



# TRIỂN KHAI ỨNG DỤNG AI, IoT

**Nhóm 7 : Nhận diện biển số xe và phát  
hiện sai phạm vượt đèn đỏ**

**Giảng viên hướng dẫn: ThS.Lê Trung Hiếu**

**ThS.Nguyễn Thái Khánh**

# Mục lục

---

- 1.Đặt vấn đề
- 2.Các nghiên cứu liên quan
- 3.Mục tiêu
- 4.Phương pháp
- 5.Xây dựng hệ thống
- 6.Bước thực hiện
- 7.Train Model
- 8.Model Nhận diện vật thể
- 9.Model Nhận diện biển số
- 10.Training Model
- 11.Kết quả cuối cùng
- 12.Kết luận

# 1. Đặt vấn đề

Vượt qua đèn đỏ là một trong những hành vi vi phạm giao thông phổ biến, gây nguy hiểm và khó kiểm soát. Hiện nay, việc xử lý vi phạm chủ yếu dựa vào con người, dẫn đến hạn chế về tính chính xác, minh bạch và hiệu quả.

Hệ thống "Nhận diện biển số xe và phát hiện sai phạm vượt đèn đỏ" ứng dụng AI và thị giác máy tính nhằm:

- Nhận diện biển số xe bằng OCR.
- Phát triển phương tiện thông tin hiện đại (ô tô, xe máy, xe bubu).
- Xác định phạm vi dựa trên đèn tín hiệu và vạch dừng.
- Lưu trữ & xử lý dữ liệu vi phạm để hỗ trợ xử lý tự động.





## 2. Các nghiên cứu liên quan

### License plate recognition and red light violation detection YOLOV8

Mô tả : Dự án này triển khai mô hình YOLOv8 để phát hiện biển số xe vi phạm

Nguồn GitHub:

<https://github.com/autogyro/yolo-V8>



## 2. Các nghiên cứu liên quan

### TrafficDetection with YOLOv8

Mô tả : Dự án này triển khai mô hình YOLOv8 để phát hiện các loại xe khác nhau

Nguồn GitHub:

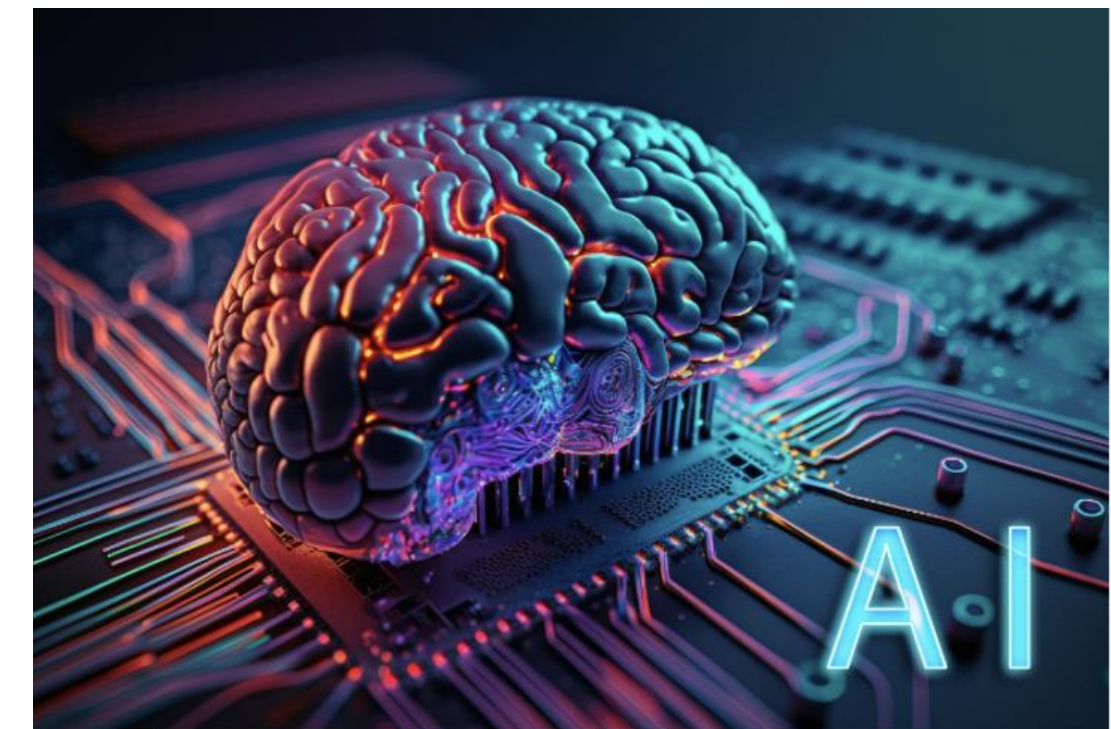
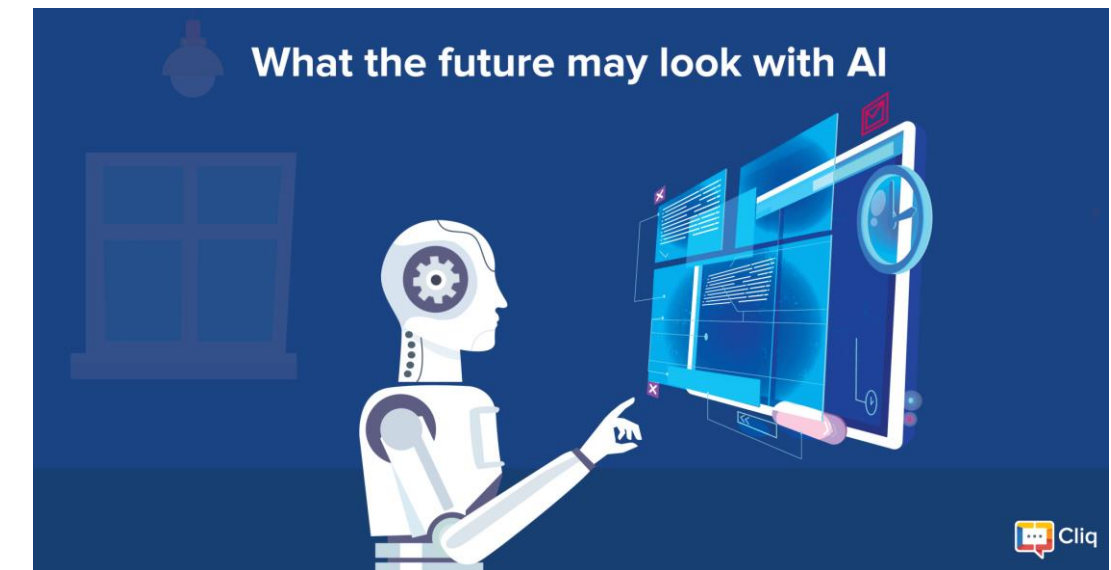
<https://github.com/antonmilev/TrafficDetection>





### 3. Mục tiêu

- Xây dựng hệ thống biển số nhận dạng tự động
- Phát hiện và nhận dạng biển số từ hình ảnh/video bằng cách sử dụng các thuật toán học sâu và hình ảnh xử lý.
- Trích xuất thông tin biển số xe với độ chính xác cao.
- Phát hiện vi phạm vượt qua đèn đỏ
- Nhận thông tin trạng thái giao diện trong video theo thời gian thực
- Theo dõi và xác định vi phạm của phương tiện khi đèn đỏ xảy ra



## 4. Phương pháp

YOLOv8 có độ chính xác cao

So với YOLOv5, YOLOv8 có mAP cao hơn và khả năng nhận diện tốt hơn với vật thể nhỏ như biển số xe.

Tốc độ xử lý nhanh

- YOLOv8 sử dụng kiến trúc CNN nhẹ hơn và hỗ trợ tăng tốc GPU với TensorRT, ONNX.

- Trong ứng dụng của em (nhận diện biển số + vượt đèn đỏ), tốc độ xử lý là yếu tố quan trọng. YOLOv8 có thể chạy real-time

Hỗ trợ nhiều mô hình khác nhau

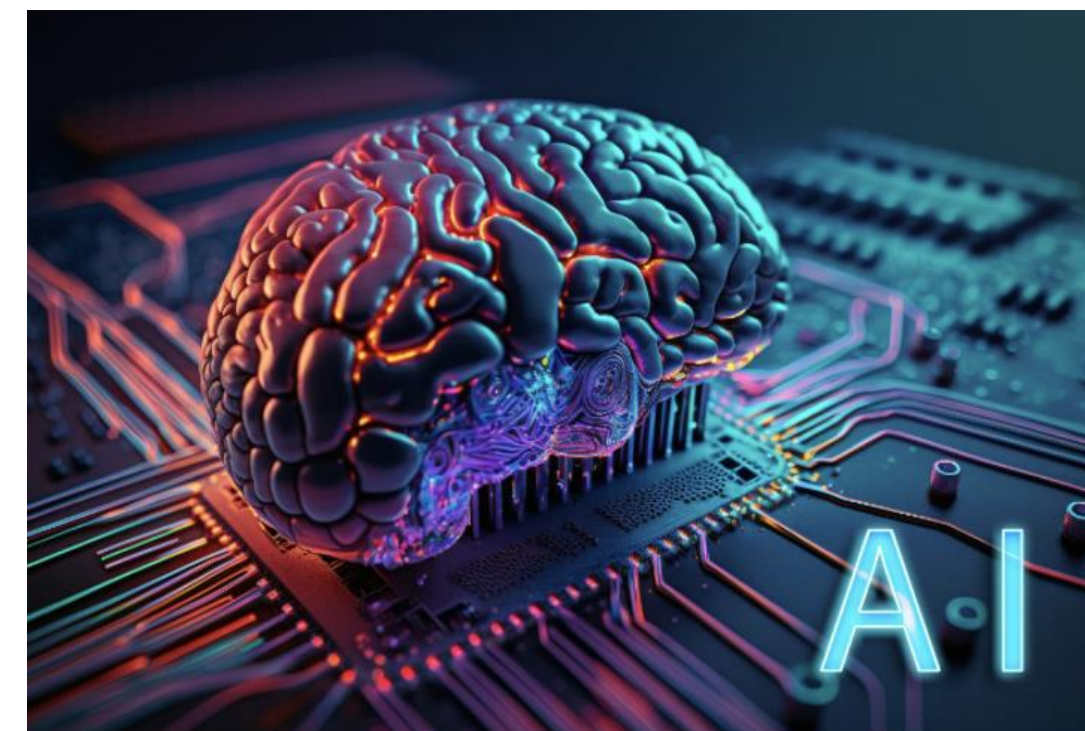
- YOLOv8n (Nano): Siêu nhẹ, chạy nhanh trên thiết bị nhúng như Raspberry Pi.

- YOLOv8s/m/l/x: Cân bằng giữa tốc độ và độ chính xác, phù hợp với bài toán giao thông.

Dễ dàng huấn luyện với dữ liệu mới

- YOLOv8 hỗ trợ tự động huấn luyện và tối ưu trên dataset riêng.

Khả năng nhận diện đa dạng đối tượng





## 4. Phương pháp

OCR (Optical Character Recognition) là công nghệ nhận dạng ký tự quang học, giúp trích xuất văn bản từ hình ảnh hoặc tài liệu scan. Dưới đây là phân tích về OCR theo các khía cạnh quan trọng:

### 1. Nguyên lý hoạt động của OCR

- **Tiền xử lý hình ảnh:** Chuẩn hóa ảnh (chuyển grayscale, làm mịn, giảm nhiễu, nhị phân hóa).
- **Phân đoạn ký tự:** Xác định vị trí văn bản, tách chữ cái hoặc từ.
- **Nhận dạng ký tự:** Sử dụng thuật toán (Pattern Matching, Machine Learning, Deep Learning) để xác định chữ cái.
- **Hậu xử lý:** Sửa lỗi nhận dạng, ghép câu, kiểm tra ngữ cảnh.

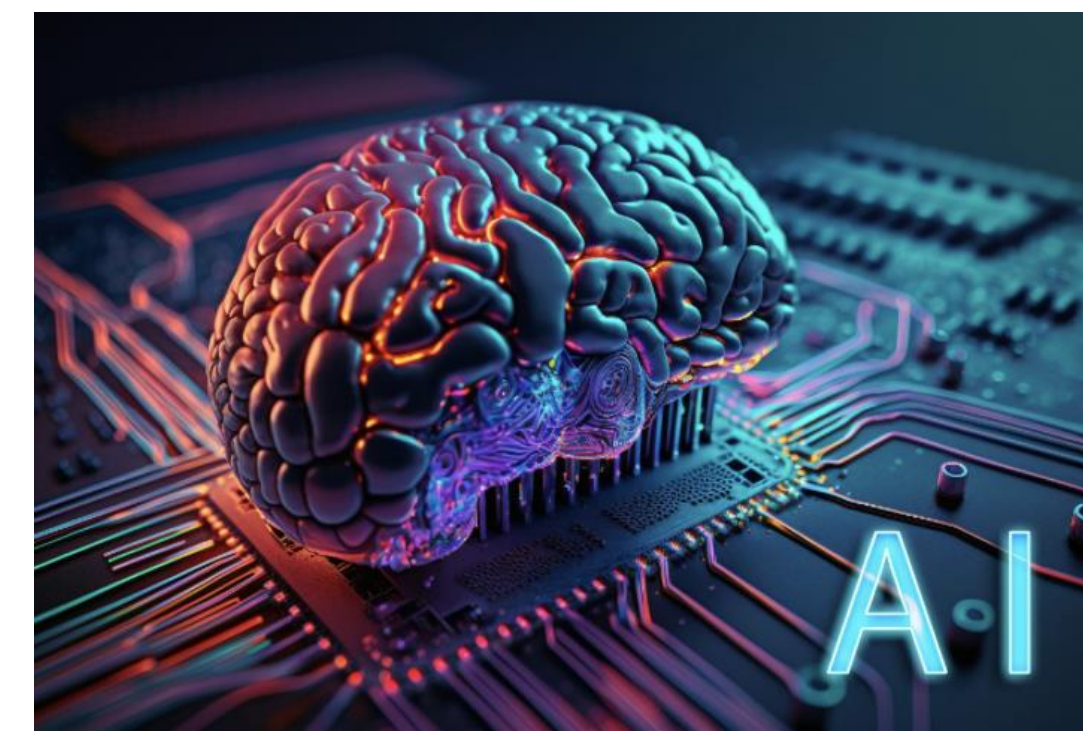
### 2. Cách hoạt động

#### • Tesseract OCR:

- Dựa trên **LSTM** và **Pattern Matching** để nhận diện ký tự.
- Cần tiền xử lý ảnh tốt (nhị phân hóa, làm rõ nét).
- Chạy trên CPU, có thể dùng với OpenCV để cải thiện kết quả.

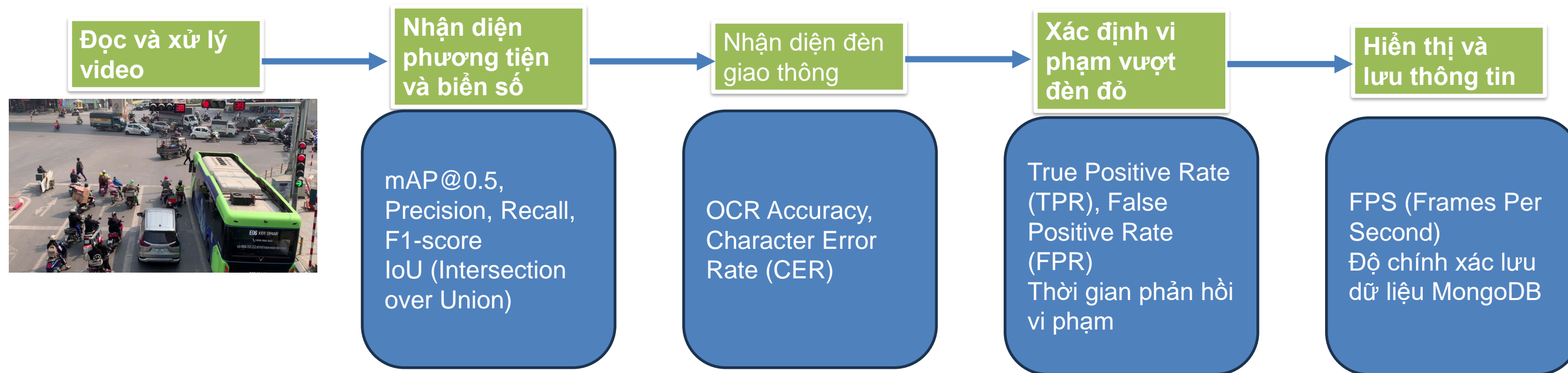
#### • EasyOCR:

- Sử dụng **CNN + BiLSTM + CTC Decoder** (Deep Learning).
- Hỗ trợ văn bản có font chữ đa dạng, nền phức tạp.
- Hoạt động tốt ngay cả khi ảnh có độ nhiễu cao.

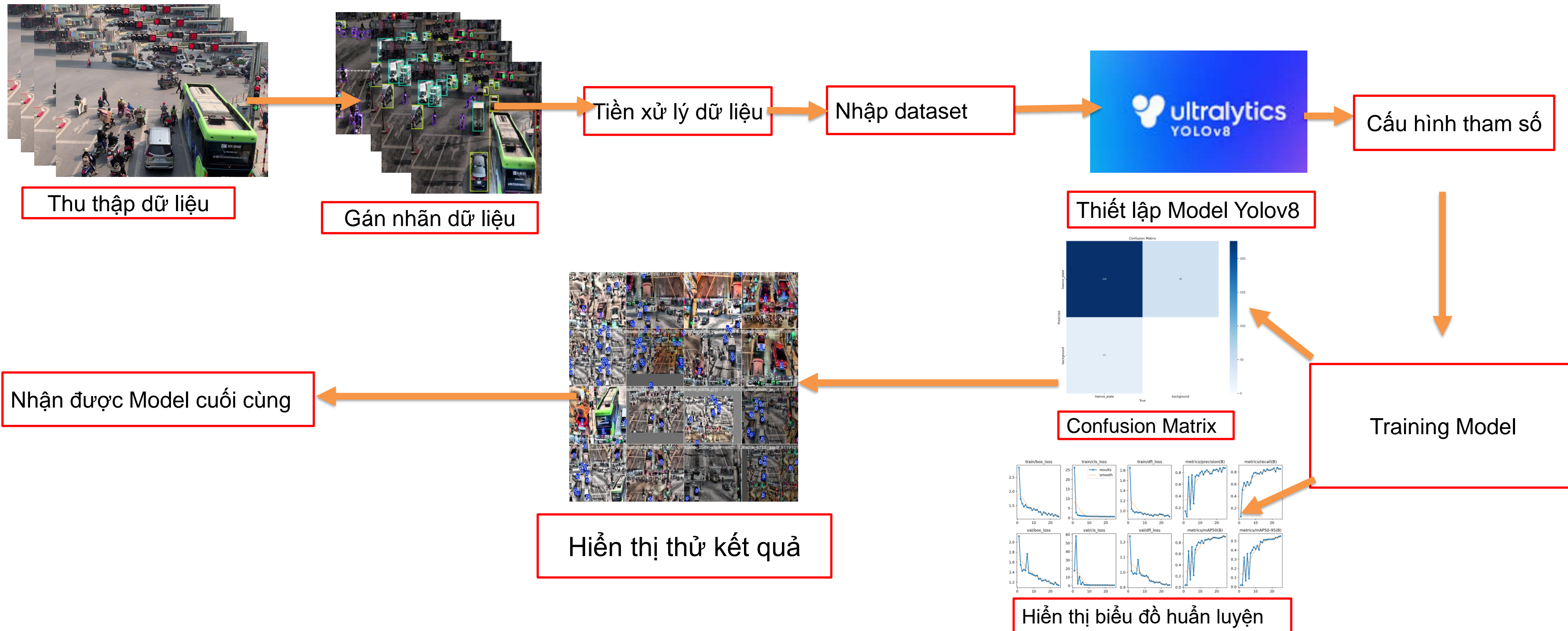




## 5. Xây dựng hệ thống

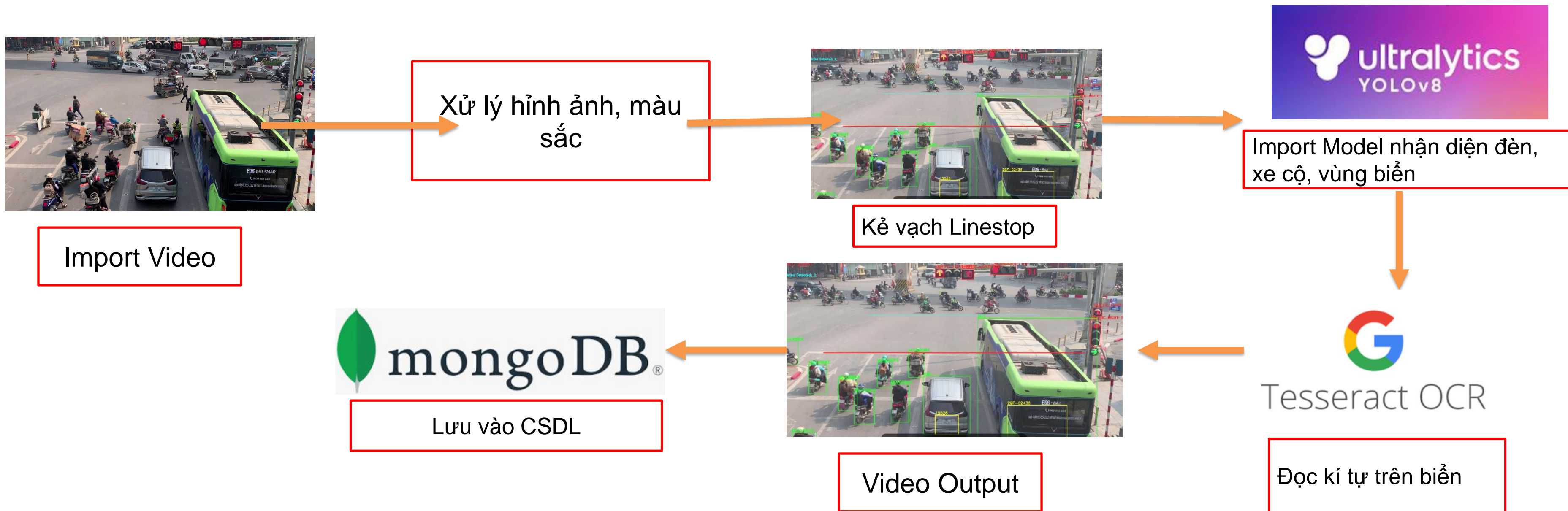


## 6. Bước thực hiện





## 6. Bước thực hiện



# 7.TRAIN MODEL

---

## 1.Chuẩn Bị Dữ Liệu

- Thu thập hình ảnh có chứa các đối tượng cần nhận diện.
- Gán nhãn cho từng ảnh bằng công cụ Roboflow, tạo ra file nhãn đi kèm.
- Tổ chức dữ liệu theo cấu trúc phù hợp:
- Thư mục ảnh huấn luyện (train).
- Thư mục ảnh kiểm tra (val).
- Thư mục ảnh kiểm tra mô hình (Test).
- Thư mục nhãn chứa file nhãn tương ứng với từng ảnh.
- File cấu hình dataset, khai báo thông tin về các lớp đối tượng.



## 8. Model Nhận diện vật thể

315 Total Images

[View All Images →](#)



Dataset Split

TRAIN SET

70%

221 Images

VALID SET

20%

63 Images

TEST SET

10%

31 Images

Số lượng ảnh của Dataset

```
names:
- BUS
- CAR
- MOTORBIKE
- TRAFFIC_LIGHT
- TRUCK
nc: 5
roboflow:
  license: CC BY 4.0
  project: vehicle_dataset-04axq
  url: https://universe.roboflow.com/thang-azym/vehicle_dataset-04axq/dataset/1
  version: 1
  workspace: thang-azym
test: ../test/images
train: ../train/images
val: ../valid/images
```

Cấu trúc file data.Yaml

## 8. Model Nhận diện vật thể

```
epoch,time,train/box_loss,train/cls_loss,train/df1_loss,metrics/precision(B),metrics/recall(B),metrics/mAP50(B),metrics/mAP50-95(B),
1,11.2531,1.55604,2.78441,1.32391,0.66825,0.59878,0.6874,0.52458,1.28717,2.06256,1.17225,0.00014443,0.00014443,0.00014443
2,15.2712,1.07542,1.06892,1.04764,0.72475,0.80895,0.82918,0.67832,0.99514,1.29282,1.0164,0.000288091,0.000288091,0.000288091
3,19.7553,0.98343,0.89025,1.00789,0.81233,0.86193,0.88801,0.74464,0.9643,0.95827,0.99996,0.000419434,0.000419434,0.000419434
4,24.1474,0.93928,0.79165,0.98131,0.87986,0.85488,0.90612,0.74751,0.93208,0.81676,0.97216,0.000538457,0.000538457,0.000538457
5,28.5284,0.88137,0.70591,0.95318,0.84679,0.88115,0.92107,0.79459,0.8366,0.69297,0.94076,0.000645162,0.000645162,0.000645162
6,32.9874,0.87088,0.69238,0.95223,0.80809,0.7988,0.85671,0.6967,0.85739,0.7465,0.95132,0.000739548,0.000739548,0.000739548
7,36.952,0.83204,0.6492,0.93259,0.8335,0.87664,0.89133,0.7428,0.88204,0.69196,0.95185,0.000821616,0.000821616,0.000821616
8,40.8873,0.82611,0.644,0.92798,0.85416,0.86004,0.89611,0.77808,0.8045,0.65509,0.92543,0.000803031,0.000803031,0.000803031
9,45.013,0.82829,0.64647,0.93541,0.86675,0.86904,0.91089,0.79874,0.81103,0.60012,0.92985,0.000759035,0.000759035,0.000759035
10,49.6418,0.80222,0.61289,0.92442,0.90147,0.88518,0.92603,0.81967,0.72856,0.56567,0.89773,0.00071504,0.00071504,0.00071504
11,53.9067,0.7649,0.59759,0.92758,0.90428,0.89328,0.93222,0.83432,0.73764,0.54006,0.9014,0.000671044,0.000671044,0.000671044
12,58.2089,0.73278,0.56348,0.90977,0.87744,0.91606,0.93047,0.8345,0.69541,0.53237,0.88807,0.000627048,0.000627048,0.000627048
13,62.5538,0.71028,0.56066,0.90255,0.89122,0.91057,0.93941,0.84211,0.71018,0.53278,0.89898,0.000583053,0.000583053,0.000583053
14,66.9254,0.70293,0.5382,0.90551,0.88201,0.87764,0.93611,0.84324,0.69012,0.53393,0.89049,0.000539057,0.000539057,0.000539057
15,71.3384,0.68918,0.53561,0.89507,0.88075,0.91202,0.93911,0.85391,0.66576,0.51705,0.88338,0.000495062,0.000495062,0.000495062
16,80.1432,0.67989,0.55618,0.89765,0.89178,0.91123,0.93828,0.85338,0.67203,0.52003,0.88149,0.000451066,0.000451066,0.000451066
17,84.5081,0.68082,0.52396,0.89461,0.90454,0.89765,0.93465,0.84596,0.73186,0.5242,0.90259,0.00040707,0.00040707,0.00040707
18,88.6972,0.65563,0.50534,0.89057,0.89116,0.91668,0.93997,0.85443,0.6754,0.50989,0.88781,0.000363075,0.000363075,0.000363075
19,92.9223,0.63171,0.4888,0.88209,0.89615,0.91365,0.94389,0.86444,0.65725,0.495,0.87855,0.000319079,0.000319079,0.000319079
20,97.0974,0.62399,0.4746,0.87787,0.88453,0.91713,0.94297,0.86054,0.66876,0.49231,0.88923,0.000275084,0.000275084,0.000275084
21,100.818,0.60879,0.46692,0.87626,0.88858,0.92004,0.94479,0.86541,0.64904,0.48267,0.87914,0.000231088,0.000231088,0.000231088
22,104.976,0.59719,0.46168,0.86936,0.8834,0.92171,0.94654,0.87085,0.64183,0.47032,0.87794,0.000187092,0.000187092,0.000187092
23,109.109,0.59623,0.45031,0.86528,0.89411,0.91308,0.94547,0.87333,0.62423,0.4633,0.87301,0.000143097,0.000143097,0.000143097
24,113.389,0.59031,0.44508,0.8702,0.91555,0.90119,0.94638,0.87586,0.6056,0.4552,0.86434,9.91012e-05,9.91012e-05,9.91012e-05
25,117.645,0.57754,0.43705,0.8682,0.91447,0.90277,0.94608,0.87806,0.60235,0.45362,0.86375,5.51056e-05,5.51056e-05,5.51056e-05
```

Thông số sau khi Train Model



# 9. Model Nhận diện biển số

281 Total Images

[View All Images →](#)



Dataset Split

TRAIN SET

70%

197 Images

VALID SET

20%

56 Images

TEST SET

10%

28 Images

Số lượng ảnh trong Dataset

```
1  names:
2    - liscence_plate
3  nc: 1
4  roboflow:
5    license: CC BY 4.0
6    project: liscence_plate-kwxxx
7    url: https://universe.roboflow.com/thang-azym/liscence_plate-kwxxx/dataset/1
8    version: 1
9    workspace: thang-azym
10 test: ../test/images
11 train: ../train/images
12 val: ../valid/images
13
```

Cấu trúc file data.Yaml

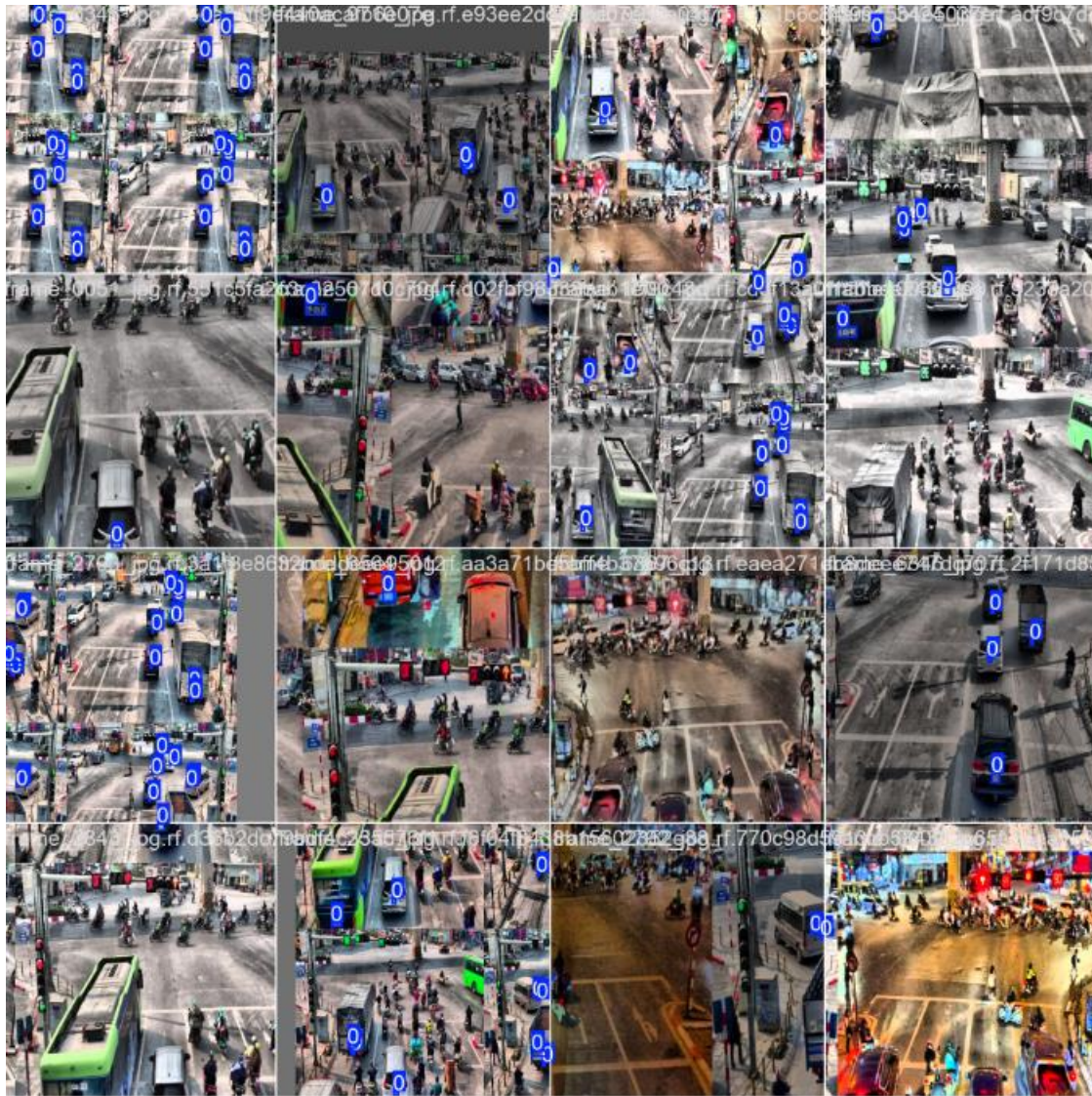
# 9. Model Nhận diện biển số

```
epoch,time,train/box_loss,train/cls_loss,train/dfl_loss,metrics/precision(B),metrics/recall(B),metrics/mAP50(B),metrics/mAP50-95(B),
1,10.443,2.84532,26.223,1.85478,0.14409,0.06024,0.03795,0.01847,2.12099,17.5515,1.23952,0.00024,0.00024,0.00024
2,14.9128,1.73605,2.77071,1.04094,0.04936,0.50602,0.03655,0.01778,1.54542,58.1831,1.00938,0.0004802,0.0004802,0.0004802
3,19.9774,1.56974,1.41696,0.99392,0.72486,0.62249,0.64944,0.32166,1.42273,2.24713,0.98853,0.000699808,0.000699808,0.000699808
4,24.9817,1.46166,1.18071,0.96948,0.17632,0.56627,0.13761,0.06873,1.45095,10.5415,0.99505,0.000898824,0.000898824,0.000898824
5,29.1644,1.52068,1.12212,0.98145,0.76074,0.63855,0.72147,0.36121,1.43652,1.51852,0.9882,0.00107725,0.00107725,0.00107725
6,32.8836,1.43346,0.97744,0.96502,0.27108,0.58233,0.21321,0.08917,1.76764,3.94592,1.08359,0.00123508,0.00123508,0.00123508
7,36.2213,1.42135,0.96864,0.97237,0.73143,0.62249,0.67088,0.37014,1.38435,1.16593,0.99498,0.00137232,0.00137232,0.00137232
8,39.9211,1.42375,0.87214,0.96382,0.76148,0.73084,0.75178,0.39958,1.37506,0.93429,0.9815,0.0014456,0.0014456,0.0014456
9,43.586,1.33356,0.80539,0.93401,0.74266,0.78313,0.807,0.43376,1.35708,0.89687,0.97647,0.0013664,0.0013664,0.0013664
10,47.205,1.38666,0.81311,0.95068,0.79447,0.79171,0.79184,0.41027,1.33886,0.89537,0.97328,0.0012872,0.0012872,0.0012872
11,51.4247,1.34396,0.82325,0.9286,0.82494,0.77595,0.83668,0.45076,1.33014,0.75646,0.97849,0.001208,0.001208,0.001208
12,55.2984,1.35723,0.78072,0.93722,0.76446,0.76904,0.78818,0.4042,1.33224,0.83742,0.96734,0.0011288,0.0011288,0.0011288
13,59.0931,1.27699,0.77629,0.91465,0.81571,0.7992,0.8502,0.49488,1.26523,0.76911,0.94486,0.0010496,0.0010496,0.0010496
14,62.8918,1.27458,0.74394,0.92067,0.84245,0.77308,0.85195,0.48446,1.26864,0.7351,0.94122,0.0009704,0.0009704,0.0009704
15,66.5928,1.17291,0.73641,0.89697,0.81564,0.83534,0.8683,0.51408,1.22853,0.69497,0.93863,0.0008912,0.0008912,0.0008912
16,75.9291,1.26779,0.72038,0.92722,0.78018,0.81246,0.85016,0.51228,1.23563,0.70479,0.93171,0.000812,0.000812,0.000812
17,79.6775,1.23593,0.71598,0.92141,0.78448,0.84784,0.8726,0.51596,1.24527,0.67967,0.93507,0.0007328,0.0007328,0.0007328
18,83.3696,1.1786,0.68145,0.91438,0.84386,0.82731,0.89016,0.5186,1.21677,0.66945,0.93233,0.0006536,0.0006536,0.0006536
19,87.0615,1.22276,0.69928,0.93917,0.84196,0.83133,0.89124,0.51869,1.22045,0.69141,0.93443,0.0005744,0.0005744,0.0005744
20,90.8216,1.16897,0.67169,0.93245,0.85705,0.84273,0.88422,0.52036,1.18492,0.65492,0.92545,0.0004952,0.0004952,0.0004952
21,94.4938,1.20871,0.65956,0.92518,0.82103,0.8659,0.87608,0.52239,1.17406,0.65453,0.92077,0.000416,0.000416,0.000416
22,98.225,1.13316,0.64152,0.89984,0.87528,0.80723,0.88395,0.53562,1.15676,0.63677,0.91892,0.0003368,0.0003368,0.0003368
23,102.131,1.17737,0.66687,0.90566,0.81314,0.87384,0.8917,0.53645,1.1947,0.6438,0.92219,0.0002576,0.0002576,0.0002576
24,105.893,1.13412,0.63922,0.91481,0.88568,0.85141,0.91579,0.54849,1.15378,0.62063,0.91482,0.0001784,0.0001784,0.0001784
25,109.59,1.10945,0.61881,0.88751,0.87596,0.85082,0.90723,0.55362,1.14051,0.62429,0.91626,9.92e-05,9.92e-05,9.92e-05
```

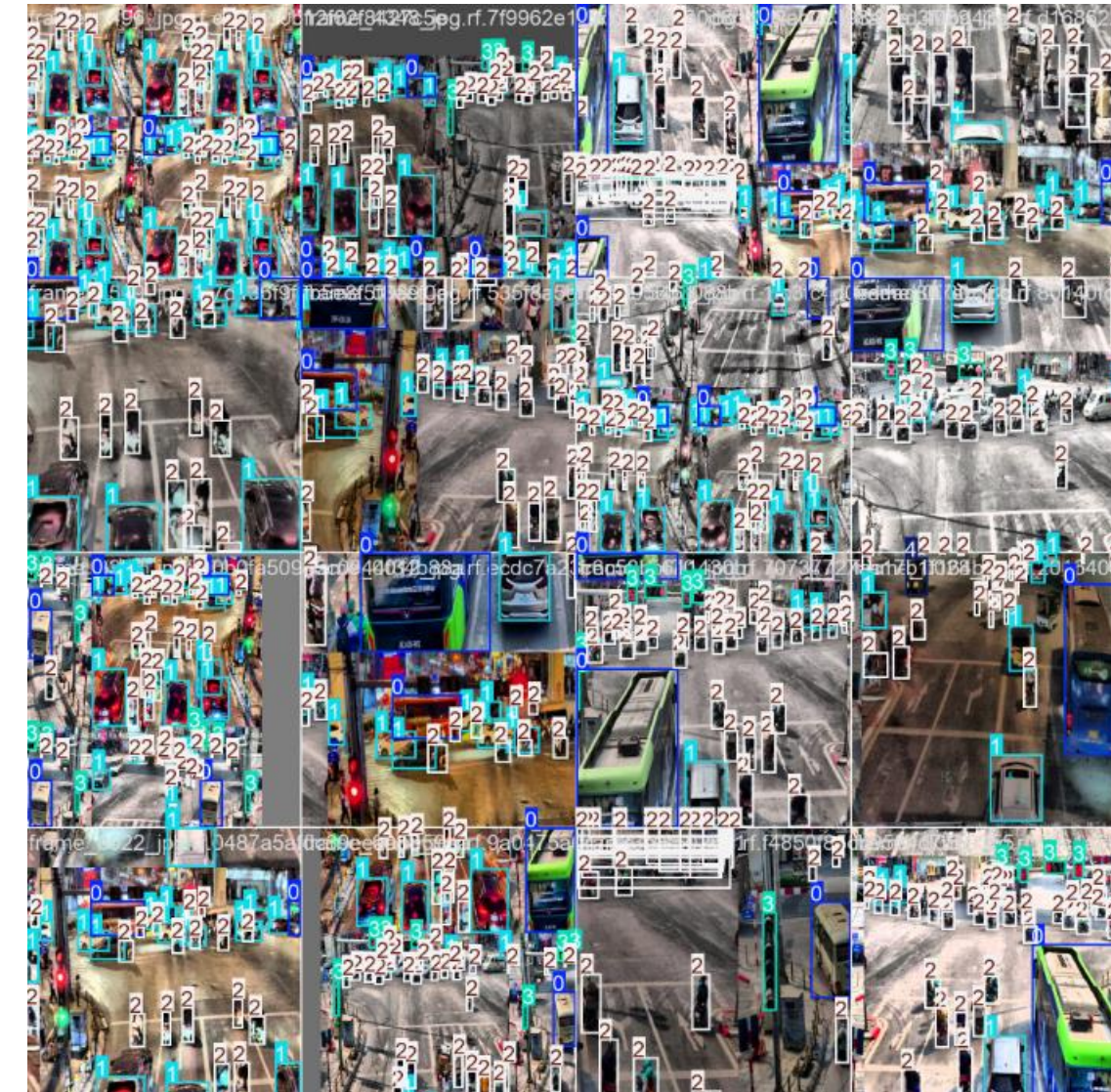
Thông số sau khi Train Model



# 10.Training Model



**Kết quả thực nghiệm model  
nhận diện biển số**

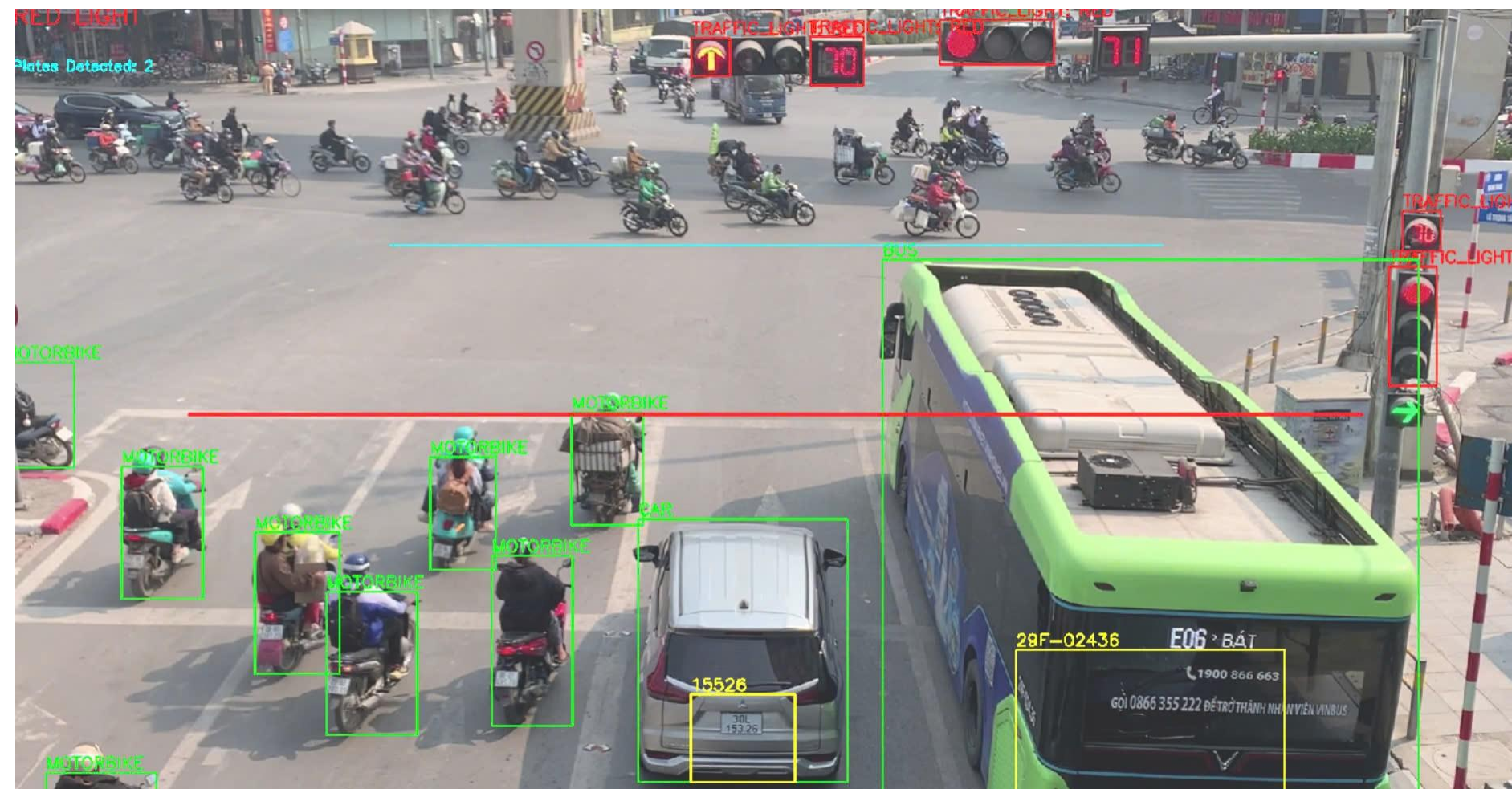


**Kết quả thực nghiệm model  
nhận diện vật thể**



# 11.Kết quả cuối cùng

- Nhận diện và phân loại phương tiện  
Các phương tiện như xe máy (MOTORBIKE), xe ô tô (CAR), xe buýt (BUS) đều được hệ thống nhận diện và đánh dấu bằng khung chữ nhật màu xanh lá cây.
- Phát hiện vi phạm đèn đỏ  
Đèn tín hiệu giao thông ở góc phải của ảnh đang hiển thị đèn đỏ.  
Có một số xe máy vượt qua vạch dừng khi đèn đỏ, cho thấy khả năng vi phạm giao thông.
- Hệ thống có chức năng đếm số lượng biển số xe  
Đường giới hạn và vùng kiểm soát  
Có các đường màu đỏ thể hiện vạch dừng khi đèn đỏ.  
Một số xe máy và ô tô đang đứng trước vạch dừng, trong khi một số xe đã vượt qua.  
Dữ liệu thời gian đèn tín hiệu ở phía trên hiển thị 71 giây và 70 giây
- Ứng dụng trong giao thông thông minh  
Hệ thống có thể được triển khai để:  
Tự động ghi nhận biển số của các xe vi phạm.  
Gửi cảnh báo đến trung tâm điều khiển giao thông. Tích hợp với hệ thống xử phạt nguội để xử lý vi phạm.





# 12. Kết luận

## Hiệu suất mô hình

- Sau quá trình huấn luyện, mô hình **YOLOv8** đã đạt hiệu suất tốt trong việc nhận diện đối tượng, cho thấy khả năng phát hiện với độ chính xác cao trong phần lớn trường hợp.
- Tuy nhiên, mô hình vẫn gặp phải một số hạn chế, đặc biệt khi môi trường ánh sáng thay đổi, hoặc khi đối tượng có đặc điểm không rõ ràng, dễ gây nhầm lẫn với các lớp khác.
- Những lỗi nhận diện thường xảy ra khi các mẫu có sự tương đồng cao hoặc ở trạng thái khó phân biệt, như vật thể bị che khuất một phần hoặc có biến dạng nhẹ.

## Hướng cải thiện

- **Mở rộng và cân bằng tập dữ liệu:** Bổ sung nhiều mẫu đa dạng hơn để giúp mô hình học được các đặc điểm khác nhau, hạn chế tình trạng overfitting hoặc nhận diện sai.
- **Tối ưu hóa tham số mô hình:** Điều chỉnh siêu tham số như learning rate, batch size, số epoch để cải thiện độ chính xác mà không làm tăng thời gian huấn luyện quá mức.
- **Kết hợp phương pháp khác:** Có thể sử dụng thêm **CNN** hoặc **Transformer-based models** để hỗ trợ việc trích xuất đặc trưng chi tiết hơn, giúp giảm thiểu sai sót trong nhận diện.



*Thank You*