

THỰC TẬP CNTT 5: TRIỂN KHAI, PHÁT TRIỂN ỨNG DỤNG AI&IOT

ĐỀ TÀI: PHÂN VÙNG PHƯƠNG TIỆN GIAO THÔNG ĐỂ ĐIỀU KHIỂN THỜI GIAN ĐÈN TÍN HIỆU

Giảng viên hướng dẫn: Th.S Lê Trung Hiếu

Kỹ sư Nguyễn Thái Khánh

Nhóm 3: Nguyễn Duy Đạt

Nguyễn Thị Lan Anh

Lê Thành Long

Đậu Cao Minh Nhật

- 1. Giới thiệu về bài toán**
- 2. Mục tiêu**
- 3. Các nghiên cứu liên quan**
- 4. Sơ đồ hệ thống**
- 5. Các thiết bị cho đề tài**
- 6. Tạo mô hình thử nghiệm**

GIỚI THIỆU VỀ BÀI TOÁN

1. Bối cảnh vấn đề

- Giao thông đô thị ngày càng phức tạp, ùn tắc vào giờ cao điểm.
- Đèn tín hiệu truyền thống cố định thời gian không linh hoạt.

2. Bài toán đặt ra

- Làm thế nào để tối ưu hóa thời gian đèn tín hiệu theo mật độ xe?
- Cần một hệ thống thông minh có thể tự điều chỉnh thời gian đèn.

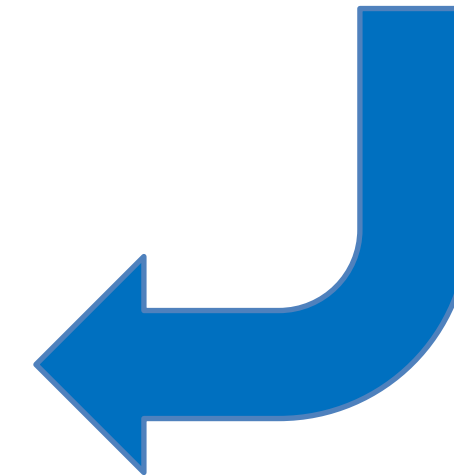
3. Giải pháp đề xuất

- Sử dụng camera (ESP 32 CAM) để đo mật độ xe theo thời gian thực.
- Ứng dụng thuật toán AI để phân tích dữ liệu và điều chỉnh thời gian của tín hiệu đèn phù hợp.



MỤC TIÊU

- Phân tích mật độ giao thông.
- Giảm Tắc Nghẽn Giao Thông..
- Tiết Kiệm Nhiên Liệu
- Tối Ưu Hóa Thời Gian Đèn Xanh.



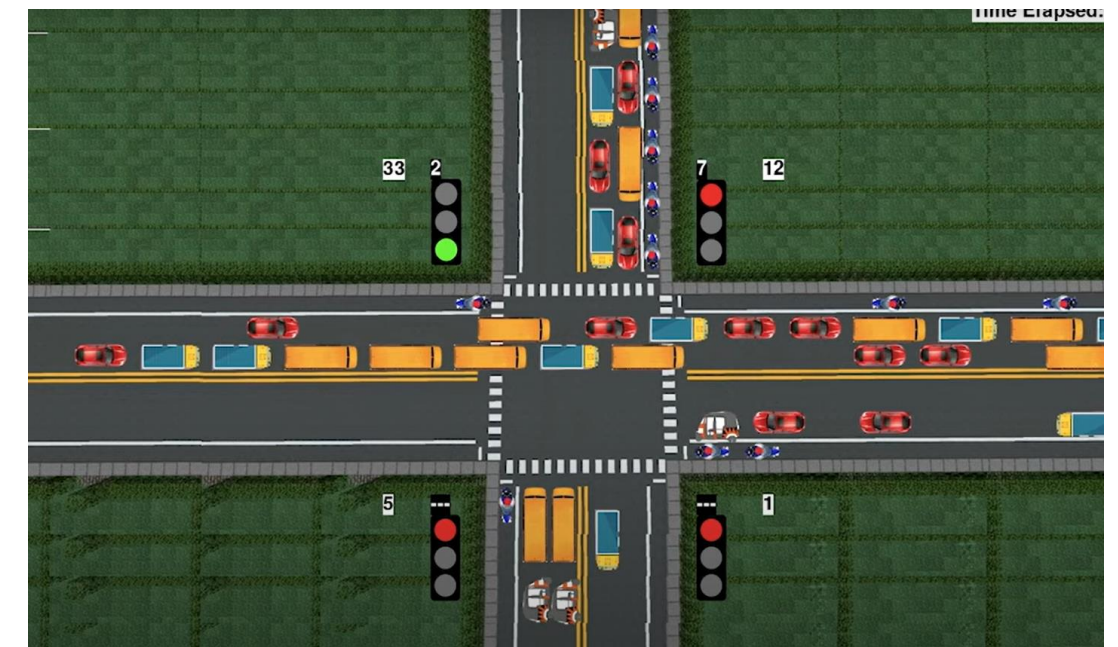
CÁC NGHIÊN CỨU LIÊN QUAN

<https://www.makerguides.com/object-detection-with-esp32-cam-and-yolo/>

- Cung cấp một hướng dẫn chi tiết về cách sử dụng ESP32-CAM kết hợp với YOLO để phát hiện và phân loại đối tượng, bao gồm cách lắp ráp phần cứng, cấu hình camera và thiết lập hệ thống nhận diện đối tượng.

<https://mecsu.vn/ho-tro-ky-thuat/phan-loai-hinh-anh-esp32cam-su-dung-machine-learning.4aG>

- Sử dụng bộ chuyển đổi USB-TTL (FTDI) để lập trình cho ESP32-CAM. Kết nối chân 5V của ESP32-CAM với VCC của FTDI, GND với GND, UOR (GPIO3) với TX và UOT (GPIO1) với RX của FTDI. Để đưa ESP32-CAM vào chế độ lập trình, nối chân GPIO0 với GND trong quá trình nạp chương trình.



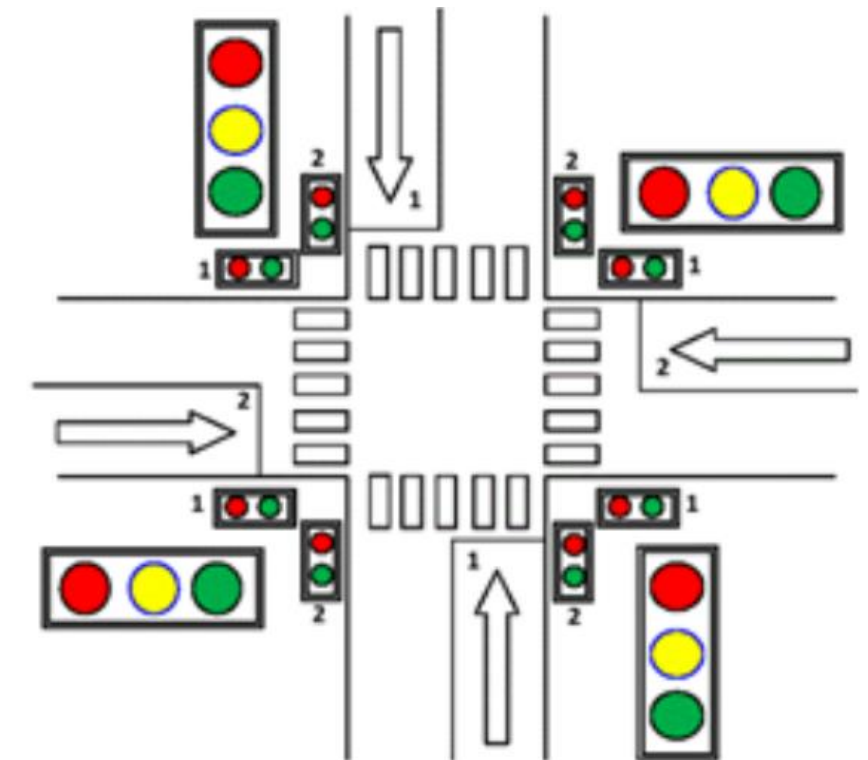
CÁC NGHIÊN CỨU LIÊN QUAN

<https://www.makerguides.com/object-detection-with-esp32-cam-and-yolo/>

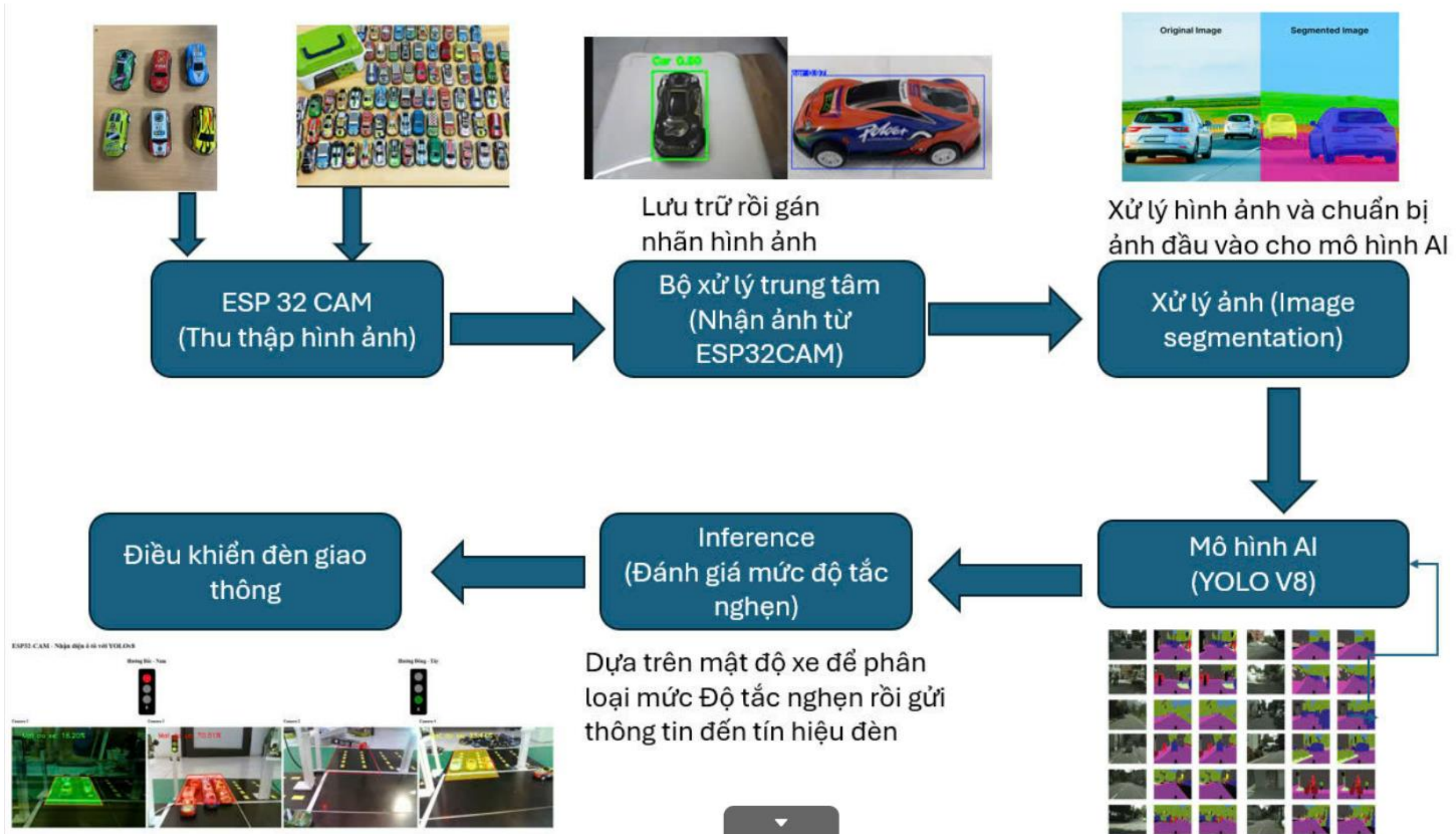
- Viết chương trình để ESP32-CAM hoạt động như một máy chủ web, cho phép truy cập và xem hình ảnh trực tiếp từ camera.

<https://tapchicongthuong.vn/nguyen-cuu-ung-dung-tri-tue-nhan-tao-de-dieu-khien-tin-hieu-den-giao-thong-tai-mot-so-nut-giao-lo-132706.htm>

- Tích hợp AI & YOLOv8 để nhận diện phương tiện.
- Phát triển thuật toán điều chỉnh đèn thích ứng dựa trên mật độ xe theo thời gian thực.
- Kết hợp dữ liệu thời gian thực từ ESP32-CAM để tối ưu điều khiển đèn.

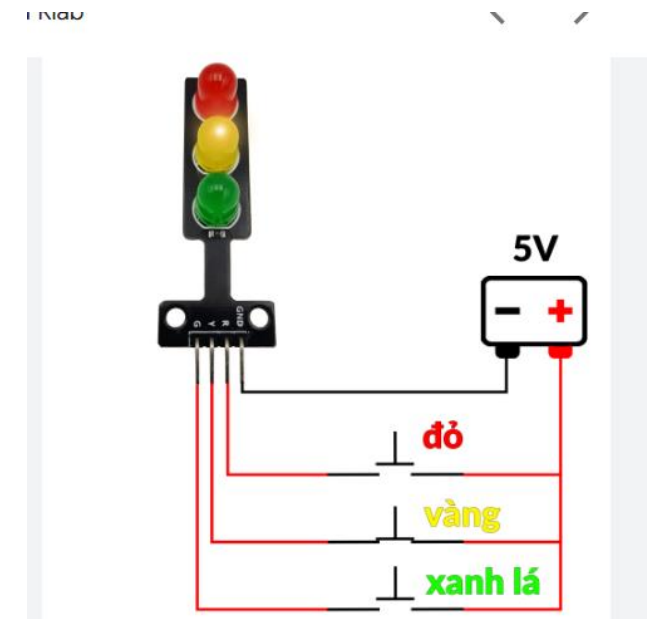
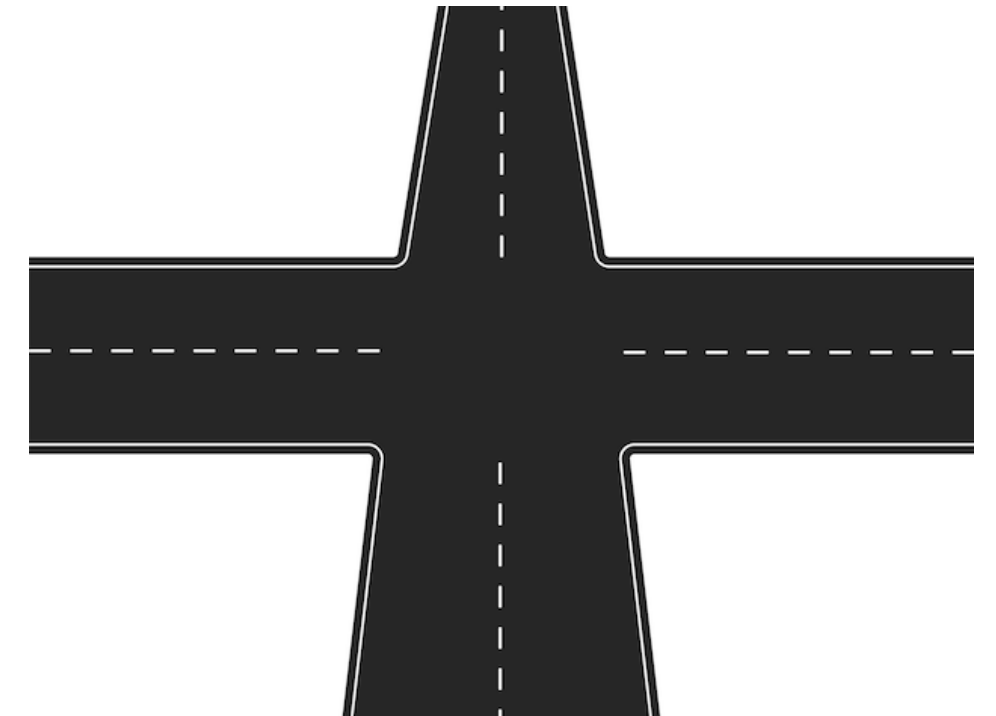


SƠ ĐỒ HỆ THỐNG



CÁC THIẾT BỊ LIÊN QUAN

- ESP 32 CAM.
- Đèn tín hiệu giao thông.
- Đường mô phỏng.
- Xe mô hình.



TIẾN HÀNH THỬ NGHIỆM MÔ HÌNH

1. Thiết lập ESP32-CAM để thu thập hình ảnh:

- Kết nối phần cứng
- Cài đặt môi trường phát triển
- Lập trình ESP32-CAM

2. Thiết lập môi trường YOLOv8 trên thiết bị xử lý cục bộ:

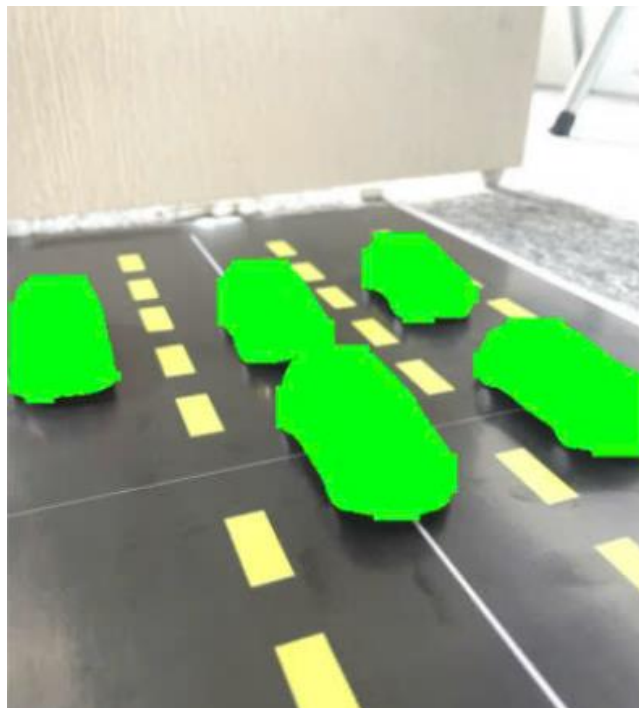
- Cài đặt Python và các thư viện cần thiết
- Tải và cấu hình mô hình YOLOv8



TIẾN HÀNH THỬ NGHIỆM MÔ HÌNH

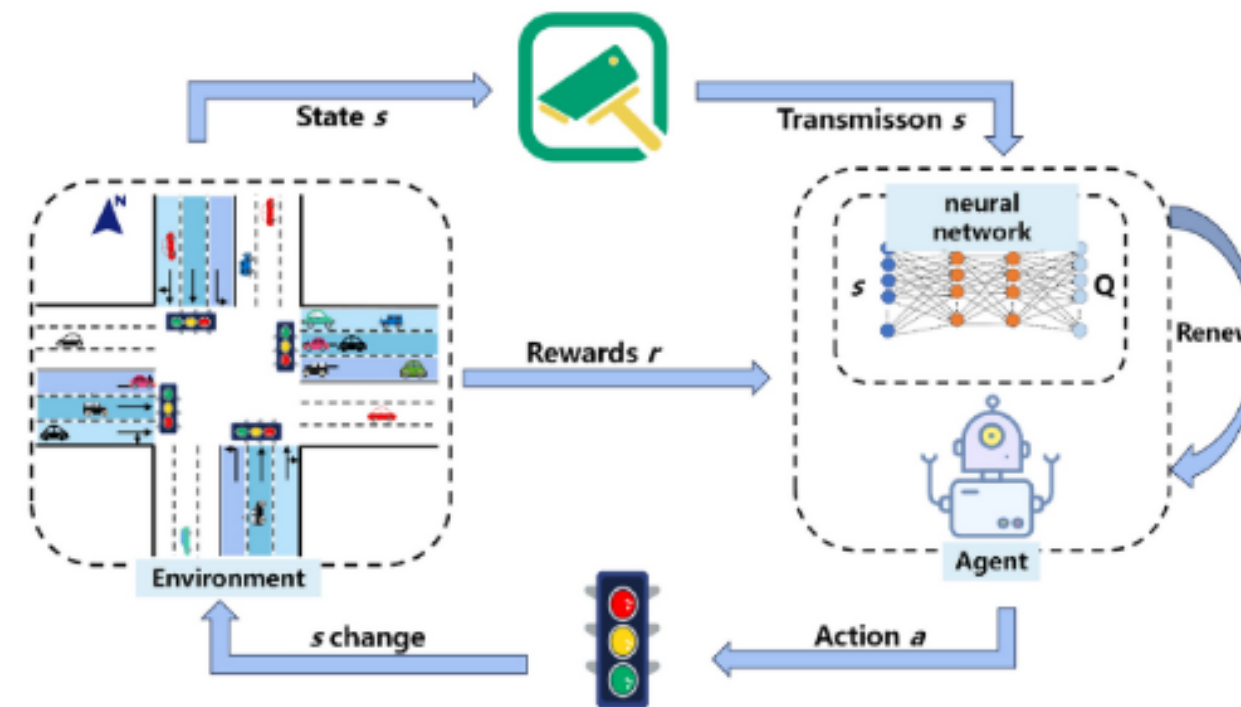
3. Dùng Image Segmentation để xử lý ảnh

- Phân loại từng pixel thành các nhóm giống nhau (VD: tất cả các xe thuộc một lớp lớn).
- Xác định từng đối tượng riêng lẻ trong ảnh (VD: phân biệt từng chiếc xe).

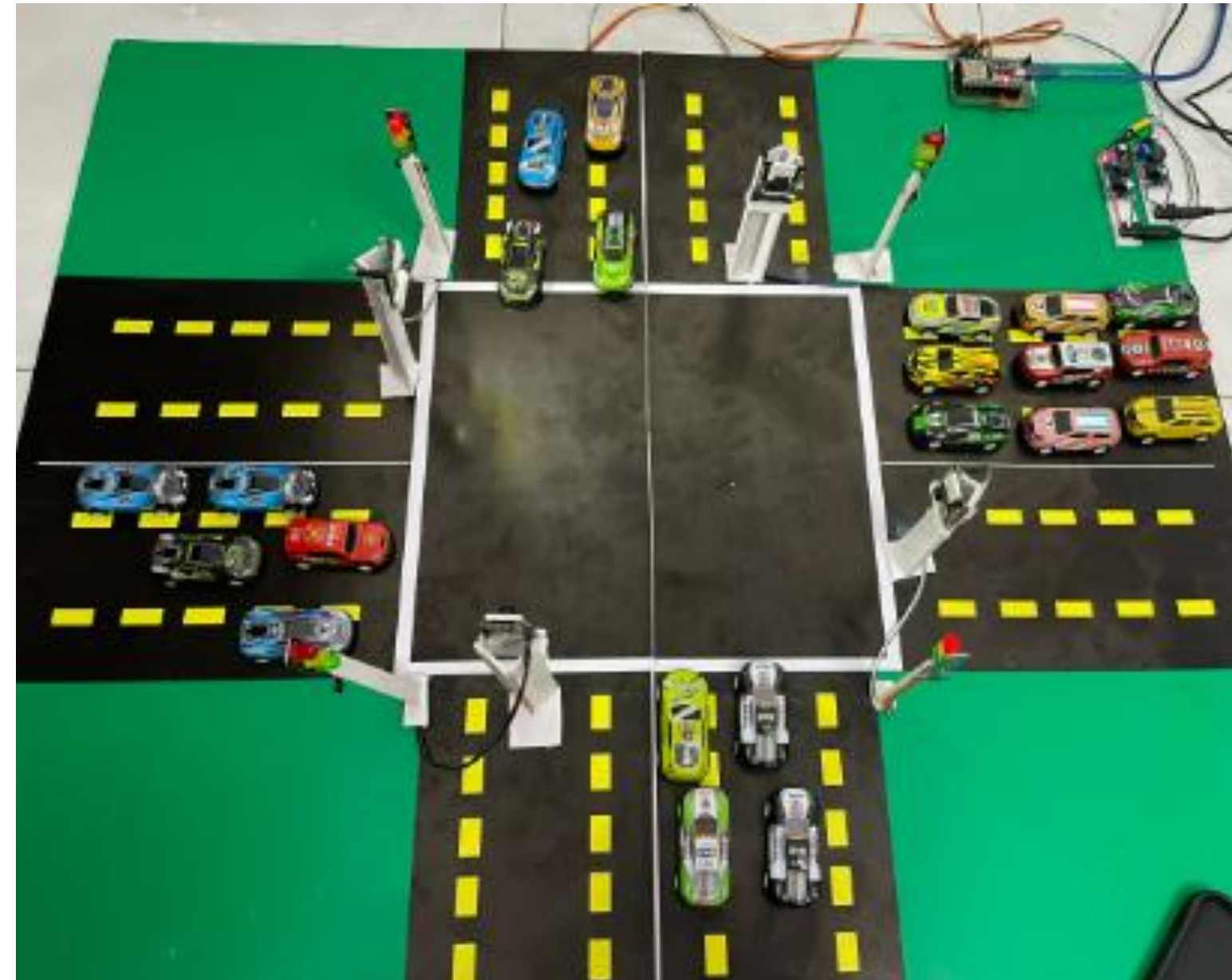
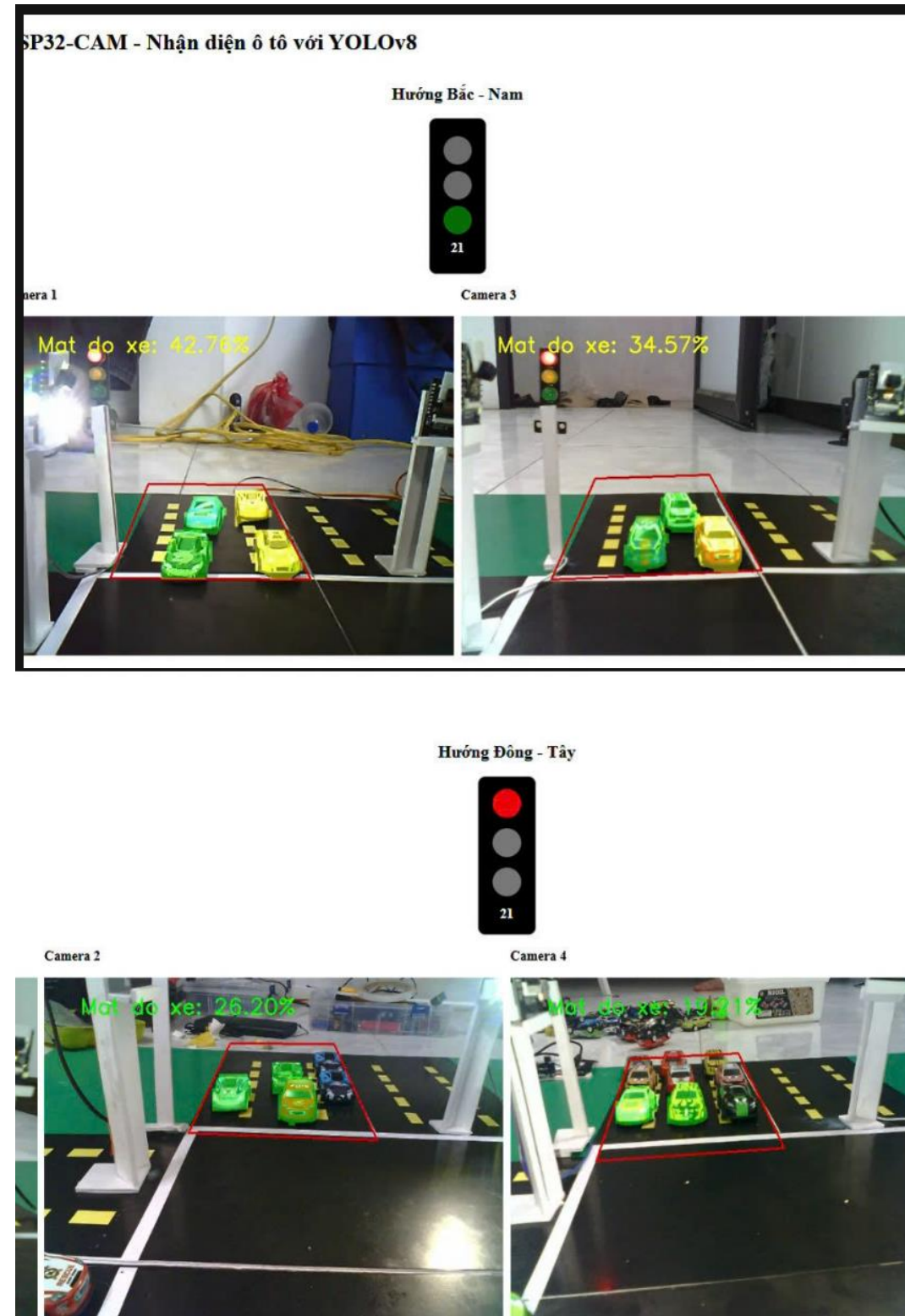


4. Adaptive Light Flow Management (ALFM) – Hệ thống quản lý đèn giao thông thông minh

Là một hệ thống điều khiển đèn tín hiệu giao thông thông minh sử dụng dữ liệu thời gian thực để điều chỉnh thời gian đèn linh hoạt, giúp tối ưu hóa luồng giao thông.



KẾT QUẢ



Điểm mạnh:

- ESP32-CAM thu thập hình ảnh giao thông thời gian thực.
- YOLOv8 giúp phát hiện phương tiện hiệu quả.
- Hỗ trợ mô hình AI hoạt động chính xác hơn.
- Thay đổi thời gian đèn theo lưu lượng thực tế.
- Tăng hiệu quả lưu thông, giảm ùn tắc.

Thách thức và cải tiến:

- ESP32-CAM có tài nguyên hạn chế, cần tối ưu truyền dữ liệu hoặc kết hợp thiết bị mạnh hơn như Raspberry Pi.
- Nên kiểm tra tốc độ phản hồi để đảm bảo hệ thống hoạt động kịp thời.



Thank You