

Mô hình hỗ trợ người lái - Phát hiện đèn giao thông bằng YOLOv5

Nguyễn Phi Long
Department of Computer Science
University of Information Technology
Vietnam, Ho Chi Minh City
19521791@gm.uit.edu.vn

Abstract

Trong vài thập kỷ qua, những tiến bộ công nghệ trong xử lý hình ảnh và video đã mang lại cuộc cách mạng lớn trong cuộc sống của chúng ta. Một số lĩnh vực chính mà những tiến bộ này đóng vai trò quan trọng là: các hệ thống xe tự hành, công nghệ máy bay không người lái, giám sát đám đông, giám sát giao thông, theo dõi đối tượng, v.v. Trọng tâm của báo cáo này liên quan đến hệ thống phát hiện đèn giao thông tự động với các khả năng được cải thiện về độ phức tạp và độ chính xác về thời gian.

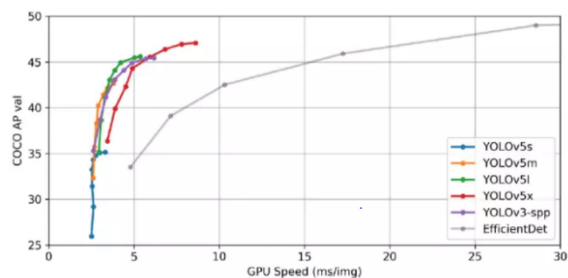
1. Introduction

Đèn giao thông là cần thiết để kiểm soát giao thông tại các giao lộ trong môi trường đô thị. Tuy nhiên, việc phát hiện đèn giao thông thường được thực hiện bằng camera. Phát hiện chính xác đèn giao thông trong thời gian thực vẫn là một thách thức mở. Nhiều yếu tố ảnh hưởng đến việc phát hiện đèn giao thông bao gồm điều kiện ánh sáng, giao lộ phức tạp, cấu trúc khác nhau của đèn giao thông và các đối tượng tương tự.

1.1 YOLO

YOLO-"You Only Look Once" một phương pháp phổ biến và được yêu thích

cho các bài toán về detection. YOLOv1 được ra mắt vào tháng 5 năm 2016 bởi Joseph Redmon với paper "You Only Look Once: Unified, Real-Time Object Detection". Tới tháng 12 năm 2017 thì Joseph introduced tiếp tục công bố một version khác của YOLO với paper "YOLO9000: Better, Faster, Stronger" nó được biết đến với tên gọi là YOLO9000. Sau đó chưa đầy một năm đến tháng 4 năm 2018, một phiên bản mong đợi nhất của YOLO đã được ra mắt với tên gọi YOLOv3 với paper "YOLOv3: An Incremental Improvement". Đúng 2 năm sau Alexey Bochkovskiy đã giới thiệu YOLOv4 với paper "YOLOv4: Optimal Speed and Accuracy of Object Detection". Sau đó vài ngày thì Glenn Jocher đã phá thành YOLOv5. Glenn đã trình làng PyTorch based version of YOLOv5 với những sự cải tiến đáng kể.



Hình 1 1 YOLOv5 in mAP on the COCO bench-mark

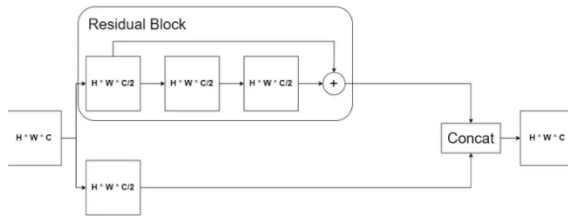
1.2 YOLOv5

Trong khi YOLOv4 là sự lắp ghép có chọn lọc của các nghiên cứu trong Object Detection lại với nhau thì YOLOv5 không có quá nhiều thay đổi so với YOLOv4. YOLOv5 tập trung vào tốc độ và độ dễ sử dụng.

1.2.1 Backbone

1.2.1.1 C3 Module

YOLOv5 cải tiến CSPResBlock của YOLOv4 thành một module mới, ít hơn một lớp Convolution gọi là C3 module.



Hình 1 2 C3 Module.

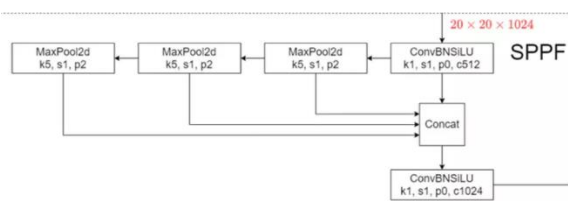
1.2.1.2 Activation function

YOLOv4 sử dụng Mish hoặc LeakyReLU cho phiên bản nhẹ, còn sang YOLOv5, activation function được sử dụng là SiLU.

1.2.2 Neck

1.2.2.1 SPPF

YOLOv5 áp dụng một module giống với SPP, nhưng nhanh hơn gấp đôi và gọi đó là SPP - Fast (SPPF).



Hình 1 3 SPPF architecture.

Thay vì sử dụng MaxPooling song song như trong SPP, SPPF của YOLOv5 sử dụng MaxPooling tuần tự. Hơn nữa, kernel size trong MaxPooling của SPPF toàn bộ là 5 thay vì là [5,9,13] như SPP của YOLOv4.

1.2.2.2 Loại bỏ Grid Sensitivity

Để giải quyết Grid Sensitivity, YOLOv5 sử dụng công thức sau:

$$b_x = 2\sigma(t_x) - 0.5 + c_x$$

$$b_y = 2\sigma(t_y) - 0.5 + c_y$$

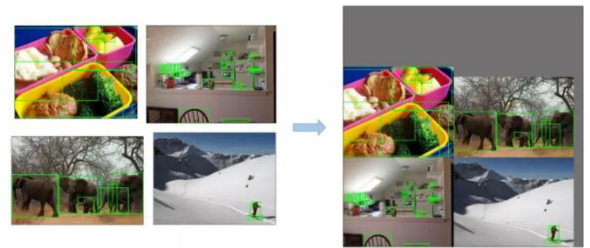
$$b_w = p_w(2\sigma(t_w))^2$$

$$b_h = p_h(2\sigma(t_h))^2$$

1.2.2.3 Xử lý data

Các kỹ thuật Data Augmentation được áp dụng trong YOLOv5 bao gồm:

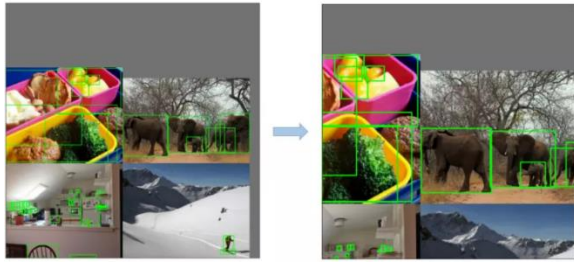
- Mosaic Augmentation



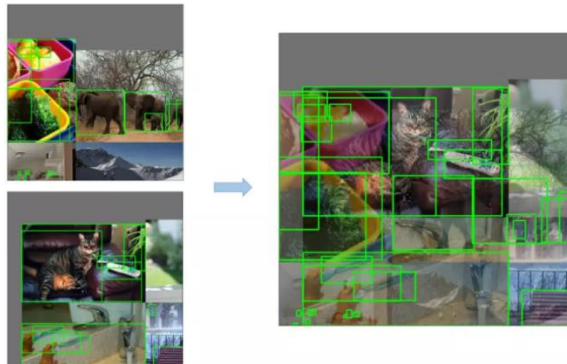
- Copy-paste Augmentation



- Random Affine transform



- MixUp Augmentation



- Và các thay đổi về màu sắc cũng như là Random Flip của Albumentations.

1.2.3 Head

Giữ nguyên từ YOLOv3.

2. Dữ liệu

Dữ liệu yêu cầu cho bài toán phải là dữ liệu được quay từ camera hành trình trên xe ô tô.

Dữ liệu được thu thập từ những thành phố nổi tiếng trên thế giới và ở Việt Nam như: Los Angeles, Beijing, New York, Seoul, Hồ Chí Minh, Hà Nội và được quay vào ban ngày, không có mưa, không có sương mù, tầm nhìn quang đãng.



Hình 1 4 Los Angeles.



Hình 1 5 New York.



Hình 1 6 Beijing.

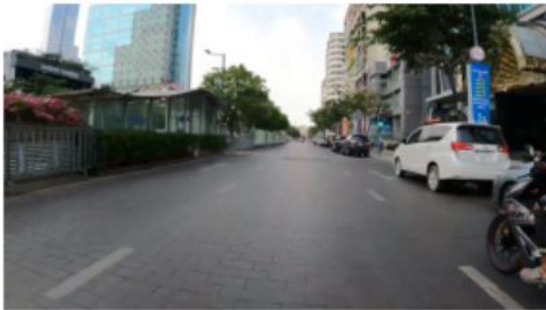


Hình 1 7 Seoul.



Hình 1 8 Ho Chi Minh.

Dữ liệu ở Việt Nam đúng với yêu cầu khá là ít nên nhóm đã sử dụng tăng cường dữ liệu: Flip, Blur, Brighten, Darken, High Contrast.



Hình 1 9 Original.



Hình 1 10 Flip.



Hình 1 11 Gaussian Blur.



Hình 1 12 High Contrast.



Hình 1 13 Darken.

2.1 Tiền xử lý dữ liệu

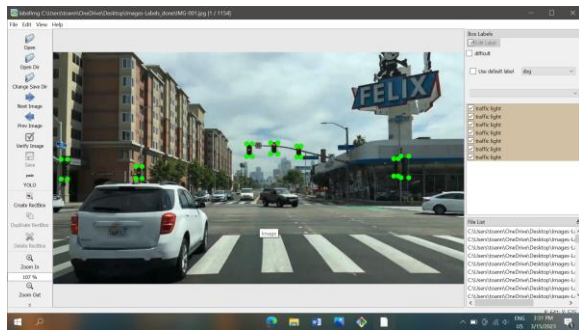
Các khung hình được trích xuất từ các video có kích thước 720 x 1280. Theo như nhóm tìm hiểu thì đầu vào của YOLOv5 là một tấm ảnh hình vuông nên nhóm quyết định giảm kích thước ảnh xuống 720 x 720. Vừa giảm tham số trong lúc huấn luyện, vừa khớp với đầu vào của YOLOv5. Ngoài ra, nhóm còn giảm dung lượng của ảnh nhưng vẫn giữ nguyên chất

lượng với mục đích giảm thời gian tải ảnh lên Google Drive cũng như lưu trữ.

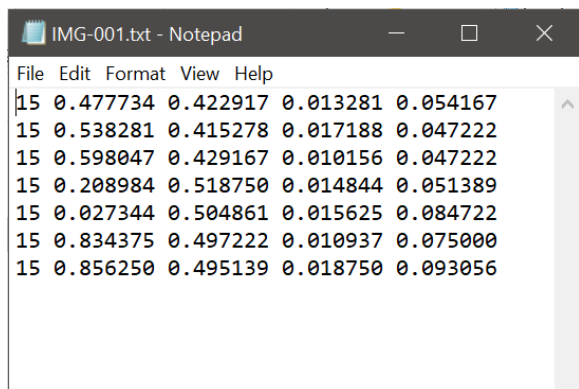
2.2 Gán nhãn

Để gán nhãn cho ảnh, nhóm sử dụng labelImg với một class duy nhất là traffic light, các nhãn này nhóm cấu hình về định dạng YOLO (file.txt). Định dạng annotation của YOLO có dạng như sau:

<object-class><x><y> <width> <height>



Hình 1 14 LabelImg.



Hình 1 15 YOLO Format.

2.3 Tập dữ liệu

Dataset: 1154 ảnh

Sẽ được tách thành:

- Training set: 923 ảnh.
- Validation set: 115 ảnh.
- Testing set: 116 ảnh.

3. Huấn luyện mô hình

Vì tập dữ liệu khá ít nên nhóm quyết định chọn mô hình pre-trained YOLOv5s.

Tạo 1 file YAML để cấu hình dữ liệu:

- Train, test, val: Đường dẫn đến tập train, test, val.
- Nc: số lượng class trong dữ liệu.
- Name: Tên của class. Chỉ mục của các class sẽ được sử dụng như một **identifier** cho class đó trong mô hình.

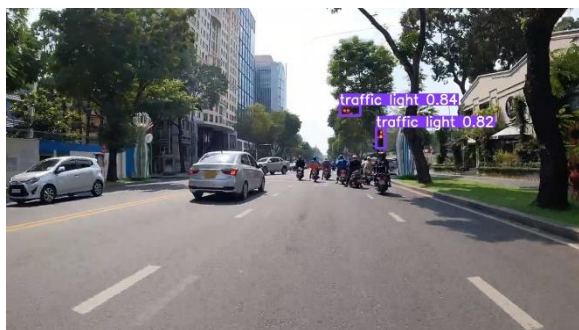
Tiến hành huấn luyện dữ liệu:

- Batch: 32
- Epochs: 100
- Cfg: yolov5s.yaml
- Hyp: hyp.scratch-high.yaml
- Weights: yolov5s.pt
- Worker: 24
- Data: road_sign_data.yaml

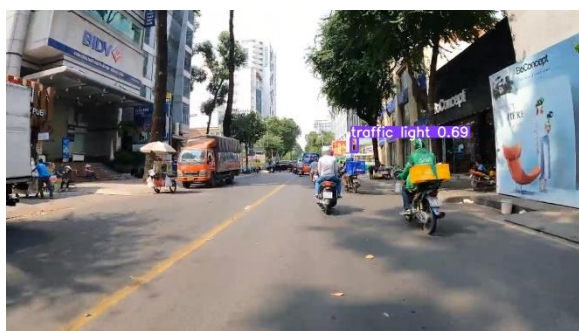
Sau hơn hai tiếng huấn luyện thì đây là các kết quả mà nhóm thu được:



Hình 1 16 Predict.



Hình 1 17 Predict.



Hình 1 18 Predict.



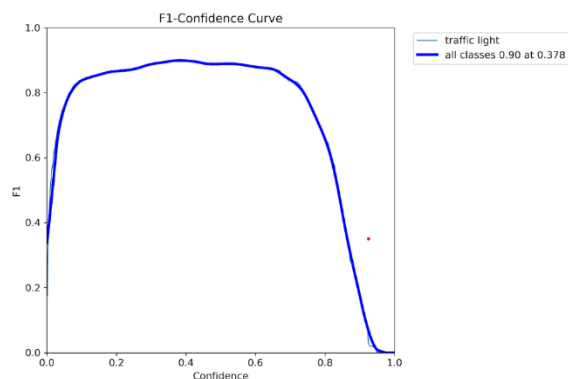
Hình 1 19 Predict.



Hình 1 20 Predict.

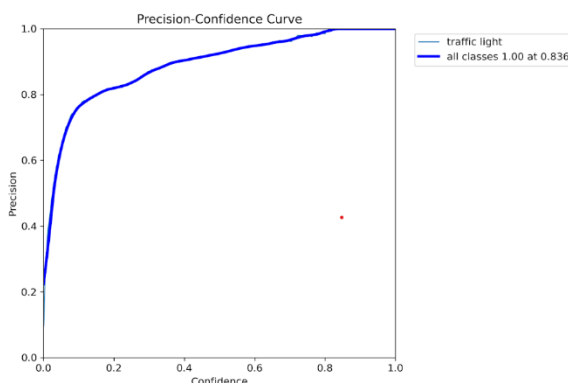
4. Đánh giá mô hình

Kết quả cho các độ đo hiệu suất khác nhau đã thu được trong quá trình huấn luyện bằng cách sử dụng tập train và validation.



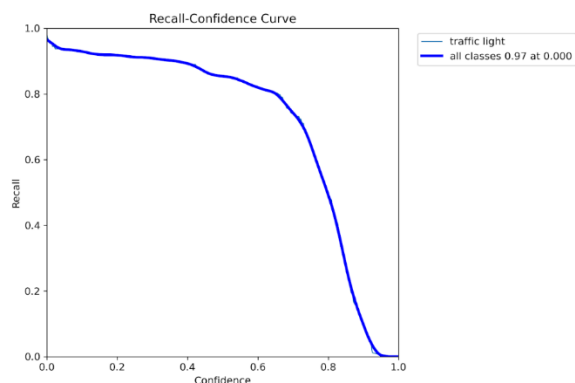
Hình 1 21 F1 curve.

Từ biểu đồ đường cong F1, giá trị confidence tối ưu precision và recall là 0.378, đường màu xanh khá cân bằng.

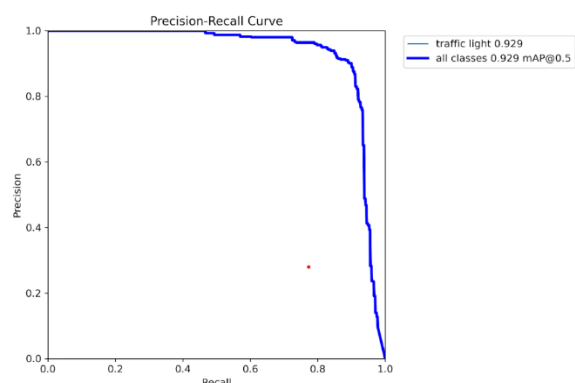


Hình 1 22 Precision curve.

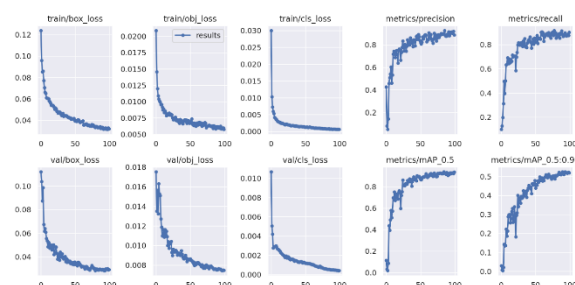
	P	R	mAP@.5	mAP@.5-.95
Traffic light	0.9	0.896	0.929	0.524



Hình 1 23 Recall curve.



Hình 1 24 Precision Recall curve.



Hình 1 25 Result.

Trong 3 biểu đồ đầu bên trên và 3 biểu đồ đầu bên dưới, Loss giảm dần qua các epochs. Với 4 biểu đồ cuối, độ chính xác của mô hình tăng dần qua các epochs.

Mặc dù có vài trường hợp phát hiện sai hoặc không phát hiện gì nhưng ta có thể thấy mô hình phát hiện khá tốt trên custom dataset.

Predicted	traffic light -	0.98	1.00
	background -	0.02	
		traffic light -	background -
		True	

Hình 1 26 Confusion matrix.

Từ confusion matrix ta có thể thấy 98% traffic light được phát hiện đúng, trong khi đó 2% bị nhầm với background. Không có trường hợp nào nhầm background với traffic light.

5. Kết luận

Một cách tiếp cận hiệu quả về chi phí để phát hiện đèn giao thông bằng cách sử dụng mô hình YOLOv5 cho độ chính xác cao và thời gian tính toán thấp.