

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**  
**KHOA HỆ THỐNG THÔNG TIN**



**ĐỀ TÀI: PHÁT HIỆN VÀ LÀM MỜ BIỂN SỐ XE**  
**MÃ LỚP: CS114.N11.KHCL**

**GIẢNG VIÊN HƯỚNG DẪN: PHẠM NGUYỄN TRƯỜNG AN**

**LÊ ĐÌNH DUY**

**SINH VIÊN THỰC HIỆN:**      **20521925 – TRẦN DUY THANH**  
                                         **20521576 – TRẦN VĂN LONG**  
                                         **20522114 – NGUYỄN ANH TUẤN**

*TP. Hồ Chí Minh, tháng 1 năm 2023*

# Mục lục

|                                                  |    |
|--------------------------------------------------|----|
| <b>I. Tổng quan</b>                              | 3  |
| I.1. Giới thiệu về đề án                         | 3  |
| I.2. Mục tiêu của đề án                          | 3  |
| I.3. Input và Output của bài toán                | 4  |
| I.4. Các bước tiến hành                          | 4  |
| <b>II. Mô tả và xây dựng bộ dữ liệu</b>          | 4  |
| II.1. Mô tả bộ dữ liệu                           | 4  |
| II.2. Xử lý bộ dữ liệu                           | 5  |
| II.3. Gán nhãn dữ liệu                           | 6  |
| II.4. Thống kê dữ liệu                           | 7  |
| <b>III. Thuật toán và huấn luyện cho mô hình</b> | 8  |
| III.1. Thuật toán lựa chọn                       | 8  |
| III.2. Training các mô hình                      | 9  |
| 1.YoLoV4                                         | 9  |
| 2.YoLoV5                                         | 12 |
| <b>IV. Kết quả và đánh giá mô hình</b>           | 14 |
| IV.1. Thang đo                                   | 14 |
| IV.2. Kết quả của model                          | 16 |
| IV.3. Kết quả tốt                                | 20 |
| IV.4. Kết quả chưa tốt                           | 21 |
| IV.5. Đánh giá video                             | 23 |
| <b>V. Cải thiện dataset</b>                      | 24 |
| <b>VI. Hướng phát triển</b>                      | 24 |

# I. Tổng quan

## I.1. Giới thiệu về đề án

Trong đời sống hiện đại, xe cộ đóng vai trò rất quan trọng trong việc di chuyển và vận chuyển hàng hóa. Tuy nhiên, việc quản lý xe cộ cũng đặt ra nhiều thách thức cho các cơ quan chức năng. Biển số xe là một yếu tố quan trọng trong việc giám sát giao thông đường bộ và là cách để xác định chính xác các phương tiện. Tuy nhiên, thông tin về biển số xe cũng là thông tin nhạy cảm về sự riêng tư của chủ xe. Nếu thông tin này bị lộ ra ngoài, chủ xe có thể bị mất quyền lợi và an toàn. Vì vậy, cần có những biện pháp bảo vệ thông tin biển số xe, đảm bảo tính riêng tư và bảo mật.

Với sự phát triển của công nghệ thông tin và truyền thông, việc đăng tải hình ảnh chứa thông tin biển số xe trở nên phổ biến và dễ dàng hơn bao giờ hết. Điều này làm tăng nguy cơ rò rỉ thông tin và làm mất quyền riêng tư của chủ xe, đặc biệt là trong các trường hợp tội phạm sử dụng thông tin này để thực hiện những hoạt động bất hợp pháp. Do đó, việc phát triển hệ thống nhận diện và làm mờ biển số xe giúp ngăn chặn việc đăng tải hình ảnh biển số xe trên mạng xã hội và giảm thiểu nguy cơ lộ thông tin cá nhân.

Ngoài ra, việc áp dụng các phương pháp tiếp cận nhận diện và xử lý hình ảnh để phát hiện và nhận diện biển số xe, và sau đó sử dụng thuật toán làm mờ để che giấu phần thông tin của biển số xe cũng là một bước tiến trong công nghệ nhận diện hình ảnh và bảo vệ thông tin cá nhân. Vì vậy, đề án này có thể được áp dụng rộng rãi trong

- Các lĩnh vực như giao thông, an ninh,
- Bảo vệ thông tin cá nhân và các lĩnh vực khác liên quan đến hình ảnh và thông tin.

## I.2. Mục tiêu của đề án

Mục tiêu của đề án là thiết kế và xây dựng hệ thống nhận diện và làm mờ biển số xe. Hệ thống này sẽ sử dụng các phương pháp tiếp cận nhận diện và xử lý hình ảnh để phát hiện và nhận diện biển số xe trên đường. Sau đó, hệ thống sẽ sử dụng thuật toán làm mờ để che giấu phần thông tin của biển số xe và bảo vệ sự riêng tư của chủ xe.

### I.3. Input và Output của bài toán

- Input: đoạn video .mp4 chứa các khung hình có chứa biển số xe cần được nhận diện và làm mờ.
- Output: đoạn video đã được xử lý sao cho các khung hình chứa biển số xe đã được nhận diện và làm mờ

### I.4. Các bước tiến hành

Để thực hiện đề án, nhóm đã thực hiện các bước như sau:

- Thu thập dữ liệu: Nhóm đã thu thập một số lượng lớn hình ảnh và video chứa các biển số xe để huấn luyện và kiểm thử hệ thống.
- Chuẩn bị dữ liệu: Dữ liệu thu thập được được chia thành các tập dữ liệu để huấn luyện và kiểm thử mô hình.
- Huấn luyện mô hình: Chúng tôi đã sử dụng một mô hình mạng nơ-ron tích chập để huấn luyện cho nhiệm vụ nhận diện biển số xe trên hình ảnh hoặc video.
- Kiểm thử và tinh chỉnh: Sau khi huấn luyện, chúng tôi đã tiến hành kiểm thử và tinh chỉnh mô hình để đạt được độ chính xác cao nhất.
- Làm mờ biển số xe: Sau khi xác định được vị trí của biển số xe trên hình ảnh hoặc video, chúng tôi đã sử dụng thuật toán làm mờ để che đi các ký tự trên biển số xe.

## II. Mô tả và xây dựng bộ dữ liệu

### II.1. Mô tả bộ dữ liệu

- Video tự quay: 25 video, trung bình mỗi video có độ dài 2 phút.
- Hình ảnh lấy trên Mĩ AI,

- Video tự quay: Khu vực Dĩ an – Bình Dương như quốc lộ 1K, các con đường ở các quận của Hồ Chí Minh như quận Thủ Đức(Võ Văn Ngân), Quận Bình Thạnh(QL25-Xa lộ Hà Nội) , bãi gửi xe của Đại học Sư phạm kỹ thuật,...

Cách thu thập:

- Sử dụng điện thoại cá nhân quay với lượng video 720p
- Khi tiến hành đi quay, ngồi phía sau xe cầm điện thoại để quay với mục đích có được thông tin của biển số xe đang lưu thông hoặc đang dừng, đỗ.
- Nhóm tiến hành quay vào lúc trời sáng trưa, chiều và gần tối.

Lấy từ Youtube và mì AI:

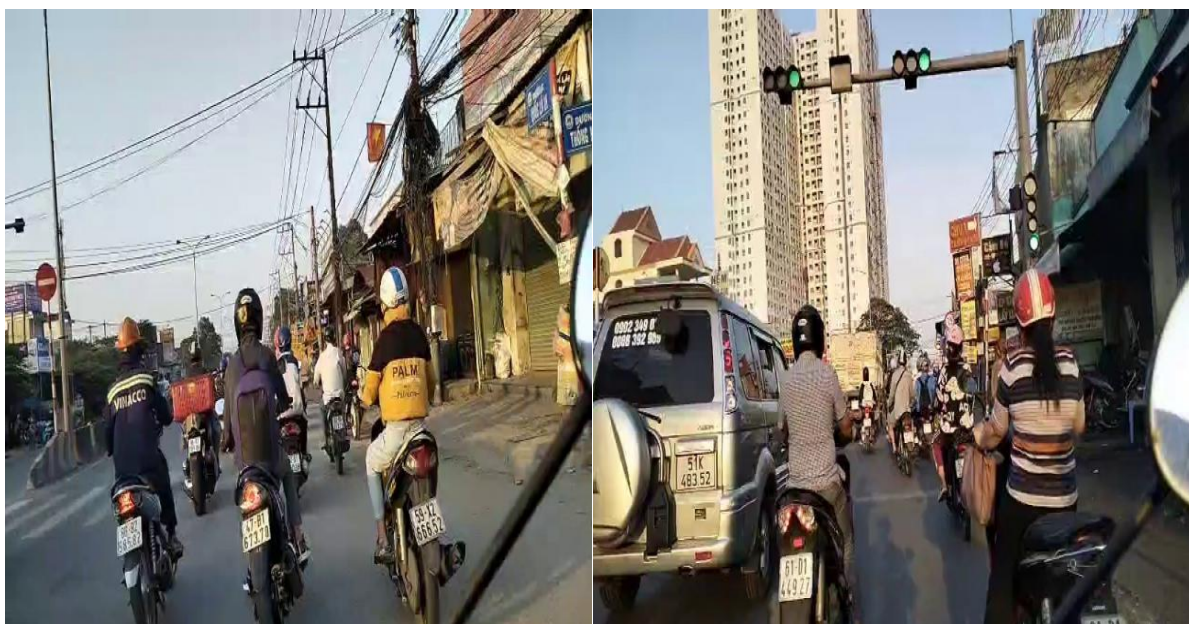
- Hình ảnh lấy trên Mì AI (216 tấm ảnh) và youtube kênh <https://www.youtube.com/@vochongsuaxe> (khi cắt ra thì được 129 tấm)
- Lý do: Do trong lúc thu thập thủ công, bộ dữ liệu thu thập còn thiếu các biển loại khác như biển vàng, biển đỏ và biển ngoại giao, biển xanh khó. Trong lúc thu thập, dữ liệu bị nhiễu nên nhóm tiến hành thu thập dữ liệu do MìAI cung cấp

## II.2. Xử lý bộ dữ liệu

Sau khi có được dữ liệu từ video, nhóm tiến hành cắt ghép ảnh từ video với 1 frame/second- tương đương với 1 giây là 1 hình ảnh. Sau đó các hình ảnh bị mờ không thấy rõ biển số sẽ được xóa bỏ. Và giữ lại những tấm thấy được biển số để gán nhãn.

\*Một số hình ảnh trong tập dữ liệu





### II.3. Gán nhãn dữ liệu

Nhóm sử dụng Roboflow - là một framework dành cho nhà phát triển Thị giác Máy tính để thu thập dữ liệu tốt hơn, xử lý trước và các kỹ thuật đào tạo mô hình .

Đường dẫn truy cập: <https://roboflow.com/>

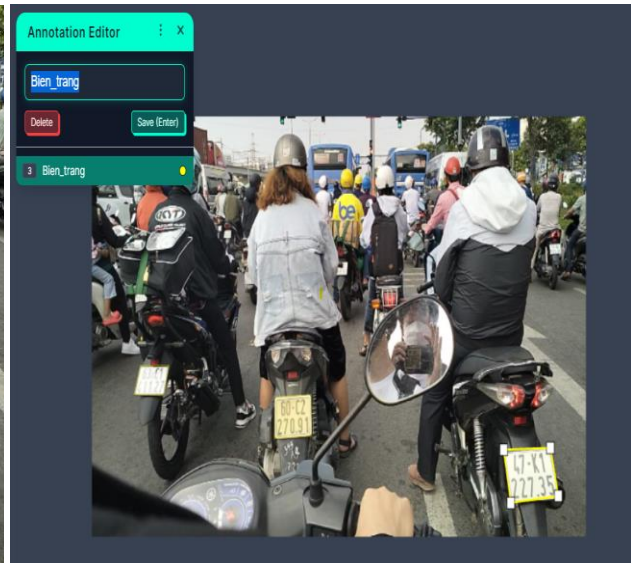
- Nhóm tiến hành gán nhãn với 5 loại biển số của Việt Nam:

Biển đỏ, biển ngoại giao, biển trắng, biển xanh, biển vàng

\*Cách gán nhãn:

- Trong hình ảnh các biển số thấy được thông tin(ngay cả khi phóng to) sẽ dùng các bouding box để bao sát biển. Sau đó tiến hành gán nhãn cho biển số

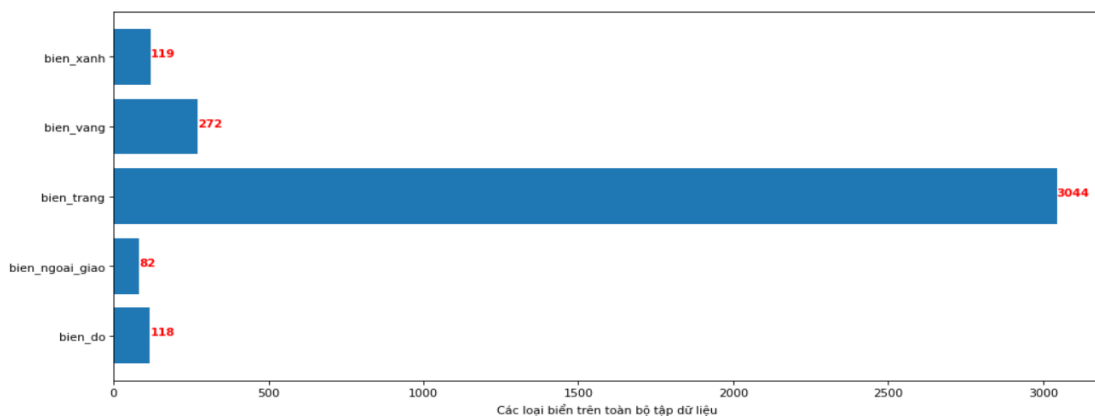


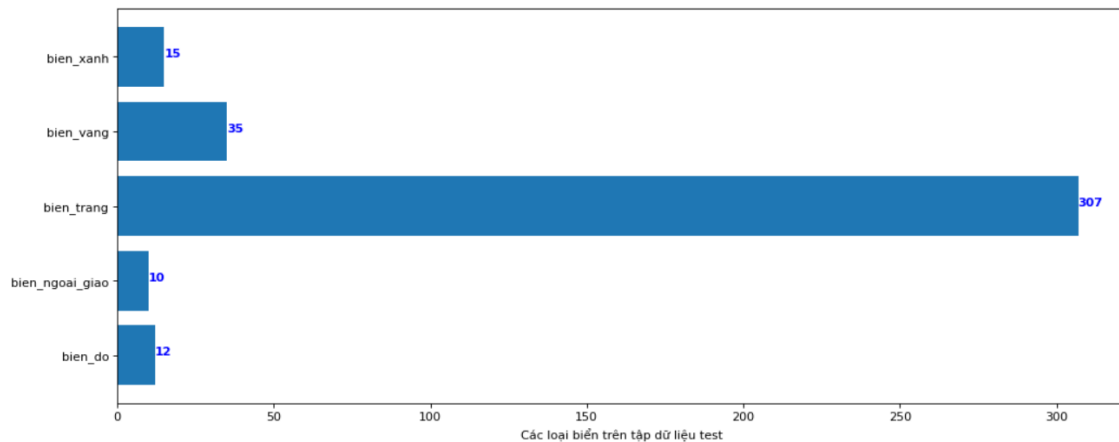
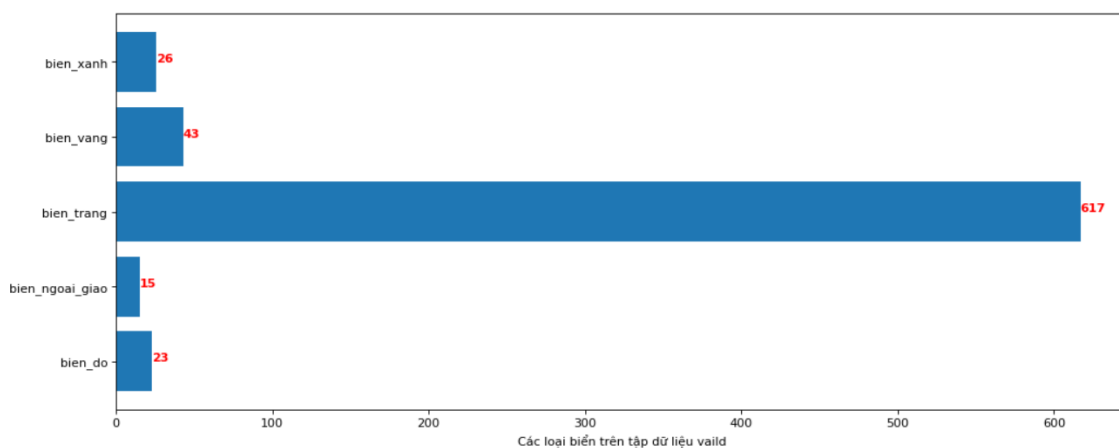
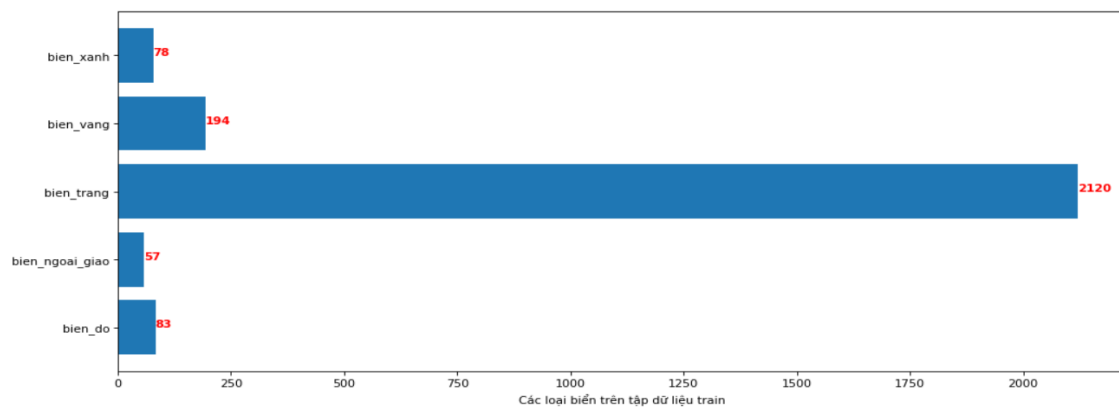


## II.4. Thống kê dữ liệu

Tập dữ liệu sau khi gán nhãn có 2131 bức ảnh. Nhóm tiến hành chia tập dữ liệu lần lượt thành các tập train, valid, test với tỷ lệ 70%, 20%, 10%. Sau khi chia tập dữ liệu huấn luyện train 1493 ảnh, tập đánh giá mô hình valid 414 ảnh, tập kiểm thử test 224 ảnh.

Với mỗi bức ảnh có đồng thời một đến nhiều loại biển trong tấm ảnh. Các loại biển phân bố trong các tập của dữ liệu như sau:





Link dataset của nhóm: <https://app.roboflow.com/uit-5rhkt/biensoxe-7geys/overview>

## III. Thuật toán và huấn luyện cho mô hình

### III.1. Thuật toán lựa chọn

Sau khi xem xét nhóm đã quyết định lựa chọn 2 mô hình để giải quyết bài toán mà nhóm đưa ra là: YoLoV4 với YoLoV5

Một số tiêu chí mà nhóm chọn 2 mô hình trên:



1. Tính năng: Cả YoloV4 và YoloV5 đều có nhiều tính năng mới và được cải tiến để tăng cường độ chính xác và tốc độ xử lý.
2. Hiệu năng: Cả YoloV4 và YoloV5 đều có hiệu năng cao hơn so với nhiều mô hình nhận dạng vật thể khác hiện nay. Nhờ sự cải tiến về kiến trúc mạng và thuật toán tối ưu hóa, cả hai mô hình đều có khả năng xử lý các tập dữ liệu lớn một cách nhanh chóng và hiệu quả.
3. Độ chính xác: Cả YoloV4 và YoloV5 đều đạt được độ chính xác cao trong việc phát hiện và nhận dạng vật thể. Tuy nhiên, YoloV4 có độ chính xác tốt hơn so với YoloV5 trong một số trường hợp đặc biệt, nhưng YoloV5 lại có khả năng phát hiện vật thể nhỏ hơn.
4. Khả năng tùy chỉnh: Cả YoloV4 và YoloV5 đều cho phép người dùng tùy chỉnh mô hình để phù hợp với nhu cầu của họ.  
Điều này cho phép người dùng tối ưu hóa mô hình để đáp ứng các yêu cầu đặc biệt của ứng dụng của mình.

## III.2. Training các mô hình

### 1.YoLoV4

#### a. Giới thiệu

- YoloV4 (You Only Look Once Version 4) là một mô hình mạng nơ-ron sử dụng để nhận dạng vật thể. Nó được phát triển bởi Alexey

Bochkovskiy, Chien-Yao Wang và Hong-Yuan Mark Liao.

- YoloV4 là phiên bản nâng cấp của các phiên bản trước đó của YOLO và là một trong những mô hình phổ biến nhất trong lĩnh vực nhận dạng vật thể.

#### \* Ưu điểm

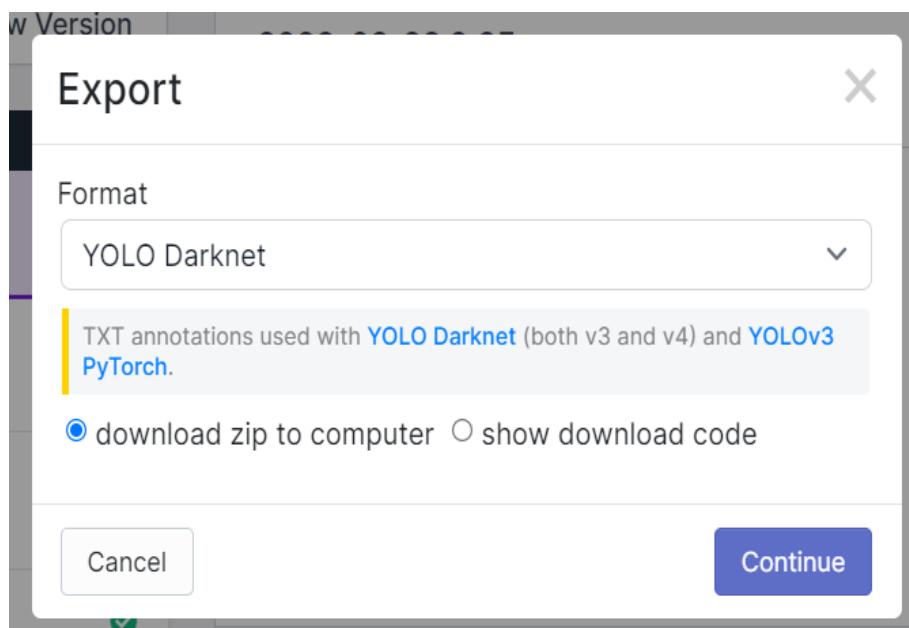
1. Tính nhanh và hiệu quả: YoloV4 có thể xử lý và phân loại các vật thể trong ảnh với tốc độ nhanh hơn so với các mô hình khác.
2. Tính đa dạng: YoloV4 hỗ trợ nhiều loại đối tượng và có thể nhận dạng các vật thể trong nhiều loại ảnh khác nhau.
3. Tính ổn định: YoloV4 cải tiến về độ ổn định so với các phiên bản trước đó của YOLO, đặc biệt là trong việc xử lý các ảnh có nhiễu hoặc độ phân giải thấp.
4. Tính tùy chỉnh cao: YoloV4 cho phép người dùng tùy chỉnh mô hình để phù hợp với các yêu cầu và đặc thù của từng ứng dụng cụ thể.

#### \* Nhược điểm









1. Đòi hỏi cấu hình cao: YoloV4 đòi hỏi một máy tính có cấu hình cao và GPU mạnh để đạt được tốc độ xử lý nhanh nhất.
2. Độ phức tạp: YoloV4 có mức độ phức tạp khá cao

## b. Training model

- Chuẩn bị bộ dữ liệu để tiến hành train: Dùng export trên roboflow để xuất ra file định dạng darknet



File sau khi tải về sẽ có chứa các tập train, valid, test. Trong mỗi tập sẽ có chứa file .txt chứa thông tin vị trí và nhãn của đối tượng cũng như hình ảnh tương ứng với file .txt đó

|                                                                                     |                                          |                  |                  |       |
|-------------------------------------------------------------------------------------|------------------------------------------|------------------|------------------|-------|
|  | 0000_00532_b.jpg.rf.9f6998b9e7c02a70e... | 3/2/2023 2:17 PM | JPG File         | 34 KB |
|  | 0000_00532_b.jpg.rf.9f6998b9e7c02a70e... | 3/2/2023 2:17 PM | Tài liệu văn bản | 1 KB  |
|  | 0000_05696_b.jpg.rf.6d0f057be7438e595... | 3/2/2023 2:17 PM | JPG File         | 33 KB |
|  | 0000_05696_b.jpg.rf.6d0f057be7438e595... | 3/2/2023 2:17 PM | Tài liệu văn bản | 1 KB  |
|  | 0000_06886_b.jpg.rf.c94bd3b370e89951...  | 3/2/2023 2:17 PM | JPG File         | 32 KB |
|  | 0000_06886_b.jpg.rf.c94bd3b370e89951...  | 3/2/2023 2:17 PM | Tài liệu văn bản | 1 KB  |
|  | 0000_08244_b.jpg.rf.1e80ff4d14bfd6d36... | 3/2/2023 2:17 PM | JPG File         | 35 KB |
|  | 0000_08244_b.jpg.rf.1e80ff4d14bfd6d36... | 3/2/2023 2:17 PM | Tài liệu văn bản | 1 KB  |

Chuẩn bị các file cần thiết để training mô hình:

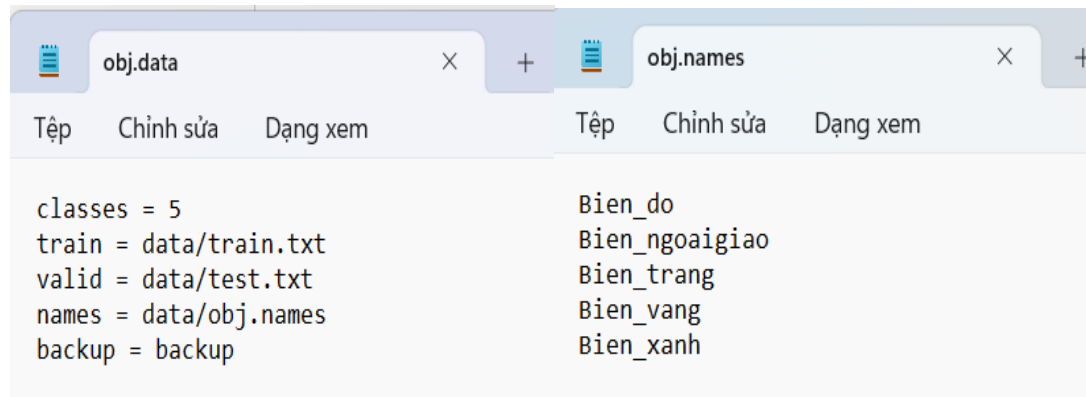
- Tạo 2 file train.txt và test.txt để có thể lấy ra được đường dẫn đến ảnh trong tập train và tập test của tập dữ liệu.

```
data/train/VID20230225100818_mp4-34_jpg.rf.6cab41c5cadfc2d17a7820cd95990794.jpg
data/train/VID20230225150209_mp4-217_jpg.rf.f3e3f875cf694494558d70cda059e955.jpg
data/train/MiConv-com_tai-xuong-13_png.rf.dbb919122bd329ca842c9d924aef83e8.jpg
data/train/bongoaigiao_augmentated_5_65_jpg.rf.aa067f85213ed24eec00f98bcd242fb.jpg
data/train/VID20230225150650_mp4-47_jpg.rf.7caa6d8c1c20858c89c6076237a3d211.jpg
data/train/VID20230225100400_mp4-105_jpg.rf.522b407b350995625c973db62d3ec0f3.jpg
data/train/VID20230225102626_mp4-105_jpg.rf.2275e70bffa84bebf95909d17c153359.jpg
data/train/VID20230225100818_mp4-7_jpg.rf.77446017cde1393342e4f307db36f951.jpg
data/train/video2_855_jpg.rf.1bd13bf3d90ce788a2d8167400f7438e.jpg
data/train/VID20230225102626_mp4-56_jpg.rf.fc7f9b3c9e31812086cc0b19bca3d499.jpg
data/train/VID20230222172933_mp4-39_jpg.rf.7868bd45a241a955560ac48e69359811.jpg
data/train/VID20230225100818_mp4-19_jpg.rf.1183106a13ac63f611a0947c3fb35b2c.jpg
```

*file train.txt chứa đường dẫn*

- Tạo 2 file obj.data và obj.name. Trong đó:
- + obj.data chứa các thông tin cấu hình cho việc huấn luyện và phát hiện đối tượng.

+obj.name chứa danh sách các tên lớp đối tượng được sử dụng để phát hiện.



- Tải file yolov4-custom.cfg theo đường dẫn

<https://github.com/AlexeyAB/darknet/blob/master/cfg/yolov4-custom.cfg>

Nhóm tiến hành chỉnh các thông số cho phù hợp với tập dữ liệu:

```
batch=64
subdivisions=32
width=416
height=416
max_batches = 10000
policy=steps
steps=8000,9000
class = 5
filters=30
width=416
height=416
```

- Tải pretrained weight để giúp cho mô hình train nhanh và hiệu quả hơn **yolov4.conv.137**

```
!wget https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v3_optimal/yolov4.conv.137
```

Sau khi có được các file cần thiết tiến hành tải darknet và tạo các đường dẫn cần thiết cho việc train.

\* Nhận xét về việc training:

- Thời gian train mô hình lâu để có thể tạo được file trọng số thích hợp
- Khi train tới epoch 2000 nhóm tiến hành dừng việc train vì thời gian thấy loss bão hòa, không giảm quá nhiều

## 2. YoLoV5

### a. Giới thiệu

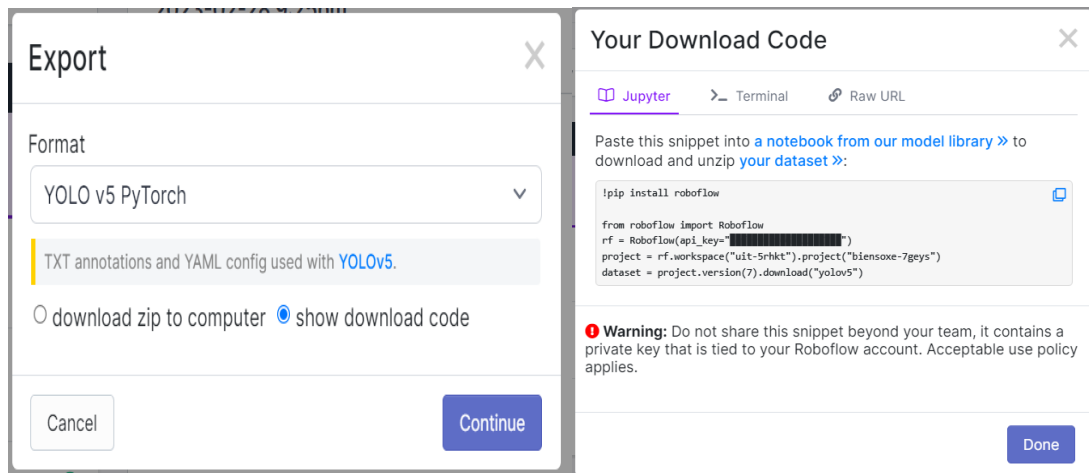
- YoLoV5 là một mô hình phát hiện đối tượng nhanh và hiệu quả được phát triển bởi Ultralytics. Nó là phiên bản nâng cấp của YoloV4 với các cải tiến về tốc độ, độ chính xác và khả năng tùy chỉnh.

Một số tính năng của:

1. Tăng tốc phát hiện đối tượng: Với kiến trúc đơn giản hóa và sử dụng các kỹ thuật tối ưu mới, YoLoV5 có thể phát hiện đối tượng nhanh hơn so với phiên bản trước đó.
2. Khả năng tùy chỉnh linh hoạt: YoLoV5 cho phép người dùng tùy chỉnh mô hình để phù hợp với các nhu cầu riêng của họ, bao gồm kích thước đầu vào, kiến trúc mạng, số lượng lớp đối tượng, v.v.
3. Độ chính xác cao: YoLoV5 đạt được kết quả tốt trong các bài kiểm tra thực tế về phát hiện đối tượng, với độ chính xác và độ nhạy cao.
4. Dễ sử dụng: YOLOv5 được viết bằng Python và có các công cụ hỗ trợ sử dụng, cho phép người dùng dễ dàng triển khai và sử dụng nó.

### b. Training model

- Đầu tiên nhóm tiến hành export từ roboflow với mã code trên jupyter



- Để train mô hình nhóm thiết lập các trọng số sau:

```
!python train.py --img 640 --batch 16 --epochs 200 --data {dataset.location}/data.yaml --weights yolov5s.pt --cache
```

Trong đó:

--img 640: Kích thước ảnh đầu vào được sử dụng trong quá trình huấn luyện.

--batch 16: Số lượng ảnh được sử dụng trong mỗi batch huấn luyện.

--epochs 200: Số lượng epochs (vòng lặp huấn luyện) để huấn luyện mô hình.

--weights yolov5s.pt: Đường dẫn đến trọng số được sử dụng để khởi tạo mô hình huấn luyện. Trong trường hợp này, nhóm sử dụng trọng số của YoLov5s. Lý do YoLov5s là một phiên bản nhỏ gọn và hiệu quả của YoLov5, phù hợp cho các ứng dụng thời gian thực hoặc các thiết bị có tài nguyên hạn chế, nhưng vẫn đảm bảo độ chính xác cao.

| Model      | AP <sup>val</sup> | AP <sup>test</sup> | AP <sub>50</sub> | Speed <sub>GPU</sub> | FPS <sub>GPU</sub> | params | FLOPS  |
|------------|-------------------|--------------------|------------------|----------------------|--------------------|--------|--------|
| YOLOv5s    | 36.6              | 36.6               | 55.8             | 2.1ms                | 476                | 7.5M   | 13.2B  |
| YOLOv5m    | 43.4              | 43.4               | 62.4             | 3.0ms                | 333                | 21.8M  | 39.4B  |
| YOLOv5l    | 46.6              | 46.7               | 65.4             | 3.9ms                | 256                | 47.8M  | 88.1B  |
| YOLOv5x    | 48.4              | 48.4               | 66.9             | 6.1ms                | 164                | 89.0M  | 166.4B |
| YOLOv3-SPP | 45.6              | 45.5               | 65.2             | 4.5ms                | 222                | 63.0M  | 118.0B |

\*Nhận xét về việc training:

Mô hình mất **2 tiếng** để hoàn thành việc học trên tập train.

```

200 epochs completed in 1.598 hours.
Optimizer stripped from runs/train/exp/weights/last.pt, 14.5MB
Optimizer stripped from runs/train/exp/weights/best.pt, 14.5MB

Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model summary: 213 layers, 7023610 parameters, 0 gradients, 15.8 GFLOPs

```

| Class          | Images | Labels | P     | R     | mAP@.5 | mAP@.75 | mAP@.5:.95 |
|----------------|--------|--------|-------|-------|--------|---------|------------|
| all            | 414    | 724    | 0.788 | 0.848 | 0.848  | 0.76    | 0.618      |
| Bien_do        | 414    | 23     | 0.919 | 0.985 | 0.973  | 0.917   | 0.726      |
| Bien ngoaigiao | 414    | 15     | 0.936 | 0.978 | 0.982  | 0.878   | 0.668      |
| Bien_trang     | 414    | 617    | 0.644 | 0.846 | 0.746  | 0.69    | 0.575      |
| Bien_vang      | 414    | 43     | 0.664 | 0.767 | 0.734  | 0.611   | 0.529      |
| Bien_xanh      | 414    | 26     | 0.776 | 0.666 | 0.802  | 0.706   | 0.594      |

```

Results saved to runs/train/exp

```

*Kết quả sau khi chạy 200 epoch*

## IV. Kết quả và đánh giá mô hình

### IV.1. Thang đo

#### a. IOU (Intersection over Union)

Chỉ ra độ khớp giữa bounding box được mô hình dự đoán và ground truth box do con người gán sẵn. Với mỗi ground truth box ta sẽ tiến hành tính IoU với tất cả các bounding box mà mô hình dự đoán.

Khi có được IOU, ta sẽ chọn một threshold để phân loại bounding box thành 3 loại: **TP, FP, FN**.

- **TP** (true positive): Bounding box nào có IoU với 1 ground truth box cao nhất trong tất cả các giá trị IoU của ground truth box đó, cao hơn threshold thì sẽ được tính là TP.
- **FP** (false positive): Bounding box không phải TP sẽ là FP.
- **FN** (false negative): ground truth box nào không có 1 bounding box được tính là TP giao với nó sẽ được tính là FN.
- **TN** (true negative): không có ý nghĩa đối với bài toán object detection. TN mang ý nghĩa những vị trí mà mô hình cho là nền và bên ground truth cũng là nền.

#### b. Precision

Là tỷ lệ trường hợp bounding box có  $\text{IoU} \geq \text{threshold}$  trong các bounding box được dự đoán.

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{TP}{\text{all detections}}$$



Precision có thể không giảm cùng với Recall. Định nghĩa về Precision cho thấy việc giảm threshold có thể làm tăng mẫu số, bằng cách tăng số lượng kết quả trả về. Nếu threshold đang chọn được đặt quá cao, giảm threshold có thể khiến tất cả các kết quả mới là true positive, điều này sẽ làm tăng độ chính xác. Nếu threshold đang chọn phù hợp hoặc quá thấp, việc tiếp tục hạ thấp threshold sẽ khiến false positive tăng, làm giảm độ chính xác.

### c. Recall

Đo lường tỷ lệ bounding box được dự đoán có  $\text{IoU} \geq \text{threshold}$  trên tổng số ground truth box.

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{TP}{\text{all ground truths}}$$

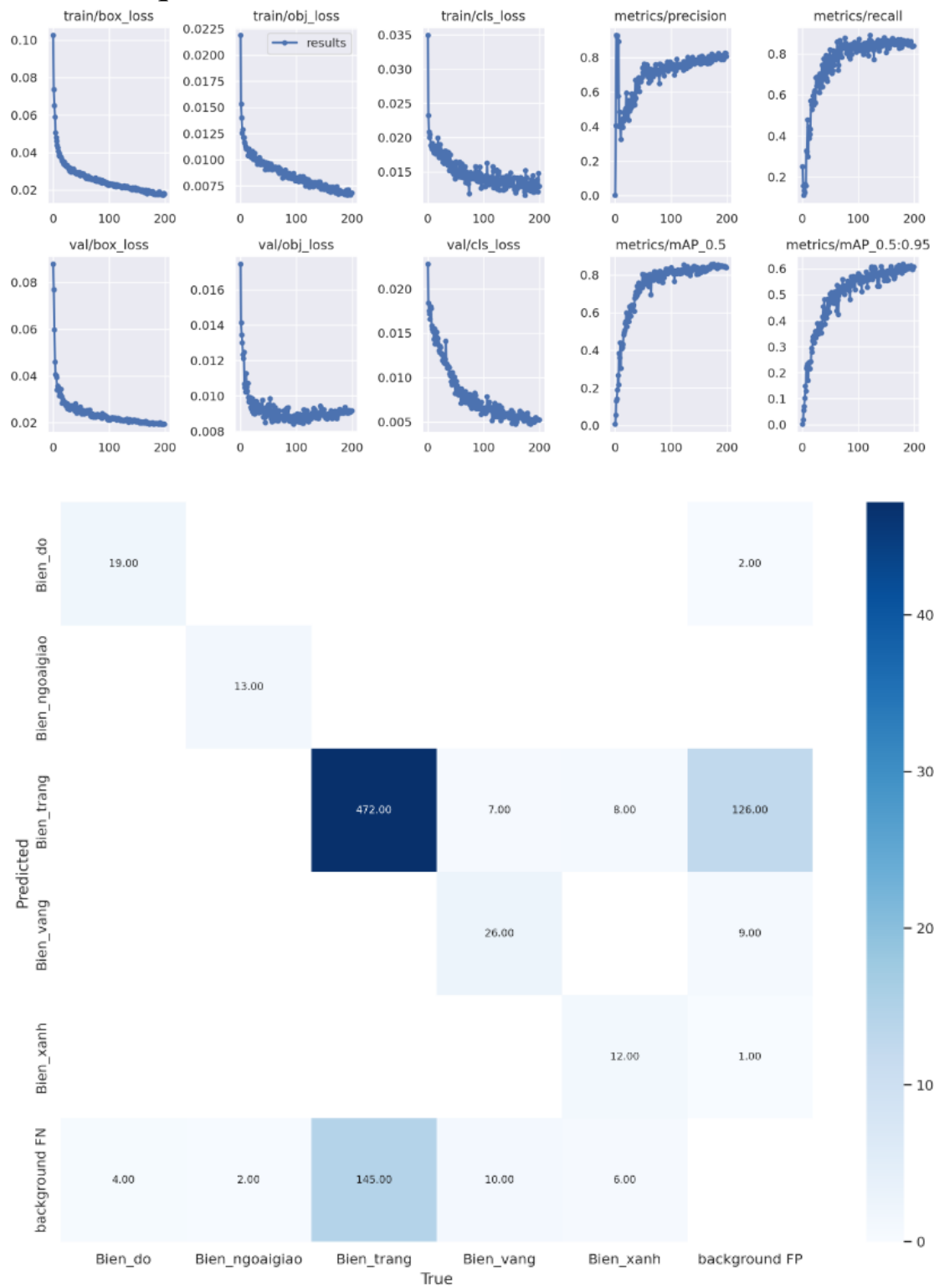
Trong đó,  $TP + FN$  không phụ thuộc vào threshold. Điều này có nghĩa việc giảm threshold có thể tăng Recall bằng cách tăng số lượng kết quả true positive. Ngoài ra, việc hạ thấp threshold có thể không thay đổi Recall, trong khi Precision lại dao động.

### d. mAP (Mean Average Precision)

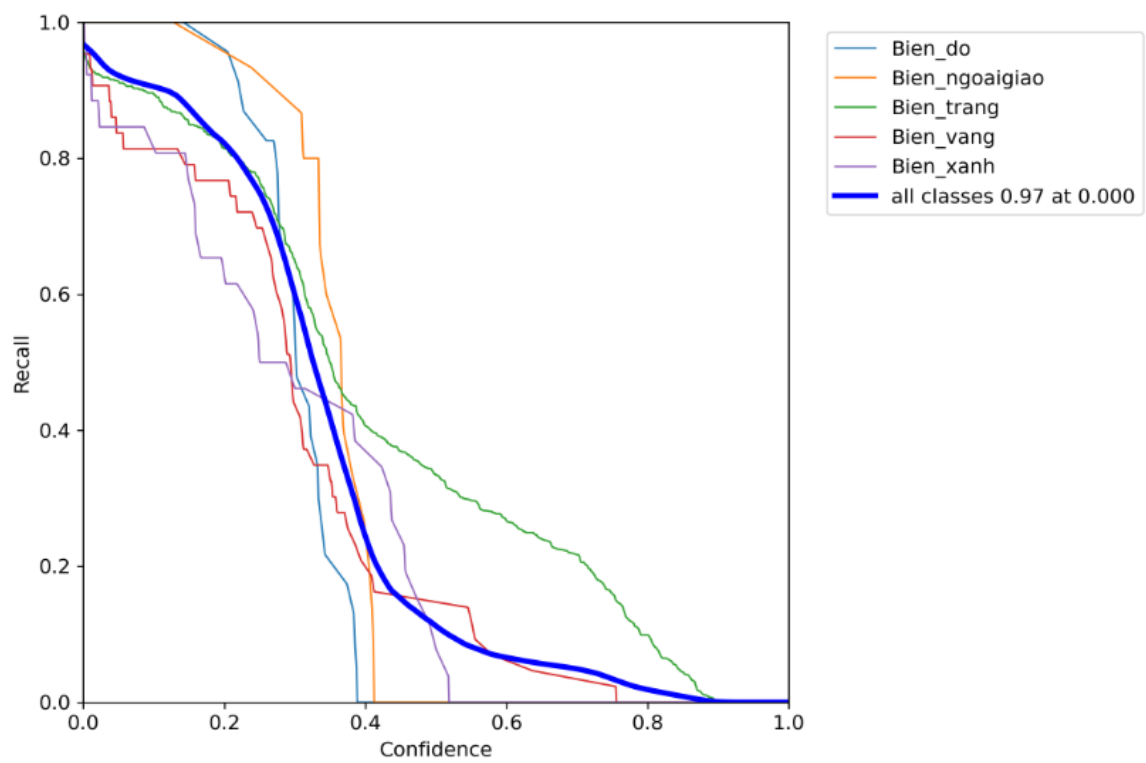
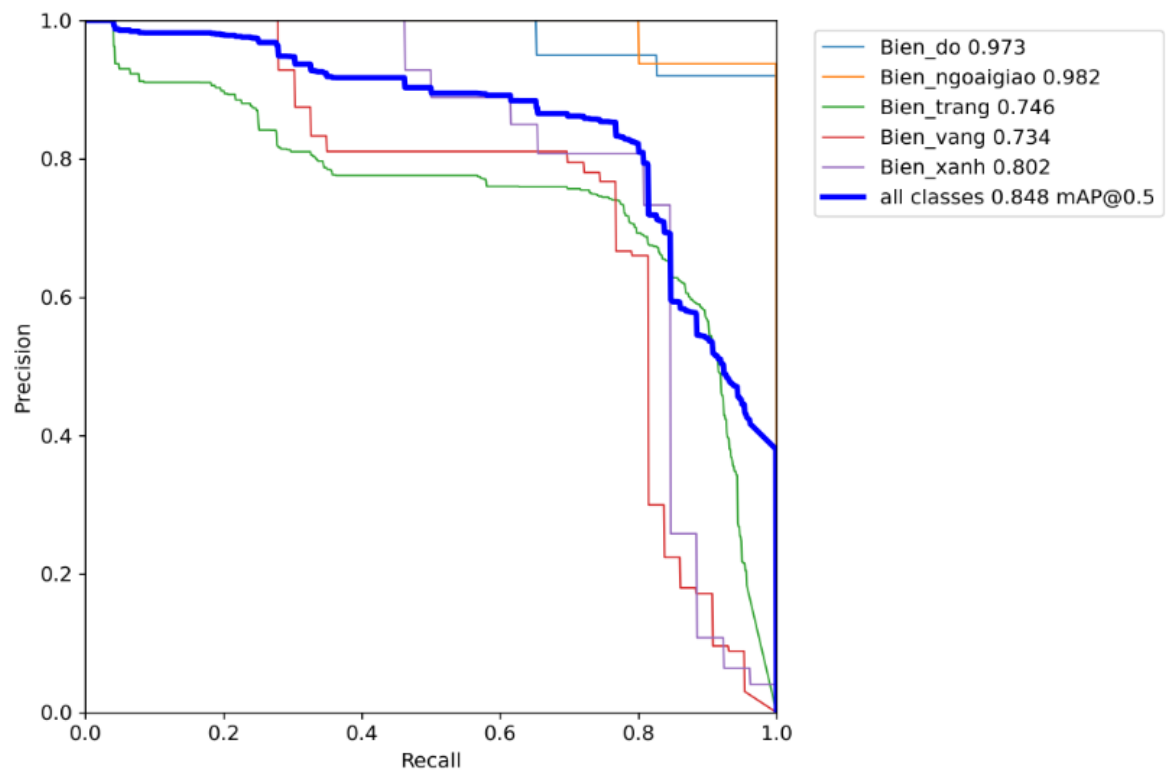
Là một chỉ số tổng hợp để đánh giá chất lượng của mô hình nhận dạng đối tượng. Nó tính toán trung bình của các giá trị precision (độ chính xác) tương ứng với các mức độ recall (tỉ lệ phát hiện) khác nhau của mô hình. Giá trị precision là số lượng đối tượng được phát hiện đúng chia cho tổng số đối tượng được phát hiện, còn giá trị recall là số lượng đối tượng được phát hiện đúng chia cho tổng số đối tượng trong tập dữ liệu.

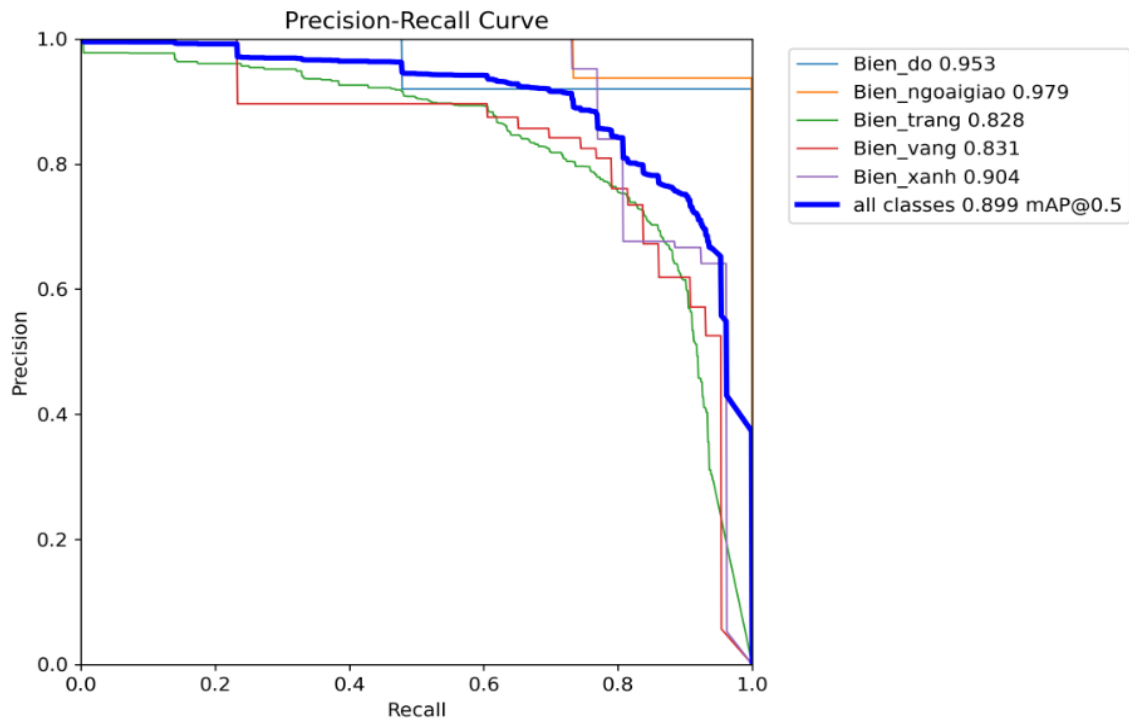
- $\text{mAP}@0.5$  là giá trị trung bình của precision tương ứng với các mức recall khác nhau khi ngưỡng IoU được cố định là 0.5. Nó chỉ đánh giá chất lượng của mô hình nhận dạng đối tượng ở mức độ IoU nhất định, và không cho biết chất lượng của mô hình ở các ngưỡng IoU khác.
- $\text{mAP}@0.5:0.95$  tính trung bình của các giá trị precision tương ứng với các ngưỡng IoU khác nhau từ 0.5 đến 0.95 với khoảng cách bước nhảy là 0.05. Nó cho biết độ chính xác của mô hình trong việc phát hiện đối tượng với các ngưỡng IoU khác nhau, từ đó giúp đánh giá khả năng tổng quát hóa của mô hình đối với các trường hợp khác nhau.

## IV.2. Kết quả của model



*Bảng kết quả và confusion\_matrix của YoloV5*





### Các biểu đồ về precision, recall, F1 score

Accuracy (độ chính xác): Tỷ lệ số lần dự đoán chính xác trên tổng số lượng mẫu.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- Precision (độ chính xác của dự đoán dương tính): Tỷ lệ số lần dự đoán dương tính chính xác trên tổng số lượng mẫu dự đoán là dương tính.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- Recall (độ nhạy): Tỷ lệ số lần dự đoán dương tính chính xác trên tổng số lượng mẫu thực sự là dương tính.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- F1-score: Kết hợp giữa Precision và Recall, giúp đánh giá hiệu suất tổng thể của mô hình

$$\text{F1-score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Trong đó:

- TP (True Positive): số lượng các đối tượng quan tâm (biển số xe) được phát hiện chính xác.
- TN (True Negative): số lượng các đối tượng không phải đối tượng quan tâm (không phải biển số xe) được phát hiện chính xác.

- FP (False Positive): số lượng các đối tượng không phải đối tượng quan tâm (không phải biển số xe) bị phát hiện nhầm thành đối tượng quan tâm.
- FN (False Negative): số lượng các đối tượng quan tâm (biển số xe) bị bỏ sót (không phát hiện được).

Sau khi tính toán ta có thể thu được các thông số:

| Accuracy | F1_score | P     | R     | mAP50 | mAP50-95 |
|----------|----------|-------|-------|-------|----------|
| 0.70     | 0.83     | 0.788 | 0.848 | 0.848 | 0.76     |

- F1-score = 0.83, cho thấy mô hình có mức độ đúng giữa Precision và Recall khá tốt.
- Precision = 0.788, cho thấy trong các đối tượng mô hình dự đoán là biển số xe, có khoảng 78.8% là đúng.
- Recall = 0.848, cho thấy mô hình có khả năng phát hiện khoảng 84.8% biển số xe có trong ảnh.
- mAP5 = 0.848, cho thấy độ chính xác của mô hình đối với các mức ngưỡng khác nhau của độ tin cậy của phát hiện đạt 84.8%.

\* Đánh giá kết quả:

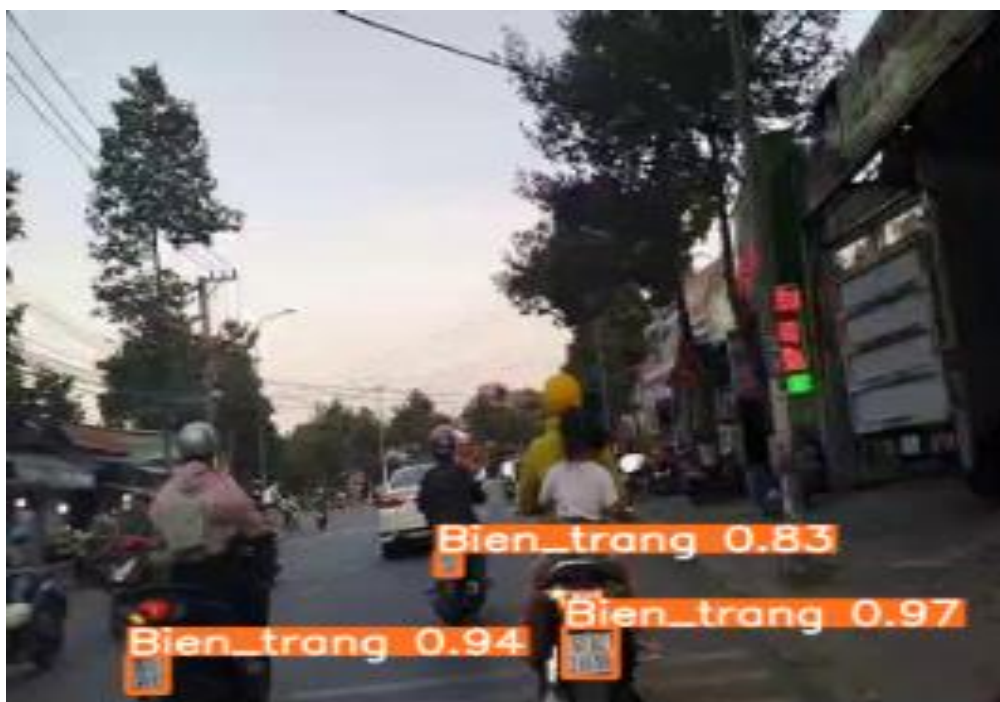
- Do bộ dữ liệu hạn chế về các loại biển ngoại giao, biển đỏ do ít dữ liệu thực bên ngoài nên mô hình phát hiện đúng đối với các loại biển này đối với bộ dữ liệu. Tỷ lệ mô hình dự đoán nền thành biển trắng khá cao cũng như không phát hiện được biển trắng. Có thể do biển bị mờ không phát hiện được hoặc trong quá trình xử lý dữ liệu không khớp.
- Với giá trị của Accuracy tính được là khoảng 0.70, ta có thể đánh giá rằng mô hình YOLOv5 đã đạt được hiệu suất tương đối tốt trong bài toán nhận dạng biển số xe.

### IV.3. Kết quả tốt

\* Đối với mô hình YoloV5



*YoloV5 nhận diện được tất cả các biển số trong điều kiện ánh sáng yếu*







#### IV.4. Kết quả chưa tốt

\*Mô hình YoloV5:



*YoloV5 nhận dạng nền thành các biển số*



\*Mô hình YoloV4



*YoloV4 các bouding\_box bị xấp chồng lên nhau và gán nhãn sai*



## IV.5. Đánh giá video

Trong việc nhận diện và làm mờ video, nhóm chọn 1 trong 2 model để tiến hành làm mờ video có chứa biển số xe là YoLoV5.

\* Nhận xét kết quả video sau khi làm mờ: Trong quá trình tiến hành che biển số trong video, phần khung che biển số sát biển, không bị lệch khỏi biển số khi chuyển đổi khung hình, che rất tốt các biển số nhìn thấy rõ. Khung hình theo sát biển số trong thời gian dài. Tuy nhiên có những phần che không chứa biển số hoặc biển số nhìn rõ nhưng không che. Nhưng với khung hình ít không quá ảnh hưởng đến việc nhận dạng và làm mờ biển số



*Trước và sau khi làm mờ*

Đường dẫn của file video sử dụng YoloV5 để làm mờ:

## V. Cải thiện dataset

- Nên thu thập nhiều biển khác nhau. Thay vì biển trắng để mô hình cho được kết quả tốt hơn. Đặc biệt là phải có được thông tin của biển đỏ, biển ngoại giao, biển xanh.
- Nên chọn các hình ảnh có chất lượng cao và có độ phân giải đủ lớn để đảm bảo rằng các biển số xe có thể được nhìn rõ và dễ dàng nhận dạng.
- Cần xử lý lại bộ dataset trong việc đóng khung các đối tượng để tránh tình trạng detect sai vị trí.
- Đối với việc cả hai mô hình đều detect ảnh nền. Vậy nên cần bộ sung dữ liệu có ảnh nền lớn hơn nhiều so với biển số xe.
- Thu thập dữ liệu từ nhiều nguồn khác nhau, bao gồm các điều kiện ánh sáng khác nhau, các góc chụp khác nhau, các môi trường chụp khác nhau, để đảm bảo rằng dataset của bạn đầy đủ và đa dạng.

## VI. Hướng phát triển

- Cải thiện chất lượng dataset: Bài toán nhận diện và làm mờ biển số xe yêu cầu một tập dữ liệu huấn luyện đủ lớn và đa dạng để mô hình có thể học được các trường hợp phức tạp. Do đó, việc thu thập và xử lý dataset cần được đảm bảo chất lượng.

Các phương pháp như data augmentation, thu thập và thêm dữ liệu mới, loại bỏ dữ liệu nhiễu có thể được sử dụng để cải thiện chất lượng dataset.

- Sử dụng kết hợp các mô hình mạng: Bằng cách kết hợp các mô hình mạng khác nhau, có thể giúp tăng cường khả năng nhận diện của mô hình. Ví dụ, sử dụng một mô hình nhận diện biển số xe và một mô hình làm mờ biển số xe có thể giúp tăng độ chính xác và hiệu quả của mô hình.
- Tối ưu hóa độ chính xác và tốc độ xử lý: YOLO là một mô hình nhanh, nhưng cần phải tối ưu hóa độ chính xác và tốc độ xử lý để có thể áp dụng vào các hệ thống thời gian thực. Các phương pháp tối ưu hóa mô hình như pruning, quantization, distillation, v.v. có thể được sử dụng để tối ưu hóa mô hình.

- Mở rộng ứng dụng cho các lĩnh vực khác nhau: Bài toán nhận diện và làm mờ biển số xe có thể được áp dụng vào nhiều lĩnh vực khác nhau, chẳng hạn như giám sát giao thông, an ninh, quản lý bãi đỗ xe, v.v. Do đó, việc mở rộng ứng dụng của bài toán cũng là một hướng phát triển tiềm năng.
- Tích hợp AI vào hệ thống camera giám sát giao thông: Bài toán nhận diện và làm mờ biển số xe có thể được tích hợp vào hệ thống camera giám sát giao thông để giúp cải thiện độ chính xác và hiệu quả của hệ thống

## TÀI LIỆU THAM KHẢO

### \*Model

- YOLOV4 trên GitHub: <https://github.com/AlexeyAB/darknet>
- YOLOv5 trên GitHub: <https://github.com/ultralytics/yolov5>

### \*Nguồn dataset:

- Thư viện MiAI: <https://miai.vn/thu-vien-mi-ai/>

\*