

CUỘC THI AI CHALLENGE NHẬN DIỆN THƯƠNG HIỆU – LOGO TRONG ẢNH

NHÓM 02 - CS114.L11.KHCL

PHẠM ANH KHOA – 19521699

PHẠM MINH LONG – 19521797

NGUYỄN TRẦN PHƯỚC LỘC - 19521764

Link Github: <https://github.com/khoaphamj1505/CS114.L11.KHCL>

Thành viên trong nhóm

- Ảnh các thành viên trong nhóm

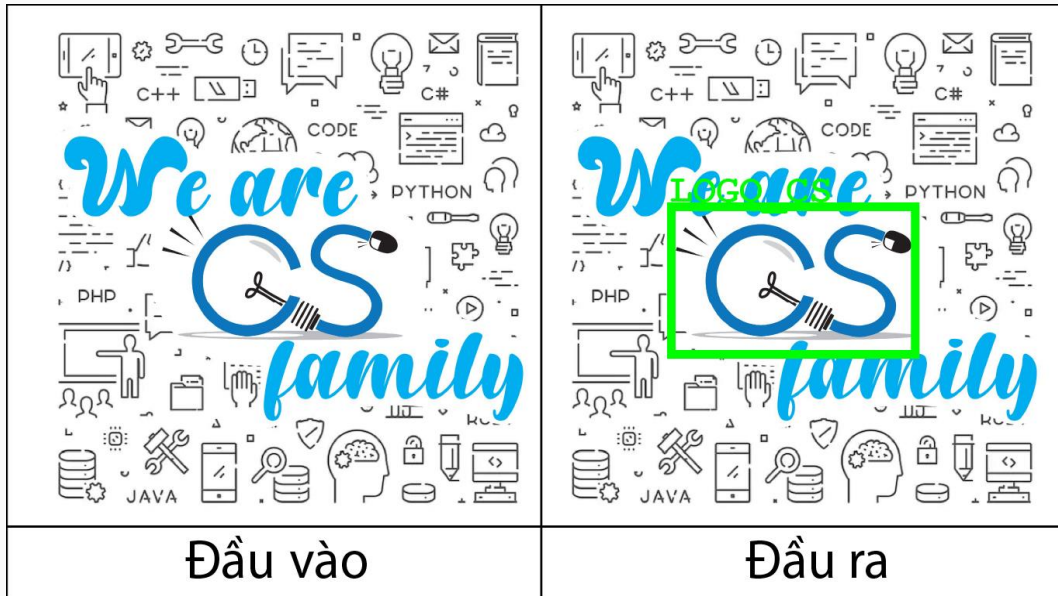


Bài toán

- Với mong muốn tạo sân chơi cho các bạn sinh viên cũng như cơ hội cho doanh nghiệp được tiếp cận và ươm mầm nhân lực chất lượng cao thuộc lĩnh vực AI, Liên chi Hội khoa Khoa học Máy tính cùng câu lạc bộ Trí tuệ nhân tạo (CS-UIT AI Club) đã có kế hoạch tổ chức cuộc thi UIT AI Challenge với chủ đề “Nhận diện thương hiệu – Logo trong ảnh”.
- Đây là bài toán có ứng dụng thực tế cao, đặc biệt cho các công tác quản lý và nhận diện thương hiệu. Với các lớp logo được sử dụng đó là: UIT, Hội sinh viên, khoa Khoa học Máy tính, khoa Công nghệ Phần mềm, khoa Kỹ thuật Máy tính, khoa Khoa học và Kỹ thuật Thông tin.

Bài toán

- Với đầu ra, đầu vào của bài toán:



Mô tả dữ liệu

- Bộ dữ liệu ban tổ chức cung cấp bao gồm hình ảnh và tệp txt chứa vị trí của logo có trong hình
- Số lượng: 3000 tấm ảnh
- Chia data thành 2 tập: tập train 80% và tập valid 20%

TIỀN XỬ LÝ DỮ LIỆU

- Sử dụng LabelImg để gán nhãn ảnh
- VD: ảnh đã được label sẽ có file txt chứa vị trí logo kèm theo



TRAIN_0 - Notepad

File Edit Format View Help

```
0 0.698270 0.086458 0.027682 0.037500
1 0.668512 0.086458 0.023529 0.033333
1 0.369896 0.363021 0.008997 0.013542
1 0.612111 0.409896 0.011765 0.013542
3 0.726644 0.089583 0.023529 0.035417
```

Phương pháp đề xuất

- Sử dụng Yolov4.
- Yolo là một mô hình mạng CNN cho việc phát hiện, nhận dạng, phân loại đối tượng. Yolo được tạo ra từ việc kết hợp giữa các convolutional layers và connected layers. Trong đó các convolutional layers sẽ trích xuất ra các feature của ảnh, còn full-connected layers sẽ dự đoán ra xác suất đó và tọa độ của đối tượng.

Cài đặt chương trình

- Ngôn ngữ sử dụng: Python
- Công cụ thực hiện: VScode, Google colab
- Thư viện sử dụng: math, os, numpy, glob2

Cài đặt chương trình

- Bước 1: Tải mã nguồn YOLOv4 về drive
- Bước 2: Tải file pre-trained
- Bước 3: Tạo file obj.names và obj.data
- Bước 4: Chỉnh sửa tệp MAKEFILE
- Bước 5: Chỉnh sửa tệp config trong darknet
- Bước 6: Bắt đầu train
- Bước 7: Quá trình predict

DEMO

- Dự đoán Logo có trong hình ảnh



DEMO

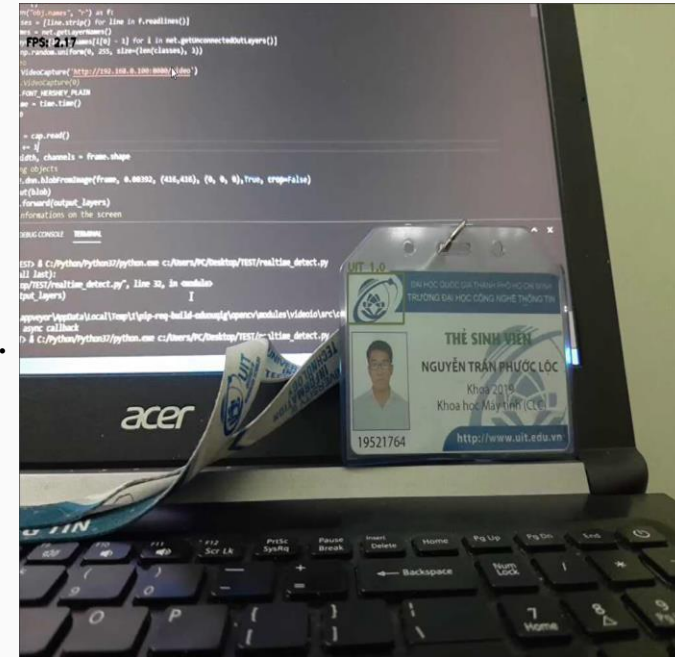
- Dự đoán Logo bằng camera laptop

-Do không muốn bị giới hạn ở việc chỉ predict mỗi hình ảnh, nhóm chúng em đã phát triển model lên thành real-time predict với camera của laptop.



DEMO

- Dự đoán logo bằng camera điện thoại
 - Nhận thấy sự thiếu bất tiện của việc camera không thể tự do xoay chuyển và khó khăn khi di chuyển camera của laptop, nhóm chúng em đã tiếp tục phát triển để model có thể dự đoán được thông qua camera điện thoại thông qua kết nối không dây giữa hai thiết bị.



ĐÁNH GIÁ CHƯƠNG TRÌNH

Độ chính xác của model (mAP) được tính bằng hàm IOU.

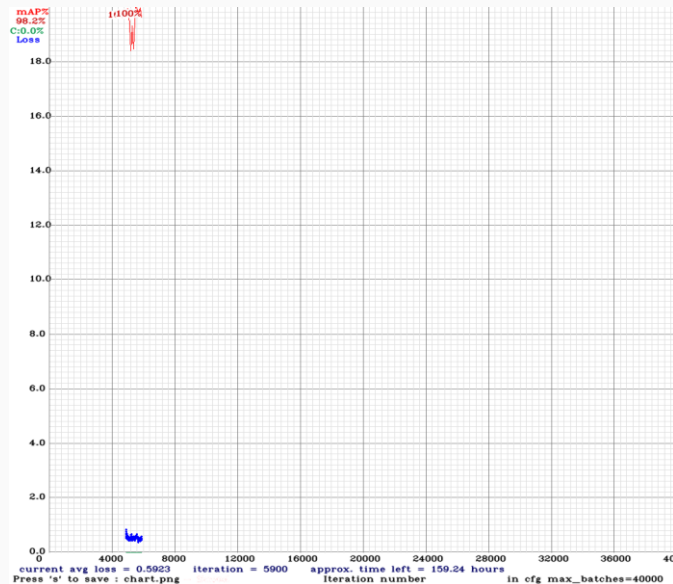
$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

Để theo dõi mAP trong quá trình training, nhóm đã thêm vào ‘– map’ sau dòng code train.

Vd:

```
!./darknet detector train obj.data cfg/yolov4.cfg yolov4.  
conv.137 -dont_show -map”
```

Sau khoảng 5900 iteration, model của chúng em đạt điểm mAP trung bình lên đến 98,2% với avg loss hiện tại là 0.5923 và không có xu hướng giảm nữa nên chúng em quyết định dừng training.



Kết quả cuộc thi

Sau cùng cuộc thi, nhóm chúng em có đạt được một giải phụ là đội “Khởi động ấn tượng nhất” với điểm số cao nhất ở lần nộp đầu tiên.



Kết luận

- Bài toán nhận diện sử dụng kỹ thuật deep learning (CNN).
- thông qua cuộc thi UIT AI Challenge 2020 nhóm chúng em đã nâng cao được sự hiểu biết hơn về cấu trúc của các cuộc thi về AI và quan trọng hơn là học hỏi thêm được những kiến thức bổ ích về Yolo và cách sử dụng cơ bản
- Cuộc thi UIT AI Challenge đã cung cấp thêm nguồn kiến thức phong phú dưới dạng hệ thống tương đối hoàn chỉnh về khái niệm, cách xây dựng một mô hình máy học cũng như ứng dụng liên quan đến thực tiễn đời sống mà chúng em có thể sử dụng cho các bài toán về sau.