

XÁC ĐỊNH PHƯƠNG TIỆN VI PHẠM TỐC ĐỘ

Sinh viên thực hiện:
Nguyễn Thị Duyên

1 Đặt vấn đề

2 Hướng tiếp cận

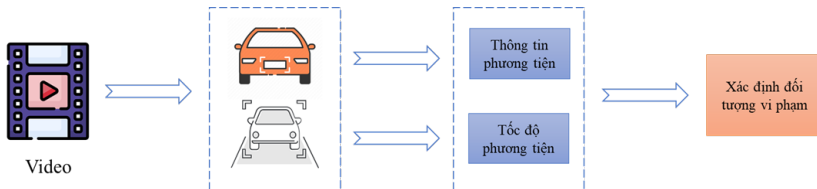
3 Kết luận

- Vi phạm tốc độ là nguyên nhân chủ yếu gây tai nạn giao thông.
- Tốc độ của phương tiện còn ảnh hưởng tới nhiều vấn đề: tắc nghẽn, ô nhiễm môi trường, ...
- Các khu vực lân cận của camera giám sát, có thể giảm vi phạm và tai nạn 25% - 35%.

→ Nhiệm vụ giám sát vận tốc các phương tiện ngày càng cấp thiết.

Bài toán cần giải quyết

- Phát hiện và theo dõi xe.
- Ước lượng tốc độ phương tiện qua ước lượng khoảng cách và thời gian.
- Xác định thông tin phương tiện qua trích xuất biển số xe.



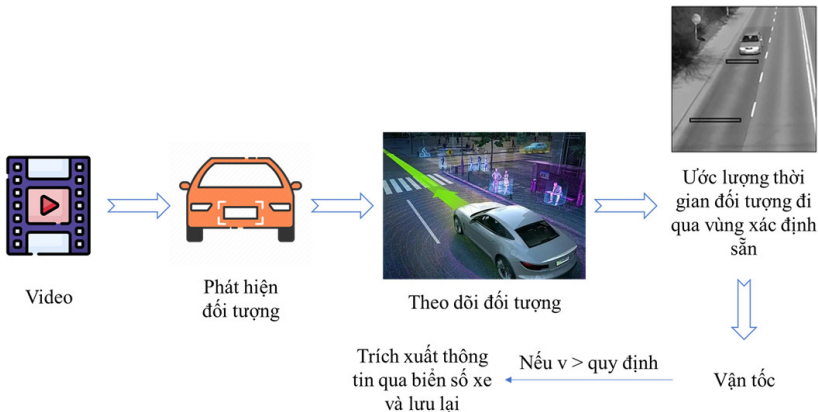
Hình 1: Nguyên lý hoạt động của hệ thống xác định vi phạm tốc độ

- Dữ liệu video được tìm kiếm qua các bài báo.
- Video lấy từ web Pexels

1 Đặt vấn đề

2 Hướng tiếp cận

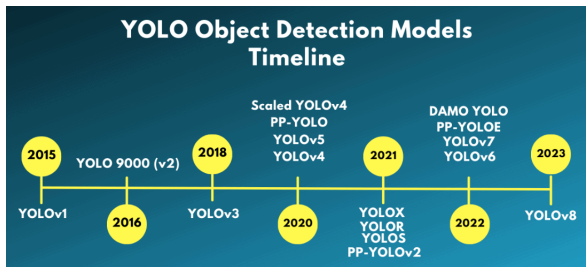
3 Kết luận



Hình 2: Hướng tiếp cận của nhóm

Phát hiện phương tiện

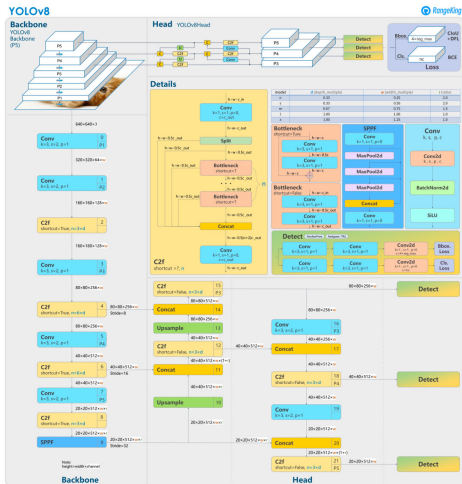
- Phát hiện phương tiện nói riêng hay phát hiện đối tượng nói chung, nhằm xác định các đối tượng và vị trí của chúng trong một hình ảnh.
- Có nhiều cách cổ điển để phát hiện đối tượng như: Background subtract, Cascade Classifier, ... nhưng chưa mang lại kết quả tốt.
- Nhiều ứng dụng của deep learning bắt đầu được ứng dụng vào năm 2012 mang lại độ chính xác và tốc độ cao.
- Dân đầu là YOLO.



Hình 3: Quá trình phát triển của YOLO

Yolo là một mô hình mạng CNN cho việc phát hiện, nhận dạng, phân loại đối tượng. Yolo được tạo ra từ việc kết hợp giữa các convolutional layers và connected layers.

Phát hiện phương tiện



Hình 4: Cấu trúc mạng YOLOv8

Phát hiện phương tiện



Hình 5: Một số hình ảnh Object Detection bằng YOLOv8

- Hiểu một cách đơn giản, theo dõi đối tượng là bài toán nâng cao của phát hiện đối tượng, khi tác vụ xử lý không còn trên hình ảnh mà là trên video.
- Có nhiều cách để phân loại các bài toán theo dõi phương tiện, như dựa trên cách tiếp cận có: Single Object Tracking hoặc Multiple Object Tracking, dựa trên phương pháp giải Online Tracking hoặc Offline Tracking, hay phân chia thành 2 dạng: Detection based Tracking hoặc Detection Free Tracking.

Bài toán thuộc phân loại: Multiple Object Tracking, Online Tracking và Detection based Tracking:

- Theo dõi tất cả các đối tượng xuất hiện trong video.
- Chỉ sử dụng 2 frame: hiện tại và ngay trước đó để tracking.
- Tập trung vào mối liên kết chặt chẽ giữa object detection và object tracking, từ đó dựa vào các kết quả của detection để theo dõi đối tượng qua các frame.

Theo dõi phương tiện

Với những lý thuyết đó, nhóm xây dựng thuật toán tracking cơ bản, với ý tưởng sử dụng khoảng cách tâm đối tượng trong 2 frame liên tiếp:

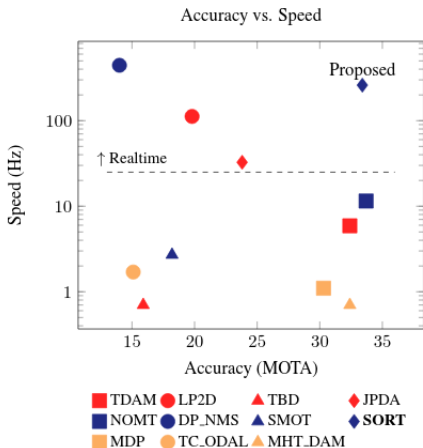
```
distance = math.hypot(center2[0] - center1[0], center2[1] - center1[1])

if distance < 10:
    tracking_objects[track_id] = pt
    track_id += 1
```

Kết quả: chưa mang lại quá tốt, khi một xe nhiều khi vẫn bị nhận nhiều ID, trong cả trường hợp không bị che lấp, hay thậm chí là xe đang đứng yên cũng bị thay đổi ID liên tục.

Theo dõi phương tiện

Để cải tiến hiệu suất của theo dõi phương tiện, nhóm tìm hiểu và triển khai xây dựng thuật toán SORT (Simple Online Realtime Object Tracking):

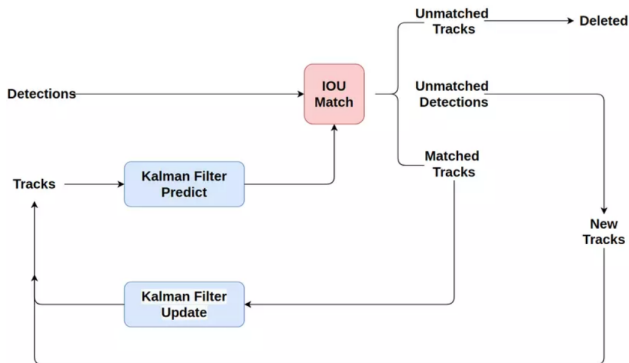


Ý tưởng chính của thuật toán SORT: Tách video thành các bài toán Object Detection riêng biệt, sau đó tìm cách liên kết các bounding box thu được ở mỗi frame, gán ID cho từng đối tượng. Với các bước thực hiện

- Detect: phát hiện vị trí các đối tượng trong frame.
- Predict: Dự đoán vị trí mới của các đối tượng dựa vào các frame trước đó.
- Associate: Liên kết các vị trí detected với các vị trí dự đoán được để gán ID tương ứng.

Theo dõi phương tiện

2 thuật toán cốt lõi của SORT là Kalman Filter và giải thuật Hungary.



Hình 6: Cách hoạt động của thuật toán SORT

Giải thuật Hungary được phát triển và công bố vào năm 1995, đề xuất để giải bài toán phân công công việc (assignment problem).

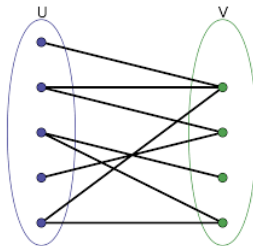
Bài toán phân công công việc

Phát biểu bài toán phân công: Có n người ($i = 1, 2, \dots, n$) và n công việc ($j = 1, 2, \dots, n$). Để giao cho người i thực hiện một công việc j cần một chi phí c . Bài toán đặt ra là cần giao cho người nào làm việc gì (mỗi người chỉ làm một việc và mỗi việc chỉ do một người làm) sao cho chi phí tổng cộng là nhỏ nhất.

Theo dõi phương tiện

Liên hệ đến bài toán theo dõi đối tượng

Có n detection ($i = 1, 2, \dots, n$) và n track predicted ($j = 1, 2, \dots, n$). Để liên kết một detection i với một track j giả sử dựa vào 1 độ đo D - D là khoảng cách giữa i và j trong không gian vector. Bài toán đặt ra là cần liên kết mỗi detection với mỗi track tương ứng sao cho sai số của việc liên kết là nhỏ nhất.



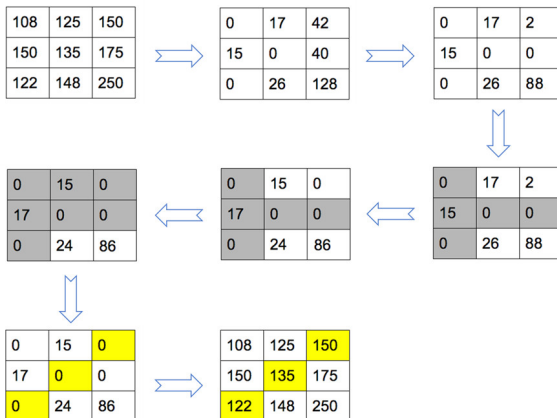
Thuật toán Hungary hoạt động dựa trên ý tưởng chính: nếu một số được cộng hoặc trừ khỏi tất cả các mục của một hàng hoặc một cột bất kỳ của ma trận chi phí, thì phép gán tối ưu cho Ma trận chi phí thu được cũng là phép gán tối ưu cho Ma trận chi phí ban đầu.

Giải thuật Hungary

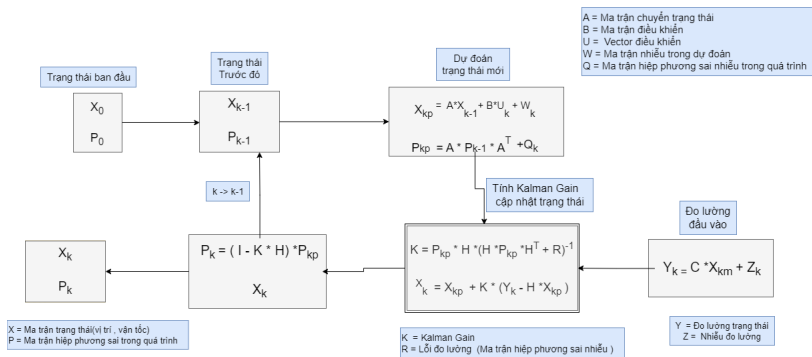
- 1. Đối với mỗi hàng của ma trận, giảm giá trị các phần tử trong hàng 1 phần bằng giá trị nhỏ nhất của phần tử trong hàng.
- 2. Thực hiện tương tự như bước 1 nhưng đối với cột.
- 3. Che tất cả các số 0 trong ma trận bằng cách vẽ các hàng ngang, dọc sao cho số lượng hàng là tối thiểu.
- 4. Nếu có đủ n hàng vẽ (ma trận chi phí có kích thước $n \times n$) thì dừng thuật toán, ngược lại, chuyển đến bước tiếp theo.
- 5. Tìm phần tử nhỏ nhất không được che bởi đường thẳng vừa kẻ nào. Trừ các phần tử còn lại cho phần tử này, sau đó quay lại bước 3.

Theo dõi phương tiện

Ví dụ:



Theo dõi phương tiện




Hình 7: Thuật toán Kalma Filter

Kalman Filter hoạt động theo các bước lặp sau:

- ➊ Dự đoán (Prediction): Dựa trên ước lượng trước đó của trạng thái và mô hình dự đoán, tính toán ước lượng dự đoán của trạng thái tiếp theo và ma trận không chắc chắn tương ứng.
- ➋ Cập nhật (Update): Nhận dữ liệu đo được mới nhất, so sánh với ước lượng dự đoán và tính toán trọng số cập nhật (gain) dựa trên ma trận không chắc chắn dự đoán và ma trận không chắc chắn đo được. Dữ liệu đo được mới nhất được sử dụng để cập nhật ước lượng trạng thái và ma trận không chắc chắn tương ứng.
- ➌ Quay lại bước 1 và tiếp tục lặp lại quá trình trên với dữ liệu đo được tiếp theo.

Độ đo khoảng cách IOU

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


Hình 8: Công thức tính IOU

Ước lượng vận tốc

Để ước lượng được vận tốc của phương tiện, cần xác định rõ thời gian t mà phương tiện cần để đi được khoảng cách S đã xác định trước đó.



Hình 9: Ước lượng vận tốc bằng cách xác định trước đoạn đường

Xác định khoảng cách

- Sử dụng 2 đường thẳng xác định quỹ đường.
- Đặc mặt định khoảng cách, dựa trên các tiêu chuẩn về đường ngăn chia làn đường.

Ước lượng thời gian

- Xác định fps của video: $\text{fps} = \text{cap.get}(\text{cv2.CAP_PROP_FPS})$.
- Tính số lượng frame mà phương tiện bắt đầu chạm vào đường line 1 và dừng khi xe đi qua đường line 2.
- Tính thời gian: $\text{số lượng frame} * 1/\text{fps}$.

Phương pháp

- Phát hiện và khoảng vùng biển số xe
- Cải thiện chất lượng ảnh biển số xe
- Đưa về ảnh nhị phân
- Đọc ký tự trong biển số xe

Phát hiện và khoảng vùng biển số xe



Video



Phát hiện
đối tượng
(YOLOv8)



Class_ids: phân
loại phương tiện

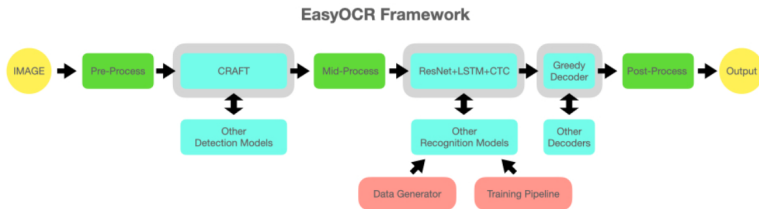
Bounding Box:
vị trí của phương
tiện trong frame
hiện tại



Xác định vị
trí biển số xe

Sử dụng model predtrain của YOLO đối với tập dữ liệu biển số xe để xác định xe có biển số không và vị trí các tọa độ của biển số xe.

Đọc ký tự trong biển số xe



Các bước của EasyOCR

- Tiền xử lý hình ảnh: Ảnh đầu vào được tiền xử lý để cải thiện chất lượng và tách riêng văn bản với nền ảnh
- Xác định vùng chứa văn bản (Text Detection): EasyOCR sử dụng một mạng nơ-ron tích chập để phát hiện vùng chứa văn bản trong hình ảnh
- Nhận dạng ký tự (Text Recognition): Sau khi xác định được vùng chứa văn bản, EasyOCR sử dụng một mô hình Transformer để nhận dạng và trích xuất văn bản từ các vùng đó.
- Cuối cùng, kết quả từ các vùng chứa văn bản được kết hợp lại để tạo ra văn bản đầy đủ và chính xác từ hình ảnh ban đầu

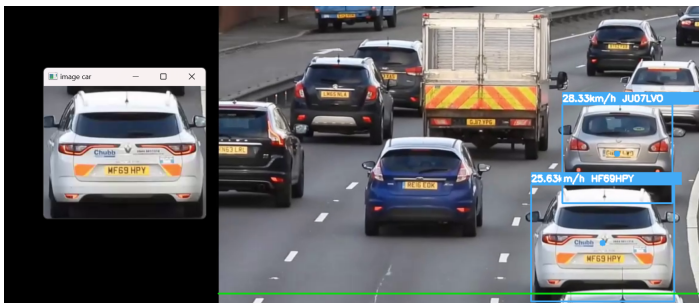
1 Đặt vấn đề

2 Hướng tiếp cận

3 Kết luận

Kết luận

Giả định các xe có vận tốc $> 25\text{km/h}$ là vi phạm tốc độ, ta có kết quả:



Sau khi tính toán vận tốc, vận tốc và thông tin xe, cùng hình ảnh chụp sẽ được hiển thị trên màn hình.

Bên cạnh đó, kết quả đồng thời trả lại một tệp txt chứa các thông tin về biển số xe, ảnh chụp lúc vi phạm, thời gian vi phạm:

```
2: (28.333332633744877, 'JU07LVO', '../output/speed_cam/2.png', '2024-01-14_05:18:27')
16: (25.63492000195965, 'HF69HPY', '../output/speed_cam/16.png', '2024-01-14_05:18:38')
38: (29.907406668952923, 'YID', '../output/speed_cam/38.png', '2024-01-14_05:19:07')
47: (25.63492000195965, 'KF17EDO', '../output/speed_cam/47.png', '2024-01-14_05:19:15')
53: (25.63492000195965, 'WE59WAS', '../output/speed_cam/53.png', '2024-01-14_05:19:26')
56: (26.916666002057635, '1551', '../output/speed_cam/56.png', '2024-01-14_05:19:28')
60: (25.63492000195965, 'MVG'NUR', '../output/speed_cam/60.png', '2024-01-14_05:19:38')
```

Một số hạn chế

- Tốc độ tính toán chậm.
- Việc tính toán vận tốc cần được khảo sát về việc nơi chỉ định 2 đường line.
- Tác vụ trích xuất thông tin từ biển số xe sử dụng OCR vẫn còn lỗi chính tả.

Lý do

Các video được đưa vào thử nghiệm chưa đáp ứng đủ về mặt kỹ thuật của một hệ thống giám sát phương tiện giao thông như: tốc độ fps, độ phân giải,... Dẫn đến việc các mô hình và code chạy trên video mờ, chất lượng kém, ...

→ Việc sử dụng các video từ hệ thống giám sát có thể mang lại cái nhìn tổng quan và kết quả tốt hơn.

Bên cạnh những hạn chế, có thể dễ dàng nhận thấy, trong thực tế, bài toán được áp dụng cho các video được ghi lại từ camera giám sát giao thông:

- Có tốc độ fps từ 60 – 120 fps, độ chi tiết của các chuyển động được ghi lại mượt mà.
- Độ phân giải thường có 1080 mega pixel, cho ra được các hình ảnh frame sắc nét.

Ngoài ra, việc vi phạm tốc độ thường xảy ra ở các khu vực rộng rãi, ít phương tiện qua lại, nên khó có thể xảy ra các trường hợp biển số xe bị che lấp bởi 1 xe khác.

→ Kết quả hiện tại có thể được đánh giá là tốt khi việc trích xuất biển số chiếm gần 50% độ chính xác đối với các xe vi phạm.

Hướng phát triển

- Sử dụng thuật toán deepSORT và thư viện có sẵn, cải thiện tốc độ tính toán.
- Cải thiện các phương pháp làm sắc nét, tăng màu sắc cho ảnh.

Thank you
for watching

- [1] Automated License Plate Detection and Speed Estimation of Vehicle using Machine Learning - Haar Classifier Algorithm
- [2] Real-time speed estimation of vehicles from uncalibrated view-independent traffic cameras
- [3] Hậu quả tai nạn giao thông
- [4] Bui Tien Dung, SORT - Deep SORT: Một góc nhìn về Object Tracking, VIBLO, December 2020, Available from
- [5] Nhận diện biển số xe tự động
- [6] Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network