

XỬ LÝ VI PHẠM VƯỢT ĐÈN ĐỎ VÀ DỪNG ĐỖ SAI DỰA TRÊN HỌC SÂU RED LIGHT AND WRONG PARKING VIOLATION DETECTION SYSTEM BASED ON DEEP LEARNING

Huỳnh Hữu Hưng, Nguyễn Văn Phú, Nguyễn Tường

Trường Đại học Bách khoa – Đại học Đà Nẵng;

hhhung@dut.udn.vn, nvphu1306@gmail.com, ntuongvn97@gmail.com

Tóm tắt - Theo xu hướng phát triển một thành phố thông minh, việc xây dựng được một hệ thống tự động phát hiện lỗi vi phạm là cần thiết nhằm giảm áp lực cho các công tác quản lý và giám sát giao thông. Trong bài báo này, nhóm tác giả đề xuất một giải pháp mới cho hệ thống phát hiện các lỗi vi phạm khác nhau như vượt đèn đỏ, dừng đỗ sai bằng cách sử dụng mạng tích chập YOLOv3 để trích xuất hình ảnh của đối tượng, sau đó thông tin về vị trí của phương tiện được xác định qua thuật toán theo dõi sẽ được dùng để phân loại các vi phạm. Giải pháp đề xuất được thử nghiệm trên tập dữ liệu video giao thông trên địa bàn thành phố Đà Nẵng và cho ra kết quả khả quan với độ chính xác trong phát hiện vi phạm lên tới 94% trong tập dữ liệu ban ngày. Với các điều kiện môi trường khác, kết quả phân loại đạt mức từ 40% đến 80%.

Từ khóa - YOLOv3; mạng tích chập; phát hiện vi phạm giao thông; học sâu

Abstract - Nowadays, smart city is a rising trend; therefore, building a system that can automatically detect violations to reduce the pressure on traffic surveillance is necessary. In this paper, the authors have proposed a novel method that can help to detect various traffic violations such as going through red lights or wrong parking by using the YOLOv3 neural network to recognize violating vehicles, then giving information about the position of the vehicles identified by tracking object which can be used to classify the traffic violations. The proposed method is evaluated on a Da Nang traffic data set and the experiment has yielded promising results with an accuracy of 94% in morning dataset. In other conditions, the results are in the range of 40% and 80%.

Key words - YOLOv3; Convolutional Neural Network; Traffic Violation Detection; Deep Learning

1. Đặt vấn đề

Với sự tăng trưởng về số lượng phương tiện, số vụ tai nạn giao thông đang tăng nhanh. Trong bối cảnh đó, để đảm bảo an toàn giao thông và giảm thiểu tai nạn, cần phải xây dựng được một hệ thống phát hiện vi phạm giao thông giúp cho việc giám sát giao thông một cách hiệu quả và giảm các trường hợp lái xe có khả năng gây nguy hiểm. Các hệ thống phát hiện vi phạm giao thông cần phải đáp ứng khả năng xử lý trong thời gian thực liên tục trong 24 giờ. Mặt khác, công cụ cần có độ chính xác cao trong khả năng nhận diện phương tiện và lỗi vi phạm ở tốc độ cao và trong những môi trường ngoại cảnh khác nhau.

Gần đây, các nghiên cứu sử dụng tín hiệu hình ảnh từ các camera giám sát đang được tập trung phát triển. Ứng dụng của thuật toán di truyền được đề cập trong [1], [2] để phát hiện lỗi vượt vạch dừng tín hiệu đèn đỏ. Xiaoling và cộng sự [3] đã sử dụng các kỹ thuật xử lý hình ảnh như tách biên và optical flow phát hiện các đối tượng vi phạm giao thông. Trong [4], một mô hình camera dựa trên các phân tích hình học được áp dụng để hỗ trợ cho việc phát hiện các đối tượng. Các mạng nơ-ron tích chập như Faster-CNN và Single Shot Multibox Detector (SSD) được sử dụng trong [5] để nhận diện các đối tượng tham gia giao thông.

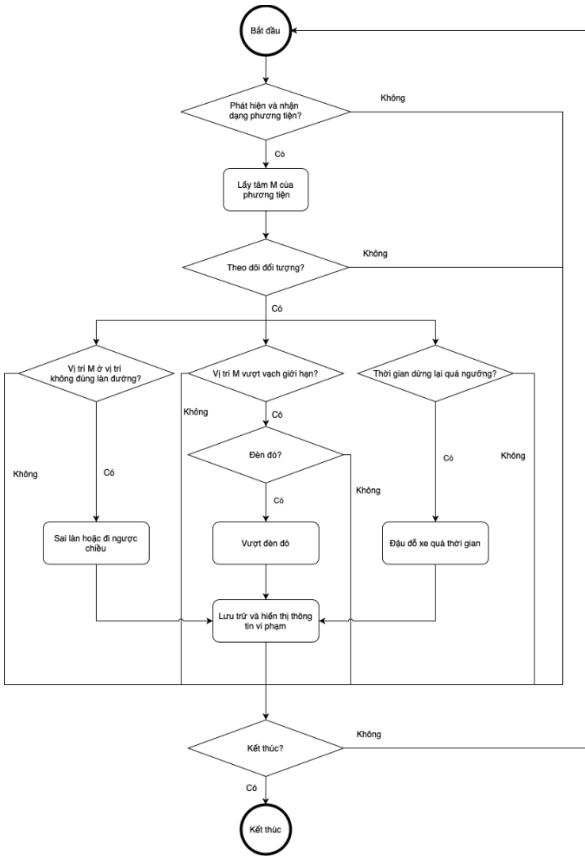
Giải pháp đề xuất sử dụng mô hình mạng YOLOv3 [6] được sử dụng để phát hiện và xác định vị trí của phương tiện đang tham gia giao thông. YOLOv3 có ưu điểm về tốc độ nhận dạng và độ chính xác hơn SSD và Faster-CNN, do đó mô hình mạng này phù hợp với yêu cầu xử lý thời gian thực của hệ thống. Các lỗi vi phạm được nghiên cứu trước đây có thể kể đến như di chuyển quá tốc độ và

vượt đèn đỏ [1], [2], trong bài báo này, giải pháp mà nhóm tác giả đưa ra có thể xử lý các vi phạm về vượt đèn đỏ và đậu đỗ sai quy định dựa trên thông tin về vị trí của phương tiện. Dữ liệu sử dụng để huấn luyện cho mô hình nhận dạng phương tiện được trích từ các video giao thông trên địa bàn thành phố Đà Nẵng [7]. Các kết quả thu được cho thấy, giải pháp đề xuất là một cách tiếp cận hứa hẹn trong việc phát triển các hệ thống phát hiện vi phạm giao thông tự động với độ chính xác cao và dễ dàng cài đặt trong thực tế. Bố cục của bài báo được trình bày như sau: tổng quan giải pháp đề xuất, cấu trúc và nguyên lý hoạt động của mạng YOLOv3, thuật toán theo dõi đối tượng, phương pháp xác định vi phạm, kết quả thử nghiệm và cuối cùng là kết luận.

2. Phương pháp đề xuất

2.1. Tổng quan

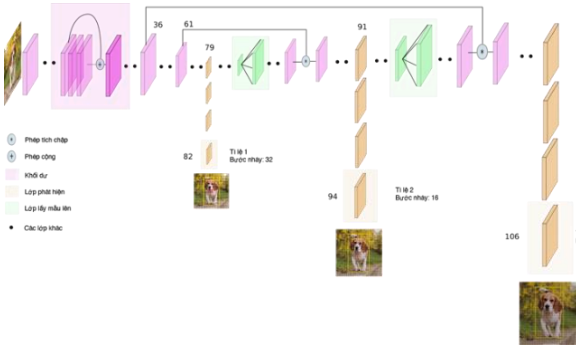
Theo Hình 1, giải pháp của nhóm tác giả có thể chia làm 2 giai đoạn: Phát hiện phương tiện và phân loại vi phạm. Giai đoạn phát hiện đối tượng sẽ thực hiện việc nhận dạng và xác định vị trí của phương tiện. Mô hình mạng YOLOv3 được huấn luyện với dữ liệu từ [7], [8] giúp xác định các khung chữ nhật giới hạn của phương tiện tham gia giao thông. Vị trí của phương tiện được xác định bởi vị trí tâm của khung giới hạn. Trong giai đoạn tiếp theo, dựa vào thông tin về vị trí của phương tiện và tham số đã biết như vị trí làn đường đúng quy định, vị trí vạch giới hạn, ngưỡng thời gian giới hạn cho việc đậu đỗ để xác định phương tiện giao thông có vi phạm hay không và phân vào nhóm. Hình ảnh vi phạm sẽ được lưu trữ để quản lý và đối chiếu nếu cần.



Hình 1. Tổng quan giải pháp đề xuất

2.2. Mô hình mạng YOLOv3

2.2.1. Cấu trúc mạng



Hình 2. Cấu trúc mạng tích chập YOLOv3 [9]

Mô hình mạng YOLOv3 là được xây dựng dựa trên cấu trúc của mạng Darknet [10] với 53 lớp. Nhưng với nhiệm vụ phát hiện, mạng được biến thể thêm 53 lớp được xếp chồng lên nhau. Từ đó mạng YOLOv3 có đến 106 lớp.

2.2.2. Hàm mất mát

Hàm mất mát sử dụng trong quá trình huấn luyện được xây dựng dựa trên hàm sai số bình phương giữa các nhãn dự đoán và nhãn đúng để tính toán độ sai lệch. Hàm mất mát trong mô hình mạng YOLOv3 bao gồm:

- Hàm mất mát khi dự đoán phân loại. Nếu một đối tượng được phát hiện, độ mất mát phân loại cho đối tượng đó là sai số bình phương xác suất giữa nhãn thật và nhãn được dự đoán:

$$\sum_{i=0}^{S^2} 1_{ij}^{obj} \sum_{c \in \text{classes}} (p_i(c) - \widehat{p_i(c)})^2 \quad (1)$$

Trong đó, 1_{ij}^{obj} 1 nếu đó là đối tượng, ngược lại 0;

$\widehat{p_i(c)}$ Biểu thị xác suất đối tượng thuộc phân lớp c ;

Ký hiệu \wedge trên đầu biểu diễn tham số được dự đoán.

- Hàm mất mát khi dự đoán vị trí đối tượng. Độ mất mát xác định vị trí được tính dựa trên lỗi sai số bình phương của kích thước và vị trí của khung dự đoán và khung được gán nhãn:

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} [(x_i - \widehat{x_i})^2 + (y_i - \widehat{y_i})^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} \left[(\sqrt{w_i} - \sqrt{\widehat{w_i}})^2 + (\sqrt{h_i} - \sqrt{\widehat{h_i}})^2 \right] \quad (2)$$

Trong đó, x_i, y_i : Tọa độ trung tâm của khung;

w_i, h_i : Chiều rộng và chiều cao của khung;

λ_{coord} : Tăng lượng mất mát cho các tọa độ của khung dự đoán.

- Hàm mất mát về độ tin cậy nếu đối tượng được phát hiện trong khung dự đoán:

$$\sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} (C_i - \widehat{C_i})^2 \quad (3)$$

$\widehat{C_i}$ là độ tự tin của khung dự đoán thứ j trong ô lưới thứ i .

1_{ij}^{obj} 1 nếu khung dự đoán thứ j trong ô lưới i chịu trách nhiệm phát hiện đối tượng, ngược lại 0.

- Hàm mất mát về độ tin cậy nếu đối tượng không được phát hiện trong khung dự đoán.

$$\lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{noobj} (C_i - \widehat{C_i})^2 \quad (4)$$

λ_{noobj} Giảm trọng lượng độ mất mát khi phát hiện ảnh nền.

- Hàm mất mát cuối là tổng các hàm mất mát dự đoán vị trí, độ tin cậy, phân loại cùng nhau ta được:

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} [(x_i - \widehat{x_i})^2 + (y_i - \widehat{y_i})^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} \left[(\sqrt{w_i} - \sqrt{\widehat{w_i}})^2 + (\sqrt{h_i} - \sqrt{\widehat{h_i}})^2 \right] + \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} (C_i - \widehat{C_i})^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{noobj} (C_i - \widehat{C_i})^2 + \sum_{i=0}^{S^2} 1_{ij}^{obj} \sum_{c \in \text{classes}} (p_i(c) - \widehat{p_i(c)})^2 \quad (5)$$

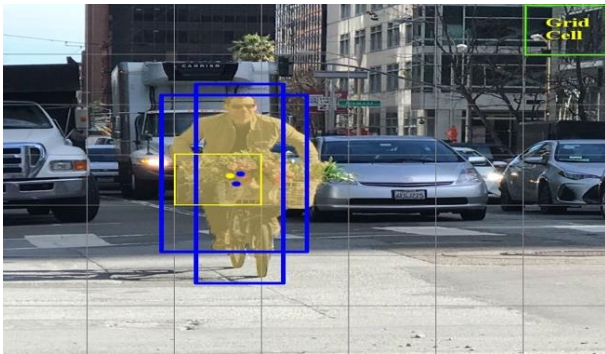
2.3. Mô tả cách hoạt động của mạng YOLOv3

Mạng YOLOv3 chia hình ảnh đầu vào thành $S \times S$ ô lưới. Mỗi ô mô lưới chỉ dự đoán một đối tượng. Ví dụ, ta xét hình ảnh bên dưới.



Hình 3. Mạng lưới $S \times S$ [11]

Ô lưới màu vàng dự đoán đối tượng nơi mà trung tâm của đối tượng đó đặt tại ô lưới. Mỗi ô lưới dự đoán một số khung giới hạn nhất định. Trong ví dụ, ô lưới màu vàng sử dụng 2 khung giới hạn để xác định đâu là vị trí của đối tượng.



Hình 4. Mỗi ô lưới dự đoán 2 khung giới hạn [11]

Mỗi ô lưới dự đoán B khung giới hạn và mỗi khung có một độ tự tin. Mỗi ô cũng dự đoán xác suất của C lớp tại mỗi khung giới hạn.

Mỗi khung giới hạn có 5 phần tử: (x, y, w, h) là vị trí của khung và một độ tự tin của khung. Độ tự tin của khung phản ánh khả năng khung đó có chứa đối tượng và độ chính xác như thế nào. Mỗi ô lưới dự đoán xác suất của C lớp và xác suất của lớp là xác suất rằng đối tượng được phát hiện thuộc về một lớp cụ thể.

2.4. Thuật toán theo dõi đối tượng.

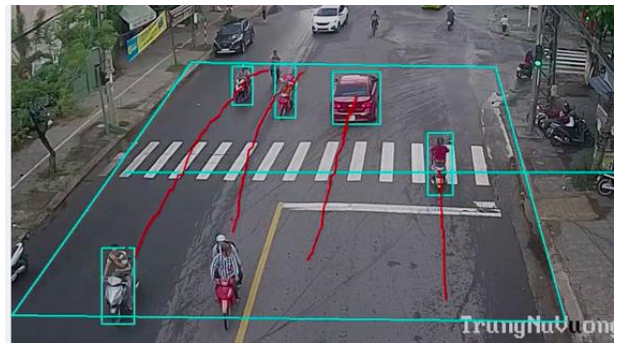
Để phát triển thuật toán theo dõi đối tượng giữa các khung hình, nhóm tác giả sử dụng mô hình đã được huấn luyện để phát hiện và nhận dạng của từng đối tượng có trong mỗi khung hình. Một tệp nhật ký được sử dụng để thu thập tất cả các tọa độ trọng tâm cho mỗi đối tượng. Thông tin từ tệp nhật ký có thể giúp vẽ và kiểm tra đường đi của mỗi đối tượng.

Danh sách tọa độ trung tâm được lưu trong tệp nhật ký:

```
traces = {
  'A': [(112, 36), (112, 45), (112, 52), (112, 54), (112, 63), (111, 73), (111, 86),
  'B': [(119, 37), (120, 42), (121, 54), (121, 55), (123, 64), (124, 74), (125, 87),
  'C': [(93, 23), (91, 27), (89, 31), (87, 36), (85, 42), (82, 49), (79, 59), (74,
  'D': [(118, 30), (124, 83), (125, 90), (116, 101), (122, 100)]
  'E': [(77, 27), (75, 30), (73, 33), (70, 37), (67, 42), (63, 47), (59, 53), (55,
  'F': [(119, 30), (120, 34), (120, 39), (122, 59), (123, 60), (124, 70), (125, 82),
  'G': [(88, 37), (87, 41), (85, 48), (82, 55), (79, 63), (76, 74), (72, 87), (67,
  'H': [(124, 35), (123, 40), (125, 45), (127, 59), (126, 59), (128, 67), (130, 78),
  'I': [(98, 26), (97, 30), (96, 34), (94, 40), (92, 47), (90, 55), (87, 64), (84,
  'J': [(123, 60), (125, 63), (125, 81), (127, 93), (126, 98), (125, 100)]
}
```

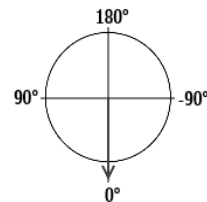
Hình 5. Danh sách tọa độ trung tâm

Các dấu vết xe cá nhân được vẽ trên hình ảnh như sau:



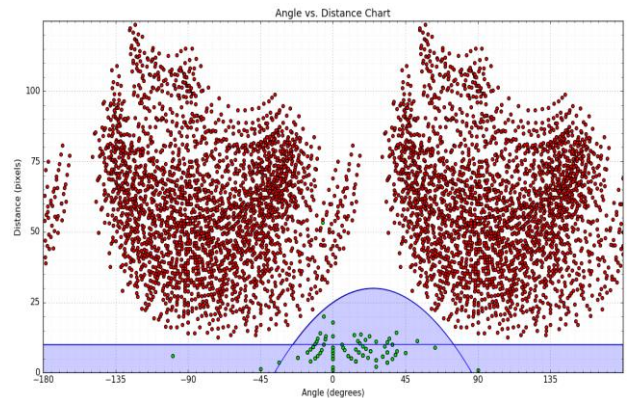
Hình 6. Vết di chuyển của phương tiện

Để phân tích chuyển động, hướng di chuyển của các đối tượng được thể hiện bởi các vector, mỗi vector là độ chuyển dời của tâm đối tượng. Sơ đồ ở Hình 7 quy định hệ quy chiếu được sử dụng trong báo cáo.



Hình 7. Sơ đồ phân tích góc

Để xác định các chuyển động hợp lệ, nhóm tác giả sử dụng biểu đồ phân tán góc so với khoảng cách theo Hình 8:



Hình 8. Biểu đồ phân tán góc so với khoảng cách

- Các điểm xanh đại diện cho chuyển động hợp lệ mà nhóm tác giả đã xác định bằng cách sử dụng danh sách các điểm cho mỗi đối tượng.

- Điểm đỏ thể hiện chuyển động không hợp lệ - vector giữa các điểm trong các làn đường giao thông liền kề.

- Hai đường cong giới hạn được định nghĩa theo (6) và (7) sử dụng để loại bỏ các vector không hợp lệ. Bất kỳ điểm nào nằm dưới một trong hai đường cong có thể được coi là hợp lệ. Các điểm hợp lệ thể hiện các chuyển động về phía camera do đó góc di chuyển phần lớn sẽ từ -45° đến 45° , các chuyển động ngang nhỏ có thể có trong quá trình di chuyển cũng được giữ lại. Tùy thuộc vào vị trí của camera mà tọa độ đỉnh parabol (6) và các tham số khác được thiết lập.

$$distance = -0.008(angle - 25)^2 + 30 \quad (6)$$

$$distance = 10 \quad (7)$$

Để thực thi thuật toán, phương tiện V cần có các thông tin sau:

- Định danh;
- Danh sách các vị trí, vị trí phát hiện gần nhất ở đầu danh sách;
- Bộ đếm nhìn thấy lần cuối - số lượng khung hình kể từ lần cuối chúng ta nhìn thấy chiếc xe này;
- Thuộc tính đánh dấu để xem chiếc xe đã được tính hay chưa.

Một danh sách L được sử dụng để lưu trữ các phương tiện đang được theo dõi. Trên mỗi khung hình, nhóm tác giả sẽ sử dụng danh sách các khung giới hạn và vị trí của các phương tiện để cập nhật L.

Thuật toán: Theo dõi đối tượng tham gia giao thông

Đầu vào: Danh sách phương tiện L, danh sách vị trí các đối tượng được phát hiện P

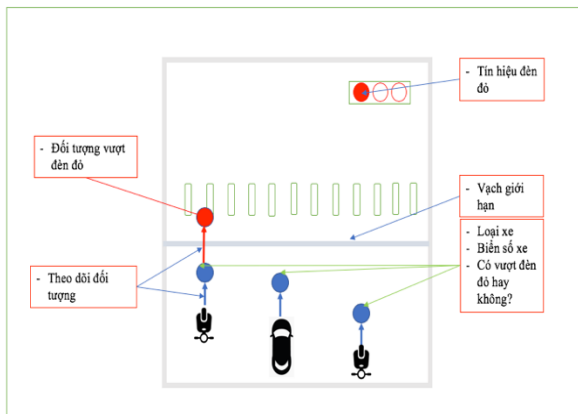
Đầu ra: Danh sách phương tiện L đã được cập nhật

1: Đối với mỗi phương tiện trong L:

- 1.1: Nếu vị trí hợp lệ, cập nhật vị trí xe và đặt lại bộ đếm nhìn thấy lần cuối.
- 1.2: Nếu không, tăng bộ đếm nhìn thấy lần cuối cho phương tiện.
- 2: Tạo phương tiện V mới đối với các vị trí còn lại và thêm vào L.
- 3: Với mỗi phương tiện trong L:
- 3.1: Nếu phương tiện chưa được đếm, cập nhật tổng số và đánh dấu phương tiện.
- 3.2: Nếu bộ đếm nhìn thấy lần cuối vượt quá ngưỡng, loại bỏ phương tiện khỏi L.

2.5. Hệ thống phát hiện vi phạm

2.5.1. Phát hiện vi phạm vượt đèn đỏ



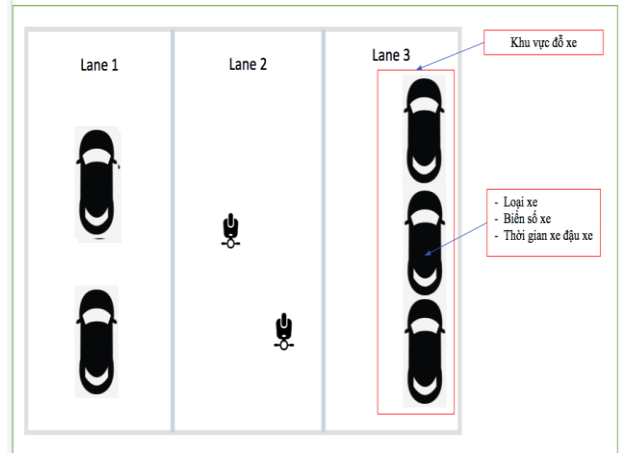
Hình 9. Minh họa bắt lỗi vi phạm vượt đèn đỏ

Hình 9 mô tả các thông số ban đầu và quá trình phát hiện vi phạm của phương tiện. Các thông số ban đầu cần được xác định là tín hiệu đèn đỏ và vị trí của vạch giới hạn. Trong bài toán này sử dụng mạng YOLOv3 (đã được mô tả ở phần trước) để phát hiện và nhận dạng phương tiện. Phương tiện được nhận dạng được trích xuất thông tin về vị trí để quyết định phương tiện có vượt đèn đỏ hay không. Trong thời gian đèn tín hiệu đèn giao thông chuyển sang màu đỏ, nếu phát hiện có đối tượng vượt qua vạch giới hạn thì lưu lại thông tin đối tượng vào cơ sở dữ liệu.

2.5.2. Phát hiện vi phạm đậu đỗ sai

Tương tự phát hiện vi phạm vượt đèn đỏ, thông số đầu vào trong trường hợp phát hiện vi phạm đậu đỗ sai là vị trí khu vực cấm đậu đỗ. Sau khi phát hiện được phương tiện tham gia giao thông, tiến hành theo dõi đối tượng. Kiểm tra

thời gian đỗ xe của đối tượng tại khu vực cấm, nếu vượt quá thời gian quy định thì thông báo và lưu thông tin vào cơ sở dữ liệu.

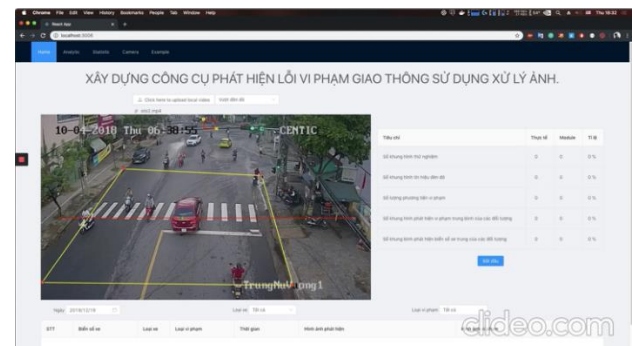


Hình 10. Minh họa phát hiện vi phạm đậu đỗ sai

3. Kết quả thực nghiệm

Dữ liệu huấn luyện cho mô hình nhận dạng phương tiện được lấy camera giám sát giao thông thông minh của thành phố Đà Nẵng và tập dữ liệu COCO [7], [8] bao gồm 30000 hình ảnh phương tiện giao thông. Mặt khác, để đánh giá khả năng phân loại vi phạm của giải pháp, nhóm tác giả đã sử dụng tổng cộng 306 video vi phạm giao thông trong các điều kiện ngoại cảnh khác nhau bao gồm điều kiện ánh sáng ban ngày, trời mưa và khi trời tối được lấy từ [7], trong đó có 195 video phương tiện vượt đèn đỏ và còn lại là các vi phạm về đậu đỗ sai quy định.

Giải pháp đề xuất được cài đặt bằng ngôn ngữ Python và các nền tảng Flask, ReactJS để cài đặt mô hình thử nghiệm cho giải pháp. Máy tính chạy mô hình được trang bị card đồ họa Nvidia GTX 1070Ti.



Hình 11. Mô hình thử nghiệm

Kết quả phát hiện vi phạm vượt đèn đỏ:

Bảng 1. Kết quả thử nghiệm ban ngày

	Thực tế	Module	Độ chính xác
Số lượng vi phạm (xe máy)	50	44	88 %
Số khung hình vi phạm trung bình (xe máy)	18	16	88,8 %
Số lượng vi phạm (ô tô)	50	46	92 %
Số khung hình vi phạm trung bình (ô tô)	18	14	77,7 %

Bảng 2. Kết quả thử nghiệm trời mưa

	Thực tế	Module	Độ chính xác
Số lượng vi phạm (xe máy)	40	34	85%
Số khung hình vi phạm trung bình (xe máy)	18	11	61,1 %
Số lượng vi phạm (ô tô)	40	35	87,5 %
Số khung hình vi phạm trung bình (ô tô)	18	12	66,7 %

Bảng 3. Kết quả thử nghiệm trời tối

	Thực tế	Module	Độ chính xác
Số lượng vi phạm (xe máy)	50	20	40 %
Số khung hình vi phạm trung bình (xe máy)	18	8	44,4 %
Số lượng vi phạm (ô tô)	50	29	58 %
Số khung hình vi phạm trung bình (ô tô)	18	9	50 %

Từ các bảng kết quả cho việc phát hiện các vi phạm lỗi vượt đèn đỏ, có thể thấy, giải pháp đề xuất hoạt động hiệu quả trong môi trường đủ ánh sáng. Cụ thể, khi phương tiện giao thông di chuyển vào ban ngày, với độ chính xác trên 90% trong Bảng 1. Ở các điều kiện khác được thể hiện ở Bảng 2 và 3, độ chính xác của giải pháp giảm. Độ chính xác trên số lượng vi phạm và độ chính xác trên số khung hình xấp xỉ nhau cho thấy sự tương đồng giữa hai kết quả này. Độ chính xác trên mỗi khung hình cao thể hiện các vi phạm trong báo cáo khá nhạy với giải pháp đề xuất.

Kết quả phát hiện vi phạm dừng đỗ sai quy định:

Bảng 4. Kết quả thử nghiệm ban ngày

	Thực tế	Module	Độ chính xác
Số lượng vi phạm	67	63	94.02%

Bảng 5. Kết quả thử nghiệm trời mưa

	Thực tế	Module	Độ chính xác
Số lượng vi phạm	22	19	86.3%

Bảng 6. Kết quả thử nghiệm buổi tối

	Thực tế	Module	Độ chính xác
Số lượng vi phạm	22	13	59%

Tương tự với các kết quả trong trường hợp phát hiện lỗi vi phạm vượt đèn đỏ, từ các Bảng 4, 5, 6 cho thấy, độ chính xác trong việc phát hiện các vi phạm dừng đỗ sai trong môi trường có ánh sáng kém chỉ đạt mức tạm ổn, điều này là do khả năng nhận dạng phương tiện của mô hình YOLOv3 thử nghiệm không tốt.

Chú ý kết quả thử nghiệm với module dừng đỗ sai nhóm tác giả không đánh giá số khung hình vi phạm của từng đối tượng vì đây là trường hợp đối tượng không di chuyển nên

kết quả đối với số khung hình thực tế bằng với kết quả module phát hiện.

4. Kết luận

Bài báo đã đề xuất một giải pháp hoàn chỉnh và có thể cài đặt trong thực tế giúp phát hiện các vi phạm về vượt đèn đỏ và đậu đỗ sai quy định. Giải pháp đưa ra có độ chính xác cao trong môi trường đủ ánh sáng và đáp ứng yêu cầu về xử lý thời gian thực với cấu hình máy tính thử nghiệm. Việc sử dụng mô hình mạng YOLOv3 tỏ ra phù hợp với việc phát hiện các phương tiện tham gia giao thông. Tuy nhiên, với hạn chế về bộ dữ liệu phương tiện giao thông nên độ chính xác vẫn chưa hoàn toàn tuyệt đối và chỉ đạt mức ổn định trong môi trường ban ngày, mặt khác, hiện tượng các phương tiện giao thông di chuyển quá nhanh cũng làm giảm khả năng nhận dạng phương tiện của mô hình. Thuật toán theo dõi đối tượng và xác định vi phạm hoạt động tốt tuy nhiên còn phụ thuộc vào khả năng nhận dạng phương tiện tham gia giao thông.

TÀI LIỆU THAM KHẢO

- [1] Aaron Christian P. Uy, Ana Riza F. Quiros, Rhen Anjerome Bedruz, Alexander Abad, Argel Bandala, Edwin Sybingco, Elmer P. Dadios; "Automated Traffic Violation Apprehension System Using Genetic Algorithm and Artificial Neural Network"; *2016 IEEE Region 10 Conference (TENCON)*; Singapore.
- [2] Aaron Christian P. Uy, Rhen Anjerome Bedruz, Ana Riza Quiros, Argel Bandala, Elmer P. Dadios; "Machine Vision for Traffic Violation Detection System through Genetic Algorithm"; *2015 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*; Philippines.
- [3] Xiaoling Wang, Li-Min Meng, Biao Zhao Zhang, Junjie Lu, K.-L. Du; "A Video-based Traffic Violation Detection System"; *Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)*; Shenyang, China.
- [4] Julien A. Vijverberg, Nick A.H.M de Koning, Jungong Han, Peter H.N. de With, Dion Cornelissen; "High-Level Traffic-Violation Detection for Embedded Traffic Analysis"; *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*; Honolulu, USA.
- [5] Jakub Špaňhel, Jakub Sochor, Aleksej Makarov; "Detection of Traffic Violations of Road Users Based on Convolutional Neural Networks"; *2018 14th Symposium on Neural Networks and Applications (NEUREL)*; Belgrade, Serbia.
- [6] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi; "You Only Look Once: Unified, Real-Time Object Detection"; *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; Las Vegas, NV, USA.
- [7] Trung tâm vi mạch Đà Nẵng; "Hệ thống camera giám sát thông minh"; <http://camera.centic.vn>
- [8] "Common Object In Context"; <http://cocodataset.org/>.
- [9] Ayoosh Kathuria; "What's new in YOLO v3?"; <http://towardsdatascience.com>
- [10] J. Redmon; "Darknet"; <https://github.com/pjreddie/darknet>
- [11] Jonathan Hui; "Real-time Object Detection with YOLO, YOLOv2 and now YOLOv3"; http://medium.com/@jonathan_hui/