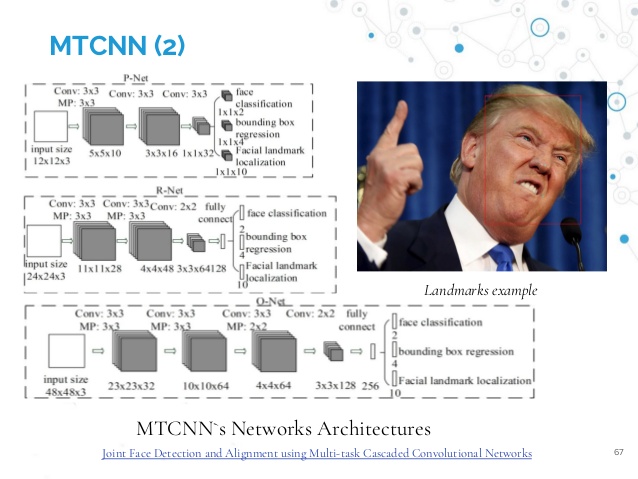


Ở bước này, thuật toán đưa ra 3 kết quả đầu ra khác nhau bao gồm: xác suất của khuôn mặt nằm trong bounding box, tọa độ của bounding box và tọa độ của các mốc trên khuôn mặt (vị trí mắt, mũi, miệng)



1. Facenet Là gì ?

**Facenet** là của Google giới thiệu năm 2015. **Facenet**, thực chất là một CNN có nhiệm vụ tách các đặc trưng của một ảnh mặt. Điểm đặc biệt tạo nên sự khác biệt của **Facenet** **là** nó sử dụng hàm lỗi Triplet để tối thiểu hóa khoảng cách giữa các gương mặt tương đồng và tối đa hóa khoảng cách đến những gương mặt không tương đồng, vì vậy **facenet** có thể phân biệt rất chính xác người với người. ( và model này thì mình cứ cho ảnh vào (đúng size của nó) thì nó trả ra 1 vector 128 features cho 1 khuôn mặt. Sau đó dùng SVM để phân nhóm các vector đó vào các nhóm để biết vector đó là mặt của ai.)

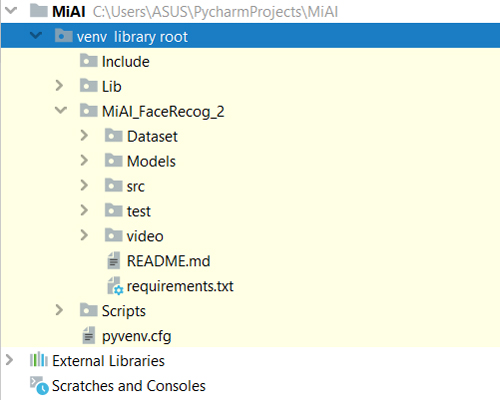
****

## **Phần 2. Khởi tạo project**

**Tạo thư mục và clone git:**

Các bạn tạo 1 project mới để lưu tất cả các file của bài này nhé. Sau khi tạo xong, các bạn gõ lệnh sau để lấy mã nguồn về:

git clone https://github.com/thangnch/MiAI\_FaceRecog\_2

****

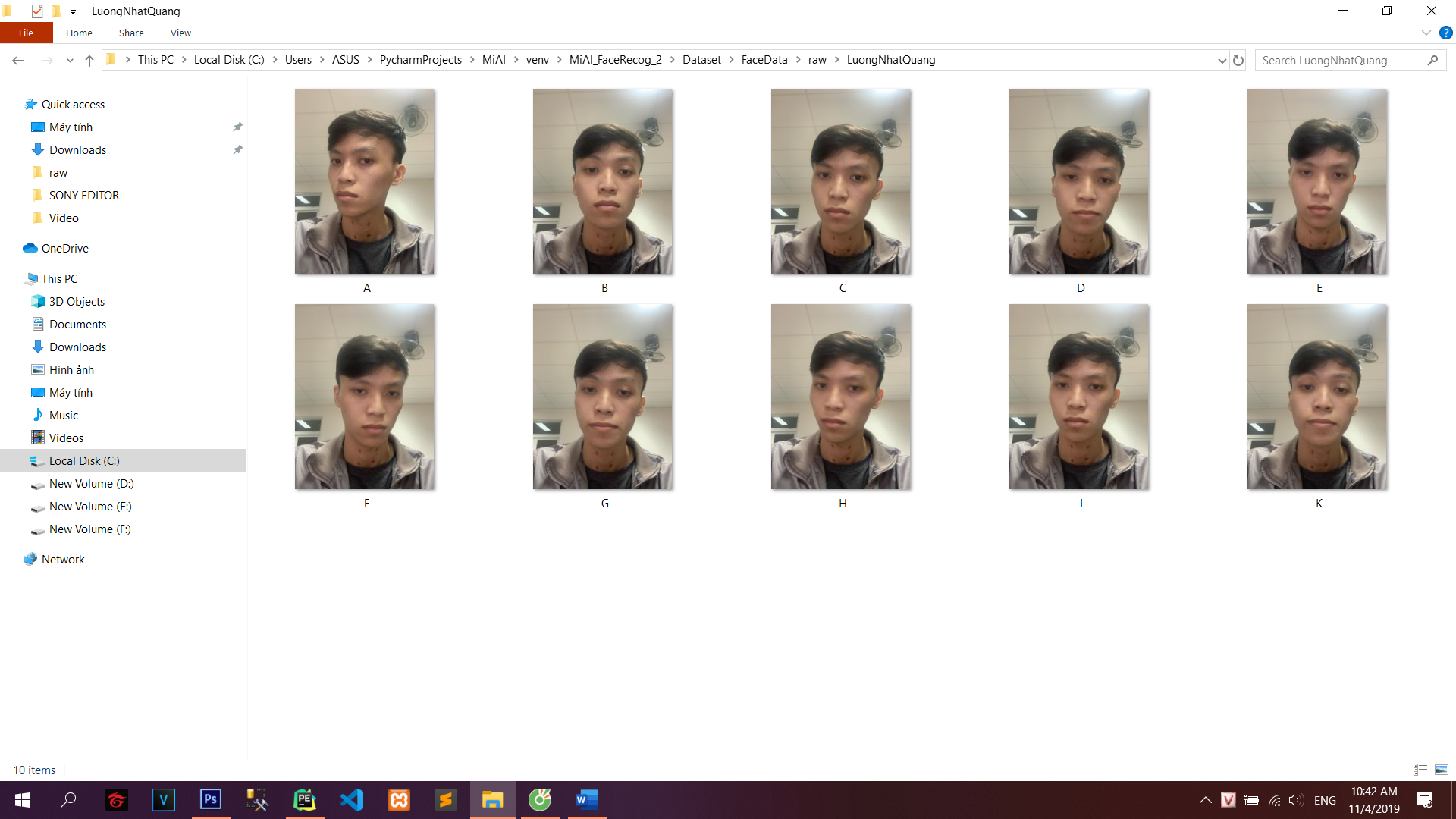
( Hình ảnh minh họa 1 )

Tiếp theo, các bạn các thư mục như sau:

* Tạo thư mục Dataset trong MiAI\_FaceReg\_2, trong đó tạo tiếp thư mục FaceData và dưới FaceData là tạo tiếp 2 thư mục raw và processed.
* Tạo thư mục Models trong MiAI\_FaceReg\_2 để chờ sẵn lát lưu model sau.

**Chuẩn bị ảnh khuôn mặt để train**

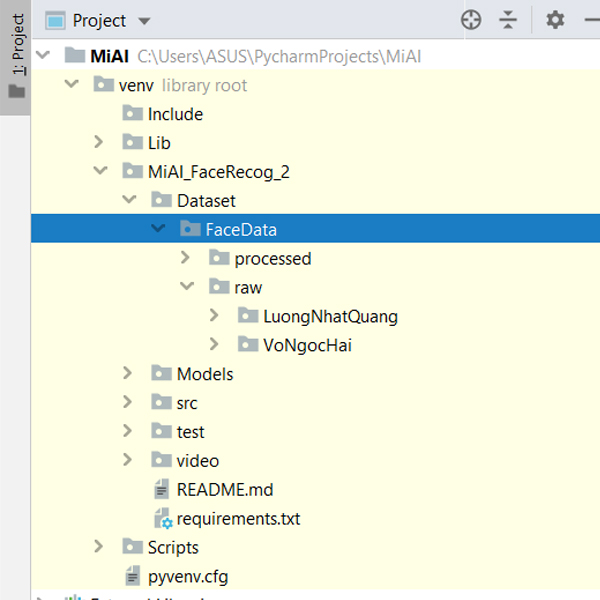
Bây giờ các bạn sưu tầm ảnh của 2 người trở lên, mỗi người 10 tấm hình rõ mặt .Mình ví dụ 2 người tên là LuongNhatQuang và VoNgocHai nhé. Các bạn tạo 02 thư mục LuongNhatQuang và VoNgocHai trong thư mục raw và copy ảnh của 2 người vào riêng 2 thư mục đó, ảnh của ai vào thư mục của người đó nhé.



(hình ảnh minh họa 2)

Chú ý: Trong các ảnh bạn sưu tầm, chỉ có đúng 1 khuôn mặt của người đó, không được có quá 1 khuôn mặt/ảnh nhé.

Ví dụ cây thư mục của mình để các bạn tham khảo:



(ảnh mình họa 3)

**Cài đặt các thư viện cần thiết**

Các bạn đứng ở thư mục MiAI\_FaceRecog\_2 chạy lệnh sau để cài tất cả các thư viện cần thiết:

pip install -r requirements.txt

Nếu gặp lỗi trong quá trình cài đặt,bạn nên xóa phiên bản tensorflow trong file requirement.txt sẽ cài được

**Tiền xử lý dữ liệu để cắt khuôn mặt từ ảnh gốc**

Với chỗ ảnh mà bạn đã sưu tầm bên trên, có thể là ảnh cả người, bây giờ chúng ta sẽ cắt riêng khuôn mặt ra để train nhé. Các bạn chuyển về thư mục MiAI\_FaceRecog\_2 và chạy lệnh :

python src/align\_dataset\_mtcnn.py Dataset/FaceData/raw Dataset/FaceData/processed --image\_size 160 --margin 32 --random\_order --gpu\_memory\_fraction 0.25

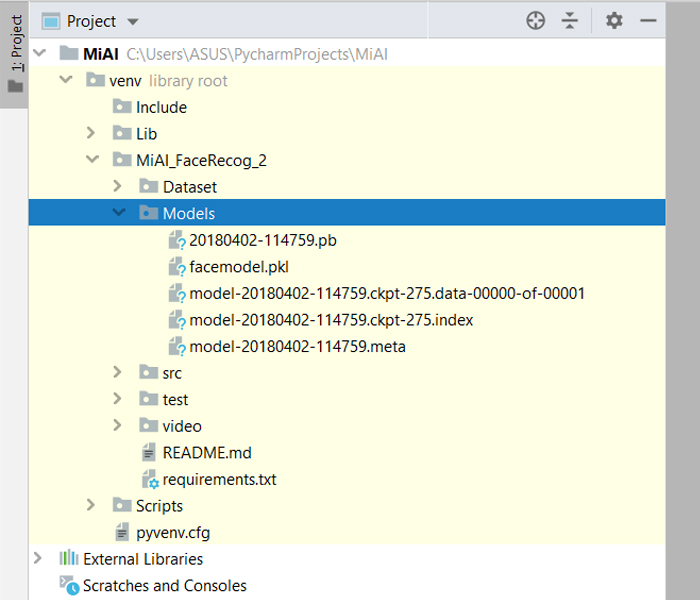
Chạy xong thấy nó hiển thị dạng “Total number of images: …” là thành công rồi đó. Các bạn để ý sẽ thấy có thêm thư mục processed có cấu trúc tương tự thư mục raw nhưng chỉ chứa dữ liệu khuôn mặt dã được xử lý. Ví dụ như ảnh dưới:



**(ảnh minh họa 4)**

**Tải dữ liệu pretrain của Facenet về máy:**

Các bạn tải weights pretrain về tại link này : (<https://drive.google.com/drive/u/0/folders/1cMln7JzcNPHJP46ovrUlveycOkRI_RPa>) . Sau khi tải xong về, các bạn copy toàn bộ file tải về vào thư mục Models, **chú ý chỉ lấy file, bỏ hết các thư mục như hình bên dưới của mình** (không có file facemodel.pkl như bên dưới đâu nhé, mình chụp nhầm chút).

****

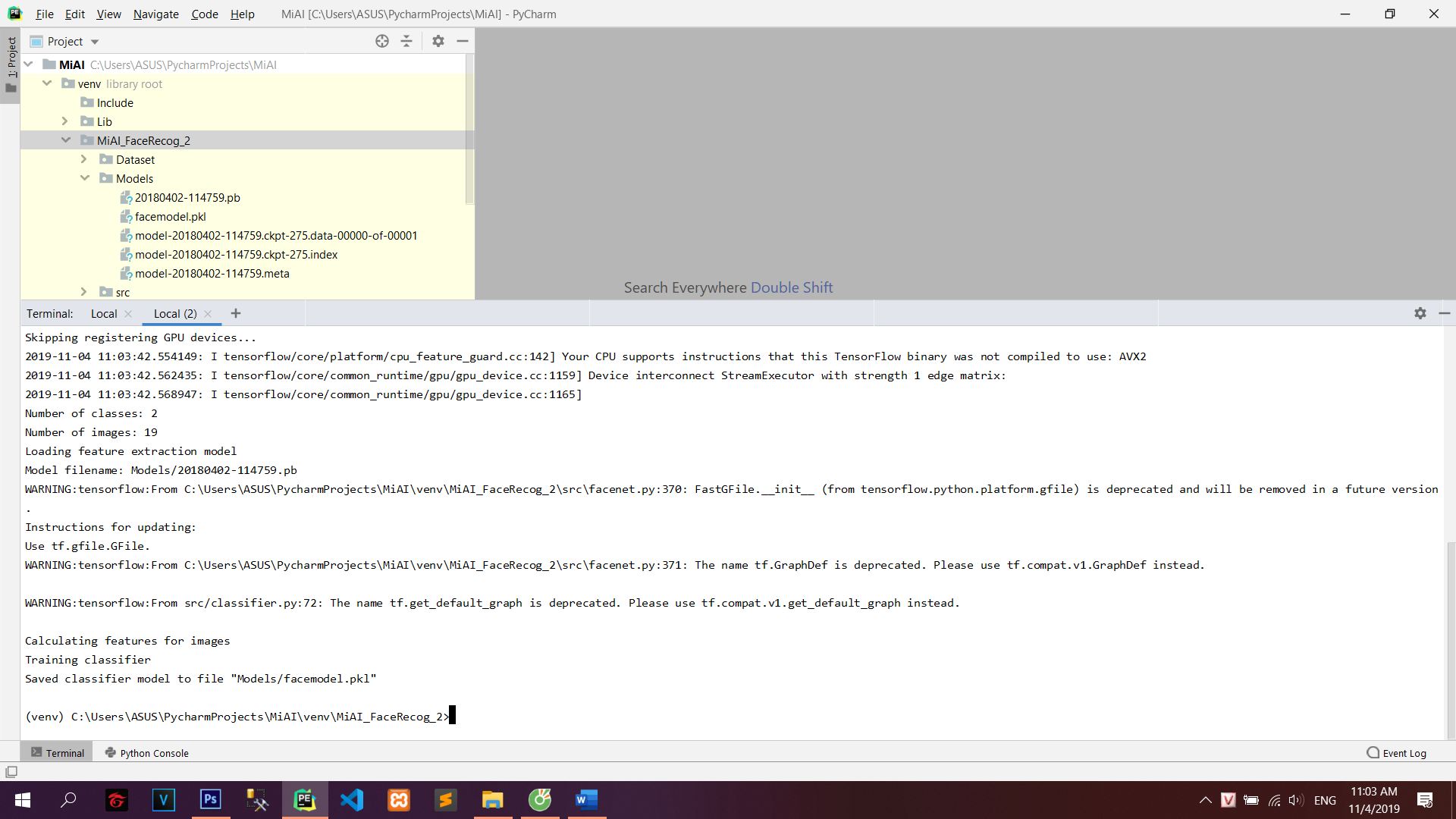
(hình ảnh minh họa 5)

## **Phần 3. Tiến hành train model để nhận diện khuôn mặt**

Bây giờ các bạn chuyển về thư mục **MiAI\_FaceRecog\_2** nếu đang đứng ở thư mục khác nhé. Sau đó chạy lệnh train:

python src/classifier.py TRAIN Dataset/FaceData/processed Models/20180402-114759.pb Models/facemodel.pkl --batch\_size 1000

Bây giờ các bạn đi lấy một tách trà và ngồi đợi nhé, khi nào màn hình hiện lên chữ “Saved classifier model to file “Models/facemodel.pkl” là done!Như hình minh họa dưới nhé. Chúng ta đã train xong rồi, tận hưởng thành quả thôi.



(hình ảnh minh họa 6)

## **Phần 4. Kết quả nhận được**

Chúng ta sẽ kiểm thử model đã train với 2 nguồn video: một là từ webcam và 2 là nhận diện trong một file video có sẵn. Bắt đầu nhé

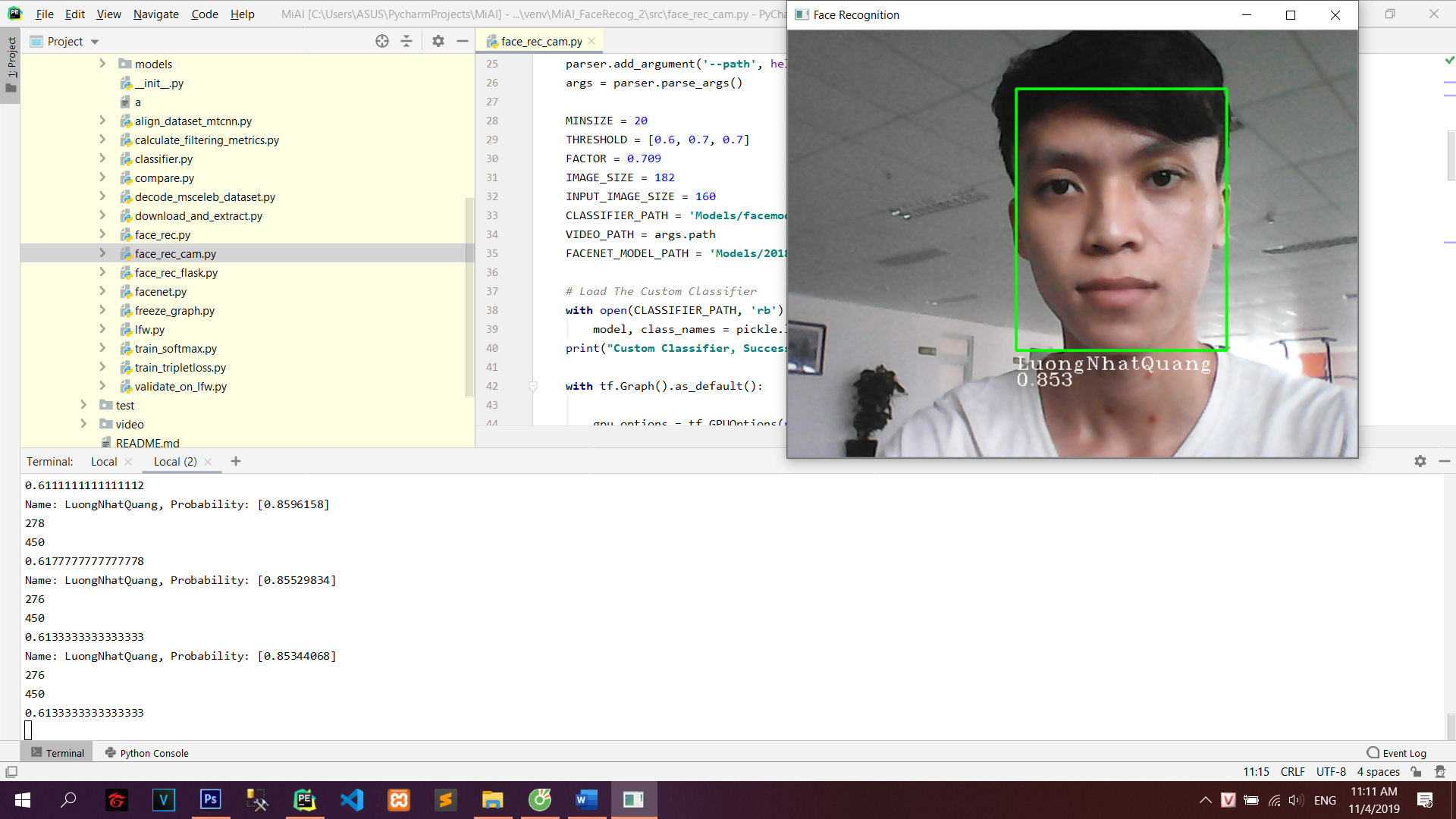
Kiểm thử với webcam:

Các bạn chạy file face\_rec\_cam.py bằng lệnh sau:

python src/face\_rec\_cam.py

Bạn đợi model load 1 chút và màn hình webcam sẽ hiện lên như sau, nhận ra rất là chuẩn 😀.

Chú ý một số máy sẽ thiếu thư viện imutils,install nó vào và chay lại là ok nhé !



Rồi bây giờ chúng ta thử nhận diện qua video xem sao nhé. Các bạn chạy lệnh:

python src/face\_rec.py --path video/camtest.mp4

Chú ý ở đây mình ví dụ với file camtest.mp4 của mình, các bạn có thể thử với bất kì file video nào của các bạn. Chú ý là giảm độ phân giải xuống tầm 320×200 hoặc 640×480 nếu máy bạn không có GPU hoặc cấu hình thấp nhé, ko là giật tung lên đấy

Và đây là thành quả của chúng ta, các bạn hãy xem video bên dưới nhé:

