

5

CÔNG NGHỆ
INTERNET OF THINGS
HIỆN ĐẠI

Xây dựng ứng dụng AI cơ bản trong mô hình IoT

Lưu hành nội bộ

A. MỤC TIÊU

- Tìm hiểu phần cứng Raspberry Pi 3, cấu trúc của bo mạch phần cứng.
- Làm quen với Tensorflow và numpy, là các thư viện, công cụ để xây dựng ứng dụng AI.
- Tìm hiểu cách hoạt động của một ứng dụng AI cũng như các bước phát triển.
- Xây dựng ứng dụng nhận diện khuôn mặt bằng mô hình facenet.
- Phát triển các giải pháp dựa vào ứng dụng nhận diện khuôn mặt.

B. GIỚI THIỆU

1. Tìm hiểu về phần cứng Raspberry Pi

Sinh viên trả lời những câu hỏi sau đây:

- ✓ Hiện tại có những loại Raspberry Pi nào?
- ✓ Vi xử lý trên Raspberry Pi 3 là loại gì? Có đặc điểm như thế nào?
- ✓ Bộ nhớ trong Raspberry Pi 3 là bao nhiêu?
- ✓ Trên Raspberry Pi 3 có các loại chân (PIN) nào? Chức năng của mỗi loại là gì?

2. Giới thiệu về Google Colab

Colaboratory hay còn gọi là **Google Colab**, là một sản phẩm từ Google Research, nó cho phép chạy các dòng code python thông qua trình duyệt, đặc biệt phù hợp với Data analysis, machine learning và giáo dục. Colab không cần yêu cầu cài đặt hay cấu hình máy tính, mọi thứ có thể chạy thông qua trình duyệt, bạn có thể sử dụng tài nguyên máy tính từ CPU tốc độ cao và cả GPUs và cả TPUs đều được cung cấp cho bạn.

Colab cung cấp nhiều loại GPU, thường là Nvidia K80s, T4s, P4s and P100s, tuy nhiên người dùng không thể chọn loại GPU trong Colab, GPU trong Colab thay đổi theo thời gian. Vì là dịch vụ miễn phí, nên Colab sẽ có những thứ tự ưu tiên trong việc sử dụng tài nguyên hệ thống, cũng như giới hạn thời gian sử dụng, thời gian sử dụng tối đa lên tới 12 giờ.

3. Giới thiệu về Tensorflow

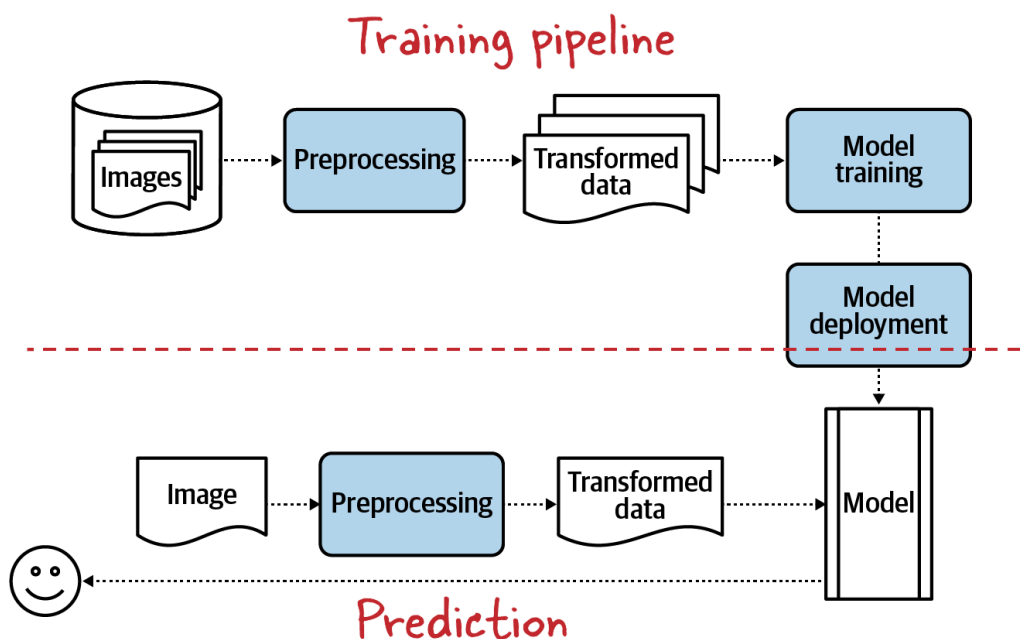
Tensorflow là thư viện mã nguồn mở cho học máy (machine learning - ML) nổi tiếng, được phát triển bởi các nhà nghiên cứu từ Google. Việc hỗ trợ mạnh mẽ các phép toán học để tính toán trong ML và học sâu (deep learning - DL) đã giúp việc tiếp cận các bài toán trở nên đơn giản, nhanh chóng và tiện lợi hơn nhiều.

Các hàm được dựng sẵn trong thư viện cho từng bài toán cho phép Tensorflow xây dựng được nhiều neural network. Nó còn cho phép người dùng tính toán song song trên nhiều máy tính khác nhau, thậm chí trên nhiều CPU, GPU trong cùng 1 máy hay tạo ra các dataflow graph – đồ thị luồng dữ liệu để dựng nên các model.

Được viết bằng C++ và thao tác interface bằng Python nên hiệu suất của Tensorflow cực kỳ tốt. Đối tượng sử dụng nó cũng đa dạng không kém: từ các nhà nghiên cứu, nhà khoa học dữ liệu và dĩ nhiên không thể thiếu các lập trình viên.¹

4. Ứng dụng nhận dạng khuôn mặt

Một ứng dụng trí tuệ nhân tạo thông thường sẽ trải qua các bước sau:



¹ <https://topdev.vn/blog/tensorflow-la-gi/>

Nhận dạng khuôn mặt là quá trình xác định khuôn mặt trong mỗi ảnh là ai, và độ chính xác là bao nhiêu phần trăm. Trong bài thực hành này, sinh viên sử dụng mô hình facenet để thực hiện quá trình Training và Prediction.

C. THỰC HÀNH

Mô tả: Sinh viên sử dụng một Raspberry Pi 3 và một Camera Raspberry Pi 3 ver2.1. Kết nối Camera vào Raspberry Pi, thiết lập VNC để sử dụng Raspberry Pi. Dựa trên mã nguồn được cung cấp sẵn, sinh viên triển khai ứng dụng nhận diện khuôn mặt trên Raspberry Pi.

- **Bước 1:** Tải tài nguyên của bài thực hành số 5 được lưu trữ [tại đây](#).
- **Bước 2:** Tạo Dataset, tạo cây thư mục chứa ảnh dùng cho huấn luyện mô hình:

```
| -Dataset  
    | ---raw  
        | -----NguyenVanA  
            | -----NguyenVanB
```

- **Bước 3:** Cài đặt các thư viện cần thiết:

```
pip install -r requirements.txt
```

- **Bước 4:** Tiền xử lý dữ liệu:

```
python src/align_dataset_mtcnn.py Dataset/raw  
Dataset/processed --image_size 160 --margin 32  
--random_order --gpu_memory_fraction 0.25
```

- **Bước 5:** Huấn luyện mô hình:

```
python src/classifier.py TRAIN Dataset/processed  
Models/20180402-114759.pb Models/facemodel.pkl  
--batch_size 1000
```

- **Bước 6:** Chạy Flask Server:

```
python src/face_rec_flask.py
```

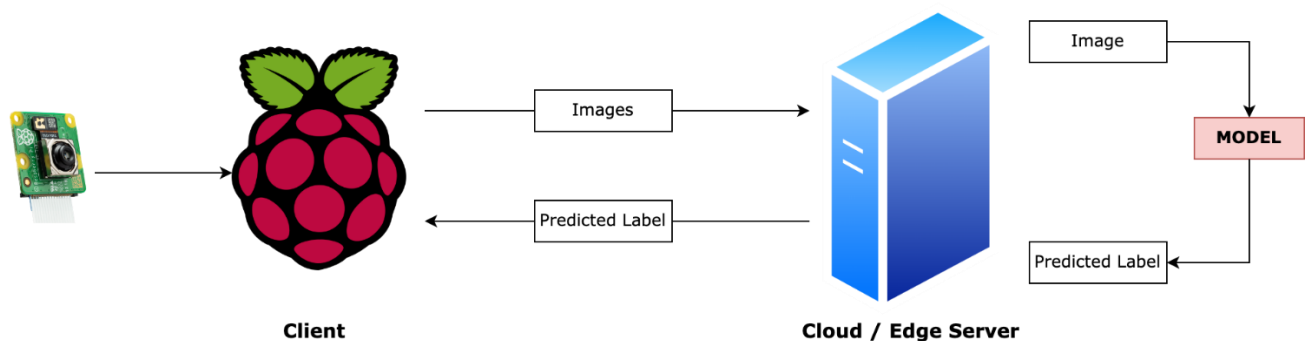
- **Bước 7:** Chạy Client demo nhận diện khuôn mặt:

```
python client.py
```

D. YÊU CẦU & NỘI BÀI

1. Yêu cầu

1. Thực hiện phần C, đảm bảo độ chính xác của các gương mặt đối với các thành viên trong nhóm **trên 60%**. Hãy nêu cách để bạn xử lý được vấn đề này. Từ đó, tìm hiểu những cách nào để tăng cường dữ liệu,... để tăng độ chính xác đối với mô hình?
2. Tích hợp mô hình đã được huấn luyện ở câu 1 vào mô hình IoT cơ bản được thể hiện tại Hình 1. Mô hình gồm có 1 Raspberry Pi và 1 PC / laptop cá nhân. Raspberry Pi đóng vai trò như Client, được sử dụng để thu thập hình ảnh từ camera về người dùng cần nhận diện. PC / laptop cá nhân đóng vai trò như một Edge / Cloud Server dùng để xử lý hình ảnh thu được từ Raspberry Pi và nhận diện hình ảnh đó. Từ đó, trả kết quả nhận diện về cho Client và hiển thị kết quả nhận diện lên màn hình. Quá trình gửi nhận dữ liệu giữa Raspberry Pi và Edge / Cloud Server sử dụng Internet (Có thể sử dụng một trong các giao thức như: HTTP, MQTT, RPC ...).



Hình 1. Tổng quan mô hình Lab 5

3. Viết thêm hàm để đo tổng thời gian nhận diện 1 hình ảnh. Thời gian được tính từ lúc Raspberry Pi gửi hình ảnh cho Edge / Cloud Server cho đến khi nhận được kết quả trả về.
4. Tìm hiểu các file trong mã nguồn được cung cấp. Trình bày các quy trình, bước xử lý, mô tả về những tìm hiểu của mình về mã nguồn trên.

2. Yêu cầu nộp bài

- Sinh viên tìm hiểu và thực hành theo hướng dẫn. Thực hiện **nhóm**.
- Sinh viên báo cáo kết quả thực hiện và nộp bài bằng file. Trong đó:
 - Trình bày chi tiết quá trình thực hành và trả lời các câu hỏi nếu có (kèm theo các ảnh chụp màn hình tương ứng).
 - Giải thích các kết quả đạt được.
 - Tải mẫu báo cáo thực hành và trình bày theo mẫu được cung cấp.

Nén tất cả các file và đặt tên file theo định dạng theo mẫu:

NhomY-LabX_MSSV1_MSSV2

Ví dụ: Nhom1-Lab05_25520001_25520002

- Nộp báo cáo trên theo thời gian đã thống nhất tại website môn học.
- Các bài nộp không tuân theo yêu cầu sẽ **KHÔNG** được chấm điểm.

HẾT

Chúc các bạn hoàn thành tốt bài thực hành!