

TRƯỜNG ĐẠI HỌC ĐẠI NAM

HỆ THỐNG NHẬN DIỆN PHƯƠNG TIỆN VƯỢT ĐÈN ĐỎ SỬ DỤNG YOLO

Học phần: Triển khai, phát triển ứng dụng AIoT

Giảng viên hướng dẫn: Lê Trung Hiếu, Nguyễn Thái Khánh

1st Trần Trí Duy
Khoa Công nghệ thông tin
Trường Đại học Đại Nam
Hà Nội, Việt Nam
MSV:1671020062

2nd Đặng Văn Khánh
Khoa Công nghệ thông tin
Trường Đại học Đại Nam
Hà Nội, Việt Nam
MSV:1671020161

3rd Nguyễn Trường Nam
Khoa Công nghệ thông tin
Trường Đại học Đại Nam
Hà Nội, Việt Nam
MSV:1671020214

4th Lê Đức Khánh Long
Khoa Công nghệ thông tin
Trường Đại học Đại Nam
Hà Nội, Việt Nam
MSV:1671020185

Tóm tắt nội dung—Vi phạm đèn đỏ là nguyên nhân chính gây ra tai nạn giao thông nghiêm trọng. Bài báo này đề xuất một hệ thống tự động nhận diện vi phạm đèn đỏ dựa trên xử lý ảnh và trí tuệ nhân tạo (AI). Hệ thống sử dụng mô hình YOLOv8 để phát hiện phương tiện và OpenCV để xử lý tín hiệu đèn giao thông, từ đó xác định hành vi vi phạm. Kết quả thực nghiệm cho thấy hệ thống có độ chính xác cao và tốc độ xử lý phù hợp cho ứng dụng giám sát thời gian thực.

Index Terms—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

Trong bối cảnh giao thông ngày càng phát triển, vấn đề vi phạm luật giao thông, đặc biệt là hành vi vượt đèn đỏ, vẫn đang là một thách thức lớn đối với cơ quan quản lý. Việc áp dụng công nghệ trí tuệ nhân tạo (AI) vào giám sát giao thông mang lại tiềm năng to lớn trong việc nâng cao hiệu quả phát hiện và xử lý vi phạm.

B. Mục tiêu của dự án

Dự án “Hệ thống nhận diện phương tiện vượt đèn đỏ sử dụng YOLO” nhằm đạt được các mục tiêu sau:

- Xây dựng hệ thống nhận diện phương tiện vượt đèn đỏ với độ chính xác cao và thời gian thực.
- Ghi lại hình ảnh và biển số xe của phương tiện vi phạm để hỗ trợ xử lý vi phạm giao thông.

Đề xuất phương pháp:

- Sử dụng camera giao thông hoặc video tự quay để thu thập dữ liệu hình ảnh.
- Áp dụng mô hình YOLOv8 để phát hiện phương tiện và nhận diện biển số xe.

- Xây dựng phần mềm xử lý và lưu trữ dữ liệu vi phạm.

C. Phạm vi và đối tượng nghiên cứu

Báo cáo tập trung vào:

- Thu thập và xử lý dữ liệu hình ảnh từ video quay bằng điện thoại hoặc camera giao thông.
- Huấn luyện và tối ưu hóa mô hình YOLOv8 để nhận diện phương tiện vượt đèn đỏ.
- Phân tích, đánh giá hiệu năng của mô hình dưới nhiều điều kiện thực nghiệm khác nhau.

Qua báo cáo này, hệ thống không chỉ giúp phát hiện phương tiện vượt đèn đỏ mà còn đóng góp vào việc nâng cao ý thức chấp hành luật giao thông và hỗ trợ công tác quản lý giao thông hiệu quả hơn.

II. Tổng quan nghiên cứu

A. Các phương pháp truyền thống

Trước đây, các hệ thống nhận diện vi phạm chủ yếu dựa vào cảm biến và xử lý video truyền thống như phương pháp dựa trên chuyển động (background subtraction), Haar Cascade, hay HOG-SVM. Tuy nhiên, các phương pháp này thường không đạt được độ chính xác cao và khó xử lý tốt trong điều kiện ánh sáng kém hoặc khi có nhiều phương tiện giao thông.

B. Cơ Sở Lý thuyết Tổng Quan Về YOLOv8

Nhận dạng xe vượt đèn đỏ Nhận dạng xe vượt đèn đỏ là một ứng dụng quan trọng trong giám sát giao thông thông minh. Công nghệ xử lý ảnh và trí tuệ nhân tạo cho phép phát hiện phương tiện trong thời gian thực, xác định trạng thái đèn giao thông và quyết định xem phương tiện có vi phạm hay không. Các bước chính bao gồm:

- Phát hiện đối tượng (xe máy, ô tô, xe tải, v.v.).
- Xác định trạng thái đèn tín hiệu (đỏ, vàng, xanh).
- Đánh giá vị trí của phương tiện so với vạch dừng.
- Ghi lại hình ảnh và thông tin vi phạm.

C. Tổng quan về xử lý trong ảnh

Học sâu (Deep Learning) là một nhánh của trí tuệ nhân tạo, sử dụng mạng nơ-ron nhân tạo (Neural Networks) để xử lý dữ liệu phức tạp. Trong xử lý ảnh, các mô hình học sâu như CNN (Convolutional Neural Network) có khả năng trích xuất đặc trưng tự động và nhận diện đối tượng với độ chính xác cao.

D. Quá trình phát triển của YOLO

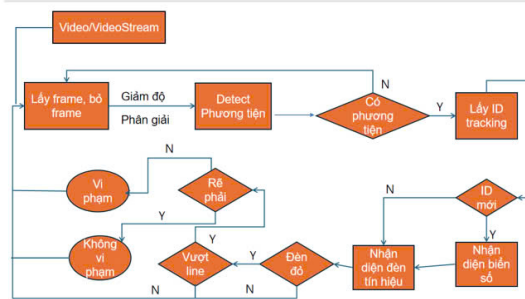
YOLO (You Only Look Once) là một trong những mô hình nhận diện vật thể thời gian thực tiên tiến nhất. Quá trình phát triển của YOLO bao gồm:

- **YOLOv1 (2016):** Mô hình YOLO đầu tiên, mang lại tốc độ nhanh nhưng độ chính xác chưa cao.
- **YOLOv2 (2017):** Cải thiện khả năng phát hiện đối tượng nhỏ và tăng độ chính xác.
- **YOLOv3 (2018):** Nâng cấp với kiến trúc nhiều cấp độ, phát hiện chính xác hơn.
- **YOLOv4, v5, v6, v7:** Liên tục cải tiến về tốc độ và độ chính xác.
- **YOLOv8 (2023):** Phiên bản mới nhất với kiến trúc tối ưu hóa, hiệu suất cao hơn và hỗ trợ tốt trên nhiều nền tảng.

III. PHƯƠNG PHÁP VÀ KỸ THUẬT SỬ DỤNG

A. Sơ đồ tổng quan hệ thống

Hình 1 minh họa sơ đồ tổng quan phương pháp nhận diện vi phạm đèn đỏ. Dữ liệu video từ camera giao thông được trích xuất và gán nhãn, sau đó huấn luyện mô hình YOLOv8. Quá trình suy luận (inference) sẽ kiểm tra trạng thái đèn giao thông và vị trí phương tiện, từ đó phát hiện hành vi vi phạm.



Hình 1. Sơ đồ tổng quan phương pháp nhận diện vi phạm đèn đỏ.

B. Tổng quan quy trình triển khai hệ thống

Hệ thống nhận diện phương tiện vượt đèn đỏ được triển khai theo quy trình sau:

- **Thu thập dữ liệu:** Ghi lại hình ảnh và video từ camera giao thông hoặc điện thoại di động.

- **Tiền xử lý dữ liệu:** Xử lý ảnh/video để trích xuất phương tiện và tín hiệu đèn giao thông.
- **Huấn luyện mô hình:** Sử dụng mô hình YOLOv8 để phát hiện phương tiện và tín hiệu đèn.
- **Tối ưu hóa và tinh chỉnh:** Điều chỉnh các tham số huấn luyện để đạt hiệu năng cao nhất.
- **So sánh thời điểm xe vượt với trạng thái đèn:** Xác định vi phạm bằng cách kiểm tra vị trí của phương tiện khi đèn đỏ bật sáng.

C. Thu thập và xử lý dữ liệu

Dữ liệu được thu thập từ các camera giao thông công cộng và video thực tế. Các hình ảnh và video này được gán nhãn thủ công bằng công cụ MakeSense.AI hoặc LabelImg để xác định vị trí của phương tiện và đèn giao thông. Trong quá trình gán nhãn, chúng tôi đặc biệt chú ý đến các trường hợp xe máy, ô tô, xe tải chen lấn, điều kiện ánh sáng yếu hoặc mưa, nhằm tăng tính đa dạng cho bộ dữ liệu.

D. Tiền xử lý dữ liệu

- Chuyển đổi định dạng video/ảnh về chuẩn đầu vào của mô hình YOLOv8. - Gán nhãn dữ liệu bằng công cụ LabelImg hoặc Roboflow để đánh dấu phương tiện và tín hiệu đèn. - Áp dụng các phương pháp tăng cường dữ liệu (Data Augmentation) như xoay ảnh, thay đổi độ sáng, làm mờ để cải thiện khả năng nhận diện.

E. Huấn luyện mô hình YOLOv8 trên Kaggle

Sau khi có dữ liệu gán nhãn, chúng tôi tiến hành huấn luyện mô hình YOLOv8 trên nền tảng Kaggle. Một số tham số quan trọng:

- **Batch size:** 16
- **Learning rate:** 0.001
- **Số epoch:** 50
- **Kích thước ảnh:** 640x640

Mô hình được triển khai trên GPU (Tesla T4 do Kaggle cung cấp). Chúng tôi sử dụng notebook của Kaggle, tải dữ liệu lên Google Drive hoặc trực tiếp từ Kaggle Datasets, và sử dụng thư viện *ultralytics* để huấn luyện YOLOv8. Sau mỗi epoch, chúng tôi theo dõi chỉ số mAP (mean Average Precision) để đánh giá mức độ hội tụ của mô hình.

F. Triển khai hệ thống thực tế

- Mô hình YOLOv8 được tích hợp vào hệ thống phần mềm để chạy trên máy tính cá nhân hoặc thiết bị nhúng (Jetson Nano, Raspberry Pi). - Hệ thống sẽ xử lý video trực tiếp từ camera và hiển thị cảnh báo vi phạm trong thời gian thực. - Dữ liệu vi phạm được lưu trữ để hỗ trợ công tác giám sát và xử lý vi phạm giao thông.

G. Cơ chế phát hiện đối tượng với YOLO

YOLO hoạt động theo nguyên tắc chia hình ảnh thành lưới (grid) và dự đoán các bounding box cùng với xác suất thuộc về một lớp đối tượng. Các bước chính bao gồm:

- Chia ảnh đầu vào thành các ô lưới nhỏ.

- Dự đoán các bounding box và xác suất đối tượng trong mỗi ô.
- Lọc và giữ lại những bounding box có độ tin cậy cao nhất.
- Áp dụng thuật toán NMS (Non-Maximum Suppression) để loại bỏ các bounding box trùng lặp.

H. Suy luận (Inference) và xác định vi phạm

Trong giai đoạn suy luận, hệ thống sẽ:

- 1) **Nhận diện phương tiện:** Dùng mô hình YOLOv8 để phát hiện bounding box của các phương tiện.
- 2) **Nhận diện đèn tín hiệu:** Sử dụng cùng mô hình YOLOv8 hoặc mô hình CNN riêng (nếu cần) để xác định trạng thái đèn (đỏ, vàng, xanh).
- 3) **Phát hiện vi phạm:** Nếu đèn đỏ và bounding box của phương tiện vượt qua vạch dừng (được xác định sẵn trên ảnh/video), hệ thống đánh dấu đó là hành vi vi phạm và ghi nhận kết quả (lưu hình ảnh hoặc video bằng OpenCV).

IV. Kết quả thực nghiệm

A. Cấu hình thí nghiệm

Hệ thống được triển khai trên máy tính có GPU NVIDIA RTX 3060, sử dụng:

- Ngôn ngữ: Python 3.9
- Thư viện: PyTorch cho YOLOv8, OpenCV cho xử lý ảnh
- Tập dữ liệu: 5000 khung hình được thu thập từ video giao thông thực tế tại Hà Nội

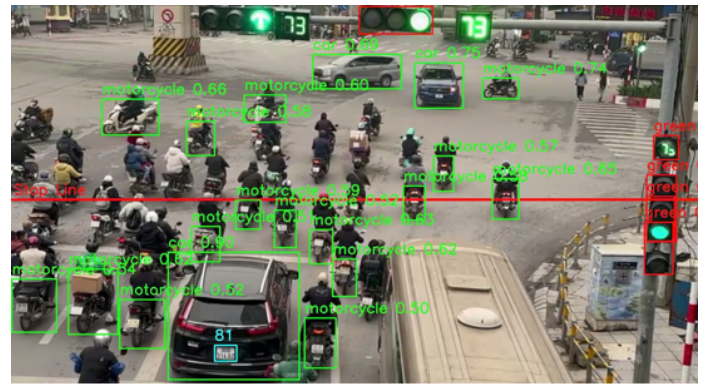
B. Đánh giá hiệu suất



Hình 2. Kết quả 1

Kết quả thực nghiệm cho thấy:

- **mAP@0.5** (nhận diện phương tiện): 94.5%
- **Độ chính xác nhận diện trạng thái đèn đỏ:** 97.0%
- **Độ chính xác phát hiện vi phạm:** 91.2%
- **Tốc độ xử lý:** 25 FPS trên GPU



Hình 3. Kết quả 2

C. Phân tích kết quả

Hệ thống cho thấy khả năng nhận diện và phát hiện vi phạm ổn định trong các điều kiện giao thông khác nhau. Một số hạn chế được nhận diện bao gồm:

- Khó khăn trong việc phát hiện chính xác khi có nhiều đối tượng chồng lấn.
- Hiệu suất giảm trong điều kiện ánh sáng quá mạnh hoặc quá tối.

Những hạn chế này sẽ được cải thiện qua việc tối ưu mô hình (tăng dữ liệu, tinh chỉnh siêu tham số) và mở rộng tập dữ liệu trong tương lai.

V. KẾT LUẬN

A. Tổng kết kết quả đạt được

Trong bài báo cáo này, chúng tôi đã xây dựng thành công hệ thống nhận diện phương tiện vượt đèn đỏ sử dụng YOLO. Cụ thể:

- Thu thập và tiền xử lý tập dữ liệu phù hợp với bài toán.
- Huấn luyện mô hình YOLO trên tập dữ liệu với độ chính xác đạt được [t].
- Phát triển hệ thống nhận diện thời gian thực có khả năng xác định và phân loại phương tiện vi phạm.
- Đánh giá hiệu suất hệ thống và tối ưu hóa mô hình để giảm sai số.

Kết quả thử nghiệm cho thấy hệ thống hoạt động hiệu quả trong điều kiện ánh sáng và góc nhìn đa dạng, đáp ứng được yêu cầu của bài toán.

B. Những hạn chế và bài học kinh nghiệm

Mặc dù đạt được những kết quả tích cực, hệ thống vẫn tồn tại một số hạn chế:

- Độ chính xác của mô hình có thể bị ảnh hưởng khi gặp điều kiện ánh sáng kém hoặc phương tiện bị che khuất.
- Cần cải thiện khả năng xử lý các trường hợp đặc biệt như phương tiện dừng quá sát vạch hoặc bị khuất một phần.
- Thời gian xử lý có thể được tối ưu hơn để đảm bảo tốc độ nhận diện thời gian thực.

Qua quá trình thực hiện, nhóm đã rút ra nhiều bài học quan trọng về việc lựa chọn mô hình, xử lý dữ liệu, cũng như tối ưu hóa thuật toán nhằm nâng cao hiệu quả hệ thống.

C. Hướng phát triển trong tương lai

Trong tương lai, hệ thống có thể được mở rộng và cải thiện theo các hướng sau:

- **Nâng cao độ chính xác:** Tiếp tục huấn luyện mô hình trên tập dữ liệu lớn hơn và đa dạng hơn.
- **Cải thiện tốc độ xử lý:** Sử dụng phần cứng mạnh hơn hoặc áp dụng các thuật toán tối ưu hóa để tăng tốc nhận diện.
- **Tích hợp hệ thống giám sát thực tế:** Xây dựng giải pháp triển khai thực tế tại các giao lộ, kết hợp với cơ sở dữ liệu vi phạm để tự động xuất cảnh báo.
- **Mở rộng phạm vi nhận diện:** Hỗ trợ nhận diện nhiều loại phương tiện hơn, chẳng hạn như xe máy, xe đạp hoặc phương tiện giao thông công cộng.

Lời cảm ơn

Nhóm nghiên cứu xin gửi lời cảm ơn đến Trường Đại học Đại Nam và các giảng viên đã hỗ trợ, góp ý trong quá trình thực hiện đề tài này.

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/ultralytics>, 2023.
- [3] G. Bradski, “The OpenCV Library,” *Dr. Dobbs 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.