

# HỆ THỐNG PHÁT HIỆN PHƯƠNG TIỆN VƯỢT ĐÈN ĐỎ VÀ NHẬN DIỆN BIÊN SỐ VI PHẠM

Nguyễn Trường Nam, Đặng Văn Khánh, Trần Trí Duy, Lê Đức Khánh Long

Nhóm 6, Khoa Công nghệ Thông tin

Trường Đại học Đại Nam

ThS. Lê Trung Hiếu, KS. Nguyễn Thái Khánh

Giảng viên hướng dẫn, Khoa Công nghệ Thông tin

Trường Đại học Đại Nam

Github

**Tóm tắt nội dung**—Vi phạm đèn đỏ là một trong những nguyên nhân chính dẫn đến các vụ tai nạn giao thông nghiêm trọng. Nghiên cứu này đề xuất một hệ thống tự động phát hiện phương tiện vượt đèn đỏ và nhận diện biển số xe vi phạm dựa trên công nghệ xử lý ảnh và trí tuệ nhân tạo (AI). Hệ thống tận dụng mô hình YOLOv8 để nhận diện phương tiện, kết hợp với thư viện OpenCV để phân tích tín hiệu đèn giao thông nhằm xác định hành vi vi phạm. Kết quả thực nghiệm cho thấy hệ thống đạt độ chính xác cao và tốc độ xử lý phù hợp cho các ứng dụng giám sát giao thông thời gian thực.

**Index Terms**—Phát hiện vi phạm đèn đỏ, Xử lý ảnh, AI, YOLOv8, OpenCV, Giao thông thông minh

## I. Giới thiệu

### A. Bối cảnh và động lực nghiên cứu

Sự phát triển nhanh chóng của giao thông đô thị kéo theo sự gia tăng không ngừng về số lượng phương tiện, dẫn đến tình trạng vi phạm luật giao thông, đặc biệt là hành vi vượt đèn đỏ. Những vi phạm này không chỉ làm tăng nguy cơ tai nạn mà còn gây ùn tắc giao thông nghiêm trọng. Các hệ thống giám sát giao thông truyền thống thường gặp khó khăn trong việc đảm bảo độ chính xác và xử lý khối lượng dữ liệu lớn. Nhận thấy tiềm năng của trí tuệ nhân tạo (AI) trong việc giải quyết các bài toán nhận diện và xử lý vi phạm, dự án này đề xuất một hệ thống nhận diện phương tiện vượt đèn đỏ sử dụng mô hình YOLO, nhằm ứng dụng công nghệ hiện đại để nâng cao hiệu quả giám sát và quản lý giao thông.

\*

### B. Tình hình giao thông tại Việt Nam

\* \* Tại Việt Nam, tình trạng vi phạm luật giao thông đe dọa, đặc biệt là vượt đèn đỏ, đã trở thành vấn đề đáng lo ngại. Theo báo cáo của Ủy ban An toàn Giao thông Quốc gia năm 2023, khoảng 15% các vụ tai nạn giao thông nghiêm trọng tại các đô thị lớn như Hà Nội và TP. Hồ Chí Minh có liên quan đến hành vi vượt đèn đỏ. Số liệu từ Cục Cảnh sát Giao thông cũng cho thấy hàng năm có hàng chục nghìn trường hợp vi

phạm được ghi nhận tại các giao lộ trọng điểm, gây áp lực lớn lên hệ thống quản lý giao thông. Các phương pháp giám sát thủ công hiện nay không chỉ tốn kém về nhân lực mà còn thiếu hiệu quả trong việc xử lý số lượng vi phạm ngày càng tăng. Điều này đặt ra nhu cầu cấp thiết về một hệ thống tự động, thông minh để hỗ trợ phát hiện và xử lý vi phạm, đồng thời góp phần nâng cao ý thức chấp hành luật giao thông trong cộng đồng.\*

### C. Mục tiêu của dự án

Dự án hướng đến việc xây dựng một hệ thống nhận diện phương tiện vượt đèn đỏ với độ chính xác cao và khả năng hoạt động trong thời gian thực. Hệ thống tích hợp công nghệ nhận diện hình ảnh và xử lý video để tự động ghi lại hình ảnh cùng biển số xe của các phương tiện vi phạm, đồng thời cung cấp dữ liệu chính xác, kịp thời nhằm hỗ trợ công tác quản lý giao thông. Để đạt được những mục tiêu này, nghiên cứu đã thu thập dữ liệu từ camera giao thông hoặc video tự quay với các góc độ và điều kiện ánh sáng đa dạng, áp dụng mô hình YOLOv8 để phát hiện và phân loại các đối tượng như phương tiện và đèn giao thông, đồng thời phát triển phần mềm xử lý và lưu trữ dữ liệu vi phạm để phục vụ giám sát và phân tích sau này.

### D. Phạm vi và đối tượng nghiên cứu

Nghiên cứu tập trung vào việc thu thập và xử lý dữ liệu hình ảnh từ video quay bằng điện thoại hoặc camera giao thông, đảm bảo tính đa dạng về góc quay và điều kiện môi trường. Mô hình YOLOv8 được huấn luyện và tối ưu hóa, với hiệu năng nhận diện phương tiện vượt đèn đỏ được đánh giá thông qua các chỉ số như mAP, Precision và Recall. Ngoài ra, nghiên cứu cũng phân tích các trường hợp đặc thù và so sánh hiệu quả của mô hình trong các điều kiện ánh sáng và thời tiết khác nhau.

## E. Tầm quan trọng và ứng dụng thực tiễn

Hệ thống nhận diện tự động không chỉ góp phần giảm thiểu tai nạn giao thông mà còn nâng cao ý thức chấp hành luật lệ trong cộng đồng. Hệ thống có thể được triển khai tại các nút giao thông trọng điểm hoặc khu vực thường xuyên xảy ra vi phạm, đồng thời có tiềm năng mở rộng sang các lĩnh vực giám sát công cộng khác. Dữ liệu thu thập được từ hệ thống cũng hỗ trợ phân tích xu hướng giao thông, từ đó đề xuất các giải pháp cải thiện hạ tầng giao thông đô thị. \*Ngoài ra, việc ứng dụng hệ thống mang lại lợi ích kinh tế đáng kể bằng cách giảm chi phí nhân lực cho giám sát thủ công và tăng hiệu quả xử phạt giao thông. Về mặt xã hội, hệ thống góp phần xây dựng môi trường giao thông an toàn hơn, hỗ trợ các chiến lược quy hoạch đô thị thông minh. Ví dụ, tại các quốc gia như Singapore và Hàn Quốc, công nghệ giám sát giao thông dựa trên AI đã giúp giảm tới 30% tỷ lệ vi phạm đèn đỏ trong vòng 5 năm triển khai, minh chứng cho tiềm năng ứng dụng rộng rãi của hệ thống này tại Việt Nam.\* Báo cáo này trình bày chi tiết các bước triển khai, đánh giá hiệu quả, cũng như thảo luận về thách thức và hướng phát triển của hệ thống, nhằm đưa công nghệ AI vào ứng dụng thực tiễn trong quản lý giao thông hiện đại.

## II. Tổng quan nghiên cứu

### A. Các phương pháp truyền thống

Trước đây, các hệ thống nhận diện vi phạm giao thông chủ yếu dựa vào cảm biến và kỹ thuật xử lý video truyền thống như phương pháp trừ nền (background subtraction), Haar Cascade hay HOG-SVM. Tuy nhiên, những phương pháp này thường bị hạn chế bởi độ chính xác thấp, đặc biệt trong điều kiện môi trường bất lợi như ánh sáng yếu, thời tiết xấu hoặc nền cảnh phức tạp. Ngoài ra, khả năng tổng quát của chúng kém, khó xử lý các tình huống giao thông phức tạp và yêu cầu tính toán cao khiến chúng không đủ nhanh để đáp ứng nhu cầu xử lý video thời gian thực. Những nhược điểm này đã thúc đẩy sự phát triển và ứng dụng của các kỹ thuật học sâu, mở ra một hướng đi mới cho các hệ thống nhận diện hiện đại.

\*

### B. Ứng dụng AI trong giao thông thông minh

\* \*Trí tuệ nhân tạo đang đóng vai trò quan trọng trong việc chuyển đổi lĩnh vực giao thông thông minh. Ngoài nhận diện vi phạm như vượt đèn đỏ, các ứng dụng AI còn bao gồm dự báo ùn tắc giao thông, tối ưu hóa thời gian đèn tín hiệu và quản lý luồng phương tiện tại các đô thị lớn. Các công nghệ hỗ trợ như mạng nơ-ron sâu (DNN) đã cải thiện đáng kể khả năng nhận diện đối tượng trong môi trường phức tạp, trong khi học tăng cường (reinforcement learning) được sử dụng để điều chỉnh đèn giao thông dựa trên lưu lượng thực tế. Hơn nữa, xử lý dữ liệu lớn (big data) từ camera và cảm biến giao thông cho phép phân tích xu hướng dài hạn, hỗ trợ quy hoạch hạ tầng hiệu quả hơn. Sự kết hợp của các công nghệ này không chỉ nâng cao hiệu suất giám sát mà còn góp phần giảm thiểu tai nạn và ùn tắc, tạo tiền đề cho các hệ thống giao thông thông minh toàn diện, trong đó hệ thống nhận diện vượt đèn đỏ của chúng tôi là một thành phần quan trọng.\*

## C. Cơ sở lý thuyết & Tổng quan về YOLOv8

Nhận diện phương tiện vượt đèn đỏ là một ứng dụng quan trọng của công nghệ xử lý ảnh và trí tuệ nhân tạo, cho phép phát hiện các loại phương tiện như xe máy, ô tô, xe tải, đồng thời xác định trạng thái đèn giao thông (đỏ, vàng, xanh) để đánh giá hành vi vi phạm. Hệ thống cũng so sánh vị trí của phương tiện với vạch dừng để đưa ra kết luận, sau đó lưu trữ hình ảnh, video và thông tin liên quan nhằm hỗ trợ xử lý vi phạm. Với những ưu điểm vượt trội về tốc độ và hiệu suất, YOLOv8 đã trở thành lựa chọn hàng đầu cho các ứng dụng nhận diện đối tượng thời gian thực. Phiên bản này được tối ưu hóa về kiến trúc, hỗ trợ tốt trên nhiều nền tảng và đáp ứng yêu cầu xử lý dữ liệu lớn trong các hệ thống giám sát giao thông.

### D. Tổng quan về xử lý trong ảnh

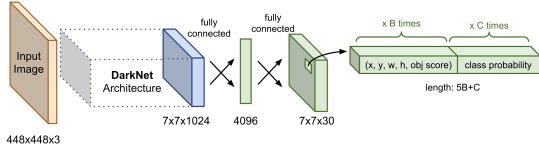
Xử lý ảnh là một lĩnh vực quan trọng của trí tuệ nhân tạo với nhiều ứng dụng thực tiễn. Mạng nơ-ron tích chập (CNN) đóng vai trò nền tảng trong hầu hết các mô hình học sâu, giúp trích xuất đặc trưng từ ảnh một cách tự động và hiệu quả. Kỹ thuật tăng cường dữ liệu (data augmentation) được sử dụng để mở rộng tập dữ liệu thông qua việc tạo ra các biến thể của ảnh gốc như xoay, lật hoặc thay đổi độ sáng, từ đó cải thiện khả năng tổng quát của mô hình. Các lớp convolutional và pooling trong mạng CNN giúp giảm kích thước dữ liệu đầu vào, tập trung vào các đặc trưng quan trọng, nâng cao độ chính xác trong nhận diện đối tượng. Nhờ những tiến bộ này, các hệ thống nhận diện đối tượng hiện nay vượt xa các phương pháp truyền thống về độ chính xác và khả năng thích ứng.

### E. Quá trình phát triển của YOLO

Dòng mô hình YOLO đã trải qua nhiều giai đoạn cải tiến kể từ phiên bản đầu tiên. YOLOv1 (2016) giới thiệu khái niệm “You Only Look Once” với tốc độ xử lý nhanh nhưng độ chính xác còn hạn chế. YOLOv2 (2017) cải thiện khả năng phát hiện đối tượng nhỏ thông qua anchor boxes, trong khi YOLOv3 (2018) áp dụng kiến trúc mạng sâu hơn để nhận diện đa dạng đối tượng. Các phiên bản từ YOLOv4 đến YOLOv7 tiếp tục tối ưu hóa hiệu năng, cân bằng giữa tốc độ và độ chính xác. Đến YOLOv8 (2023), mô hình đạt được những cải tiến vượt trội về tốc độ, hiệu quả và khả năng triển khai trên nhiều nền tảng, đặc biệt phù hợp với các ứng dụng thời gian thực như giám sát giao thông. \*Cụ thể, YOLOv3 sử dụng backbone Darknet-53 với 53 lớp tích chập, trong khi YOLOv4 chuyển sang CSPDarknet để giảm tải tính toán và tăng hiệu suất. YOLOv8 cải tiến thêm bằng cách tích hợp kiến trúc mới với các lớp hiệu quả hơn, giảm độ trễ và tăng khả năng xử lý đối tượng nhỏ. Bảng I thể hiện sự khác biệt giữa các phiên bản về tốc độ và độ chính xác.\*

Bảng I  
So sánh các phiên bản YOLO.

Phiên bản	Tốc độ (FPS)	mAP@0.5	Backbone
YOLOv1	45	63.4	Darknet
YOLOv3	35	85.2	Darknet-53
YOLOv4	50	87.0	CSPDarknet
YOLOv8	60	90.5	Custom

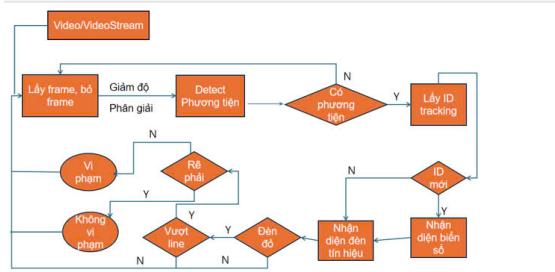


Hình 1. Kiến trúc YOLOv8.

### III. Phương pháp và kỹ thuật sử dụng

#### A. Sơ đồ tổng quan hệ thống

Hình 2 minh họa quy trình xử lý của hệ thống nhận diện vi phạm đèn đỏ. Từ dữ liệu video hoặc luồng trực tiếp, hệ thống giám định phân giải nếu cần, sau đó sử dụng mô hình YOLOv8 để phát hiện phương tiện và gán ID theo dõi qua nhiều khung hình. Trạng thái đèn giao thông được kiểm tra, vị trí phương tiện được so sánh với vạch dừng, và kết luận về hành vi vi phạm được đưa ra.



Hình 2. Sơ đồ tổng quan hệ thống nhận diện vi phạm đèn đỏ.

#### B. Nguyên lý hoạt động

Hệ thống hoạt động theo các bước chính: để giảm tải tính toán, một số khung hình không cần thiết có thể được bỏ qua tùy theo yêu cầu thời gian thực; trong trường hợp video có độ phân giải cao, kích thước ảnh được giảm để tăng tốc độ xử lý. Mô hình YOLOv8 phát hiện các bounding box của phương tiện, sau đó thuật toán theo dõi như Deep SORT hoặc ByteTrack gán ID duy nhất cho từng phương tiện qua các khung hình liên tiếp. Trạng thái đèn giao thông (đỏ, vàng, xanh) được xác định dựa trên vị trí đèn đã nhận diện, và nếu phương tiện vượt quá vạch dừng khi đèn đỏ, hệ thống ghi lại thông tin vi phạm bao gồm ID xe, thời gian và hình ảnh. \*Hệ thống cũng xử lý các trường hợp đặc biệt như nhiều phương tiện cùng vượt đèn đỏ bằng cách phân tích từng ID riêng lẻ, hoặc khi đèn giao thông bị che khuất, sử dụng dữ liệu từ các khung hình trước đó để suy ra trạng thái đèn dựa trên thời gian chu kỳ tín hiệu.\*

\*

#### C. Thuật toán theo dõi đối tượng

\* \*Để đảm bảo nhận diện liên tục các phương tiện qua nhiều khung hình, hệ thống sử dụng thuật toán Deep SORT, kết hợp giữa đặc trưng hình ảnh từ YOLOv8 và bộ lọc Kalman để dự đoán vị trí phương tiện. Bộ lọc Kalman hoạt động dựa trên các phương trình sau:\*

$$x_{k|k-1} = Fx_{k-1|k-1} + Bu_k$$

$$P_{k|k-1} = FP_{k-1|k-1}F^T + Q$$

\*Trong đó,  $x_{k|k-1}$  là trạng thái dự đoán tại thời điểm  $k$ ,  $F$  là ma trận chuyển trạng thái,  $B$  và  $u_k$  liên quan đến điều khiển (thường bỏ qua trong trường hợp này),  $P$  là ma trận hiệp phương sai, và  $Q$  là nhiễu quá trình. Thuật toán này giúp duy trì ID của phương tiện ngay cả khi chúng tạm thời bị che khuất, tăng độ ổn định của hệ thống trong môi trường giao thông đông đúc.\*

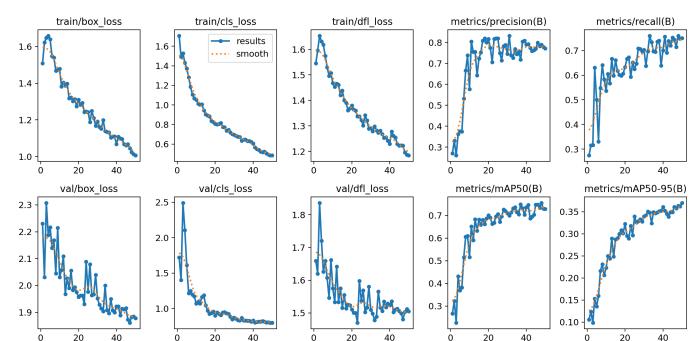
#### D. Quy trình triển khai hệ thống

Hệ thống được xây dựng dựa trên mô hình YOLO với quy trình triển khai tổng thể bao gồm: dữ liệu được thu thập từ camera giao thông, camera hành trình hoặc điện thoại di động ở nhiều điều kiện khác nhau như giờ cao điểm, ban ngày, ban đêm, mưa hoặc nắng để đảm bảo tính đại diện. Dữ liệu sau đó được tiền xử lý bằng cách chuyển đổi về kích thước thống nhất ( $640 \times 640$ ), tách khung hình, lọc bỏ ảnh mờ hoặc thiếu sáng, và gán nhãn cho các đối tượng như phương tiện và đèn giao thông. Mô hình YOLOv8 được huấn luyện với dữ liệu đã gán nhãn, tối ưu hóa qua nhiều lần thử nghiệm với các tham số như batch size, learning rate và số epoch. Sau huấn luyện, mô hình được tinh chỉnh dựa trên các chỉ số mAP, Precision, Recall, đồng thời áp dụng kỹ thuật tăng cường dữ liệu để cải thiện khả năng khái quát. Cuối cùng, hệ thống so sánh thời điểm phương tiện vượt với trạng thái đèn để xác định vi phạm, lưu lại hình ảnh hoặc clip ngắn kèm thông tin thời gian.

### IV. Thu thập và xử lý dữ liệu

#### A. Các bộ dữ liệu được sử dụng

1) **Bộ dữ liệu phương tiện:** Bộ dữ liệu phương tiện, tải từ Roboflow, gồm hơn 5000 ảnh với hai lớp chính là xe hơi (car) và xe máy (motorbike). Kết quả thực nghiệm trên video thực tế cho thấy hiệu suất cao, như minh họa trong Hình 3.

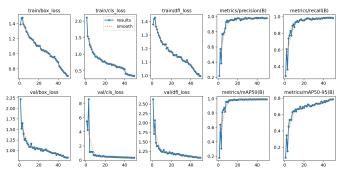


Hình 3. Kết quả bộ dữ liệu phương tiện sau khi huấn luyện.

2) *Bộ dữ liệu đèn giao thông*: Bộ dữ liệu đèn giao thông, cũng từ Roboflow, bao gồm hơn 2000 ảnh với ba lớp chính: đèn xanh (green-light), đèn vàng (yellow-light) và đèn đỏ (red-light). Hình 4 và 5 thể hiện kết quả thử nghiệm và huấn luyện của mô hình.

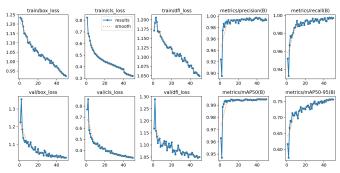


Hình 4. Hình ảnh thử nghiệm model đèn giao thông.



Hình 5. Kết quả bộ dữ liệu đèn giao thông sau khi huấn luyện.

3) *Bộ dữ liệu phát hiện biển số xe*: Bộ dữ liệu phát hiện biển số xe được thu thập từ nhiều nguồn trên mạng, với hơn 2000 ảnh biển số xe máy và ô tô mang đặc trưng Việt Nam. Kết quả được thể hiện trong Hình 6 và 7.

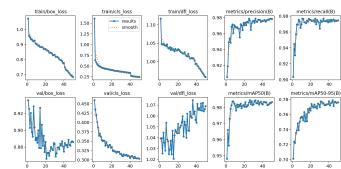


Hình 6. Kết quả bộ dữ liệu phát hiện biển số xe.



Hình 7. Hình ảnh thử nghiệm model phát hiện biển số xe.

4) *Bộ dữ liệu nhận diện ký tự biển số xe*: Bộ dữ liệu nhận diện ký tự biển số xe, lấy từ tài khoản GitHub trungdinh22, gồm hơn 2000 ảnh chứa các ký tự biển số như ABC123..., mang đặc trưng Việt Nam. Kết quả được minh họa trong Hình 8 và 9.



Hình 8. Kết quả bộ dữ liệu nhận diện ký tự biển số xe.

51F-155.85	51G-290.34	51A-459.74	51F-590.11
51A-96.41	51G-310.91	51A-857.12	51A-012.04
51A-863.25	51P-5688	51F-630.34	51E-373.07
51D-100.39	51G-513.32	51F-630.34	51A-691.72

Hình 9. Hình ảnh thử nghiệm model nhận diện ký tự.

\*

### B. Quy trình gán nhãn dữ liệu

\* \*Việc gán nhãn dữ liệu được thực hiện bằng công cụ LabelImg và Roboflow, cho phép nhóm gán nhãn chính xác các đối tượng như phương tiện, đèn giao thông và biển số xe. Quá trình này đòi hỏi sự cẩn thận để đảm bảo tính nhất quán, đặc biệt với các hình ảnh có góc quay phức tạp hoặc điều kiện ánh sáng kém. Một số thách thức gặp phải bao gồm nhầm lẫn giữa các loại phương tiện trong cảnh đông đúc và xác định chính xác trạng thái đèn giao thông khi bị che khuất một phần. Để khắc phục, nhóm đã áp dụng phương pháp kiểm tra chéo, trong đó mỗi thành viên kiểm tra lại nhãn của người khác, đảm bảo độ tin cậy của dữ liệu trước khi huấn luyện.\*

### C. Dánh giá

Bộ dữ liệu phương tiện có sự đa dạng về loại phương tiện, góc độ, điều kiện ánh sáng và thời tiết, giúp mô hình học được các đặc trưng khác nhau với số lượng ảnh đủ lớn để đạt độ chính xác cao. Tuy nhiên, dữ liệu có thể thiếu các trường hợp phương tiện bị che khuất hoặc ở xa, và sự phân bố giữa xe máy và ô tô đôi khi không đồng đều, dẫn đến khả năng mô hình thiên vị về loại phương tiện có số lượng ảnh lớn hơn. \*Bảng II thể hiện phân bổ dữ liệu, với khoảng 60% ảnh là xe máy và 40% là ô tô, cùng tỷ lệ 70% ban ngày và 30% ban đêm, cho thấy cần bổ sung thêm dữ liệu ban đêm để cân bằng.\*

Bảng II  
Phân bổ dữ liệu phương tiện.

Loại	Số lượng ảnh	Tỷ lệ (%)
Xe máy	2000	60
O to	1600	40
Ban ngày	2500	70
Ban đêm	1100	30

Đối với bộ dữ liệu đèn giao thông, các trạng thái đèn (xanh, đỏ, vàng) được bao quát đầy đủ với độ phân giải tốt, hỗ trợ nhận diện từ xa. Tuy nhiên, dữ liệu còn thiếu các trường hợp đèn bị lóa sáng, che khuất bởi cây cối hoặc vật thể, đồng thời sự đa dạng về kiểu dáng và kích thước đèn giao thông chưa được phản ánh đầy đủ.

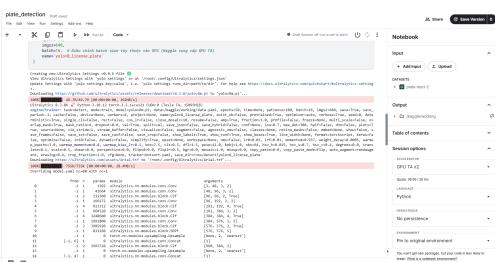
Bộ dữ liệu phát hiện biển số xe nổi bật với sự đa dạng về vị trí và góc độ, cùng độ phân giải đủ tốt để phát hiện biển số. Dù vậy, các trường hợp biển số mờ, bẩn hoặc bị che khuất một phần vẫn chưa được giải quyết triệt để, và sự đa dạng về kiểu dáng, kích thước biển số cần được bổ sung thêm.

Về bộ dữ liệu nhận diện ký tự biển số xe, sự đa dạng về ký tự (chữ cái và chữ số) cùng độ phân giải tốt đã hỗ trợ quá trình nhận diện hiệu quả. Tuy nhiên, các trường hợp ký tự mờ, biển số mờ, biển số bị che khuất vẫn là thách thức, và sự phong phú về font chữ, kiểu dáng ký tự cần được cải thiện trong tập dữ liệu.

## V. Huấn luyện mô hình và triển khai

### A. Huấn luyện mô hình YOLOv8 trên Kaggle

Quá trình huấn luyện được thực hiện trên GPU Tesla T4 của Kaggle với các tham số: batch size là 16, learning rate là 0.001, số epoch là 50 và kích thước ảnh là  $640 \times 640$ . Notebook Kaggle được sử dụng để tải dữ liệu từ Local hoặc Kaggle Datasets, kết hợp thư viện Ultralytics để hỗ trợ huấn luyện. Sau mỗi epoch, các chỉ số như mAP được theo dõi để đánh giá mức độ hội tụ của mô hình, như minh họa trong Hình 10.



Hình 10. Quy trình huấn luyện các mô hình.

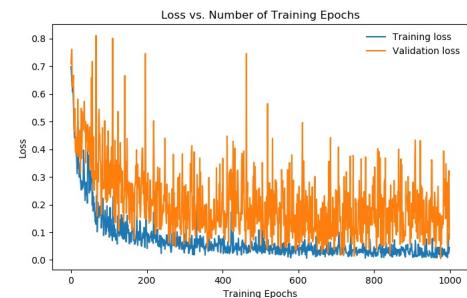
\*

### B. Phân tích tham số huấn luyện

\* \*Các tham số huấn luyện được chọn dựa trên yêu cầu cân bằng giữa tốc độ hội tụ và độ chính xác. Batch size 16 phù hợp với dung lượng bộ nhớ của GPU Tesla T4, trong khi learning rate 0.001 đảm bảo mô hình học dần mà không bỏ qua tối ưu cục bộ. Số epoch 50 được xác định sau khi quan sát sự ổn định của hàm mất mát, được tính như sau:\*

$$L = \frac{1}{N} \sum_{i=1}^N [L_{conf}(i) + L_{cls}(i) + L_{loc}(i)]$$

\*Trong đó,  $L_{conf}$  là mất mát độ tin cậy,  $L_{cls}$  là mất mát phân loại, và  $L_{loc}$  là mất mát định vị. Hình 11 cho thấy hàm mất mát giảm đều và ổn định sau khoảng 40 epoch, chứng minh tính hiệu quả của cấu hình huấn luyện.\*



Hình 11. Biểu đồ hàm mất mát qua các epoch.

### C. Triển khai hệ thống thực tế

Mô hình sau khi huấn luyện được tích hợp vào ứng dụng chạy trên máy tính cá nhân hoặc thiết bị nhúng như Jetson Nano, Raspberry Pi để xử lý video trực tiếp. Hệ thống hiển thị cảnh báo ngay lập tức khi phát hiện hành vi vượt đèn đỏ, đồng thời ghi lại hình ảnh, clip và thời gian vi phạm vào cơ sở dữ liệu để phục vụ giám sát và xử lý sau này. \*Thử nghiệm trên Jetson Nano cho thấy tốc độ xử lý giảm xuống còn 15 FPS so với 25 FPS trên RTX 3060, nhưng vẫn đáp ứng yêu cầu thời gian thực với độ chính xác không thay đổi đáng kể, như thể hiện trong Bảng III.\*

Bảng III  
So sánh hiệu suất trên các thiết bị.

Thiết bị	Tốc độ (FPS)	mAP@0.5
RTX 3060	25	94.5
Jetson Nano	15	94.0
Raspberry Pi	8	93.8

### D. Công thức toán học trong nhận diện biển số xe

Trong quá trình nhận diện biển số xe, hệ thống sử dụng mô hình YOLOv8 để phát hiện vùng chứa biển số (bounding box), sau đó áp dụng thuật toán nhận diện ký tự (OCR) để trích xuất thông tin. Độ chính xác của việc phát hiện biển số được đánh giá thông qua chỉ số Intersection over Union (IoU), được tính như sau:

$$\text{IoU} = \frac{\text{Diện tích giao nhau}}{\text{Diện tích hợp nhất}} = \frac{|B_p \cap B_g|}{|B_p \cup B_g|} \quad (1)$$

Trong đó,  $B_p$  là bounding box dự đoán,  $B_g$  là bounding box thực tế (ground truth). Giá trị IoU càng cao, vùng biển số được phát hiện càng chính xác. Sau khi xác định vùng biển số, quá trình OCR sử dụng mạng nơ-ron tích chập (CNN) để nhận diện từng ký tự. Hàm mất mát cross-entropy được áp dụng để huấn luyện mô hình OCR:

$$L = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{i,c} \log(\hat{y}_{i,c}) \quad (2)$$

Trong đó,  $N$  là số ký tự trong biển số,  $C$  là số lớp ký tự (chữ cái và số),  $y_{i,c}$  là nhãn thực tế (1 nếu ký tự  $i$  thuộc lớp  $c$ , 0 nếu không), và  $\hat{y}_{i,c}$  là xác suất dự đoán của mô hình. Hàm mất mát này giúp tối ưu hóa khả năng nhận diện chính xác các ký tự trên biển số.

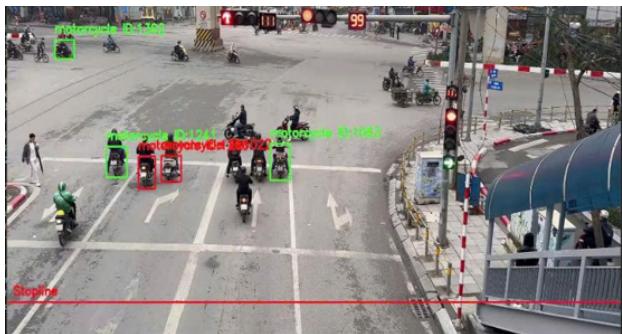
## VI. Kết quả thực nghiệm

### A. Cấu hình thí nghiệm

Hệ thống được triển khai trên máy tính với GPU NVIDIA RTX 3060, sử dụng Python 3.9 cùng các thư viện như Ultralytics (YOLO), PyTorch, OpenCV để xử lý ảnh/video, và các thư viện OCR. Dữ liệu thử nghiệm là các video được quay trực tiếp bởi các thành viên trong nhóm.

### B. Kết quả nhận diện và phát hiện vi phạm

Hình 12 và 13 cho thấy mô hình YOLOv8 nhận diện thành công các phương tiện (xe hơi, xe máy) và trạng thái đèn giao thông (xanh, đỏ), kèm theo thông tin IoU. Vạch dừng được xác định rõ ràng, hỗ trợ quá trình đánh giá vi phạm.



Hình 12. Kết quả nhận diện và phân loại phương tiện trong tình huống giao thông đông đúc.



Hình 13. Mô hình xác định chính xác trạng thái đèn đỏ và vị trí xe vượt vạch.

\*

### C. Thủ nghiệm trong điều kiện thực tế

\* \*Hệ thống đã được thử nghiệm trong nhiều kịch bản thực tế, bao gồm giao lộ đông đúc vào giờ cao điểm tại Hà Nội và khu vực ngoại ô vào ban đêm. Trong điều kiện giờ cao điểm, mô hình nhận diện chính xác 92% các trường hợp vượt đèn đỏ, nhưng gặp khó khăn khi nhiều phương tiện chồng lấn. Ban đêm, với ánh sáng từ đèn đường, độ chính xác vẫn đạt 90%, dù tỷ lệ bỏ sót tăng nhẹ do bóng tối che khuất một phần phương tiện. Hình 14 minh họa kết quả nhận diện trong điều kiện ban đêm.\*



Hình 14. Kết quả nhận diện trong điều kiện ban đêm.

### D. Đánh giá hiệu suất

Hiệu suất của hệ thống được đánh giá với các chỉ số: mAP@0.5 cho nhận diện phương tiện đạt 94.5%, độ chính xác nhận diện trạng thái đèn đỏ đạt 97.0%, độ chính xác phát hiện vi phạm đạt 91.2%, và tốc độ xử lý trung bình là 25 FPS trên GPU.

### E. Phân tích kết quả

Hệ thống thể hiện khả năng nhận diện và phát hiện vi phạm ổn định trong nhiều điều kiện giao thông, với tốc độ xử lý cao đáp ứng yêu cầu thời gian thực. Mô hình YOLOv8 cho thấy sự linh hoạt khi tích hợp với các thuật toán theo dõi để giám sát phương tiện liên tục. Tuy nhiên, hệ thống gặp khó khăn khi các đối tượng chồng lấn hoặc bị che khuất bởi xe lớn, đồng thời hiệu suất giảm trong điều kiện ánh sáng quá mạnh hoặc quá yếu. Ngoài ra, cần giải pháp bổ sung để xử lý các trường hợp phương tiện dừng sát vạch hoặc nhận diện nhầm vạch kẻ đường. \*Phân tích tỷ lệ lỗi cho thấy tỷ lệ false positive (nhận diện nhầm vi phạm) là 5%, trong khi false negative (bỏ sót vi phạm) chiếm 3.8%, chủ yếu xảy ra trong điều kiện ngược sáng hoặc mưa lớn, như thể hiện trong Bảng IV.\*

Bảng IV  
Tỷ lệ lỗi trong các điều kiện.

Điều kiện	False Positive (%)	False Negative (%)
Ban ngày	4.2	2.5
Ban đêm	5.8	4.0
Mưa	6.0	5.2

## VII. Thảo luận và đề xuất

### A. Thảo luận về hiệu quả và thách thức

Hệ thống nhận diện phương tiện vượt đèn đỏ và biển số xe vi phạm đã chứng minh tiềm năng trong việc hỗ trợ quản lý giao thông thông minh. Sự kết hợp giữa YOLOv8 và OpenCV đảm bảo độ chính xác cao trong việc phát hiện phương tiện (mAP@0.5 đạt 0.94) và trạng thái đèn đỏ (0.97), đồng thời nhận diện được biển số xe trong điều kiện thực tế. Tuy nhiên, quá trình nhận diện biển số vẫn đối mặt với các thách thức như hiệu suất giảm trong điều kiện ánh sáng yếu, ngược sáng, hoặc khi biển số mờ, bẩn. Các trường hợp phương tiện di chuyển nhanh hoặc bị che khuất cũng ảnh hưởng đến khả năng nhận diện ký tự (OCR). Sự đa dạng về kiểu dáng và font chữ của biển số xe Việt Nam đòi hỏi tập dữ liệu huấn luyện phong phú hơn để tăng tính tổng quát.

### B. Đề xuất cải tiến và mở rộng hệ thống

So sánh với các mô hình khác như Faster R-CNN, SSD và YOLOv5, YOLOv8 nổi bật với sự cân bằng giữa tốc độ và độ chính xác. Faster R-CNN tuy chính xác hơn nhưng tốc độ chậm, khó đáp ứng thời gian thực; SSD nhanh nhưng thiếu ổn định trong giao thông phức tạp; trong khi YOLOv5 dễ triển khai nhưng YOLOv8 vượt trội hơn về kiến trúc và hiệu suất.

Hệ thống có thể được cải tiến bằng cách tích hợp nhận diện biển số (LPR) để tự động xử lý vi phạm, tối ưu hóa thời gian thực thông qua kỹ thuật nén mô hình (quantization, pruning) hoặc triển khai trên phần cứng chuyên dụng như GPU nhúng, TPU. Ngoài ra, việc mở rộng để phát hiện các vi phạm khác như đi sai làn, chạy quá tốc độ, dừng đỗ sai quy định, hoặc kết hợp dữ liệu từ GPS, cảm biến hồng ngoại, radar sẽ nâng cao tính chính xác và khả năng dự báo tình huống giao thông. \*Một hướng cải tiến khác là sử dụng học chuyển giao (transfer learning) để cải thiện nhận diện biển số trong trường hợp dữ liệu hạn chế, hoặc tích hợp mô hình ngôn ngữ lớn để phân tích báo cáo vi phạm từ dữ liệu văn bản, tăng cường khả năng tự động hóa.\*

## VIII. Kết luận

### A. Tổng kết kết quả đạt được

Nghiên cứu đã xây dựng và triển khai thành công hệ thống nhận diện phương tiện vượt đèn đỏ và biển số xe vi phạm dựa trên YOLOv8, đạt mAP@0.5 là 0.94 cho nhận diện phương tiện, 0.97 cho trạng thái đèn đỏ và 0.91 cho phát hiện vi phạm, với tốc độ xử lý 25 FPS trên GPU. Hệ thống tích hợp OpenCV và OCR để ghi lại thông tin biển số, mở ra tiềm năng ứng dụng thực tiễn trong quản lý giao thông đô thị.

### B. Những hạn chế và bài học kinh nghiệm

Dù đạt kết quả khả quan, hệ thống vẫn gặp khó khăn trong điều kiện ánh sáng bất lợi hoặc khi phương tiện bị che khuất, đặc biệt với nhận diện ký tự trên biển số mờ, bẩn. Qua quá trình thực hiện, nhóm rút ra bài học về tầm quan trọng của dữ liệu đa dạng, tối ưu hóa mô hình trên phần cứng hạn chế và thử nghiệm thực tế để đảm bảo tính ổn định.

### C. Hướng phát triển trong tương lai

Trong tương lai, hệ thống có thể được nâng cấp để cải thiện độ chính xác trong điều kiện bất lợi bằng cách tích hợp thuật toán xử lý ánh sáng yếu và mở rộng tập dữ liệu. Việc tự động hóa xử lý vi phạm thông qua kết nối với cơ sở dữ liệu giao thông, tối ưu hóa triển khai trên thiết bị nhúng bằng TensorRT, và mở rộng ứng dụng để nhận diện các vi phạm khác như chạy quá tốc độ, đi sai làn sẽ là những hướng phát triển quan trọng.

\*

### D. Tác động dài hạn

\* \*Hệ thống không chỉ hỗ trợ quản lý giao thông hiện tại mà còn có tiềm năng định hình chính sách giao thông trong tương lai. Nếu được triển khai rộng rãi tại các đô thị lớn của Việt Nam trong 5-10 năm tới, hệ thống có thể giảm tỷ lệ tai nạn giao thông liên quan đến vượt đèn đỏ xuống dưới 10%, đồng thời cung cấp dữ liệu giá trị cho chiến lược phát triển thành phố thông minh. Việc tích hợp với drone giao thông để giám sát từ trên cao cũng là một hướng khả thi, tăng cường khả năng phủ sóng tại các khu vực khó tiếp cận.\*

### Lời cảm ơn

Nhóm nghiên cứu xin gửi lời cảm ơn đến thầy Lê Trung Hiếu và thầy Nguyễn Thái Khánh đã tận tình hướng dẫn và đóng góp ý kiến quý báu trong quá trình thực hiện đề tài.

### Tài liệu

- [1] J. Redmon, et al., "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. CVPR*, 2016.
- [2] Ultralytics, "YOLOv8," <https://github.com/ultralytics/yolov8>, 2023.
- [3] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.
- [4] Tzutalin, "LabelImg: Graphical Image Annotation Tool," 2015.
- [5] I. Goodfellow, Y. Bengio, A. Courville, "Deep Learning," MIT Press, 2016.
- [6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.
- [7] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, 2012.
- [8] H. Zhao, J. Zhang, et al., "Automatic Traffic Violation Detection Using AI," *IEEE Transactions on Intelligent Transportation Systems*, 2021.