



Tạp chí điện tử  
Khoa học và Công nghệ Giao thông  
Trang website: <https://jstt.vn/index.php/vn>



#### Article info

##### Type of article:

Original research paper

##### DOI:

<https://doi.org/10.58845/jstt.utt.2025.vn.5.2.69-80>

##### \*Corresponding author:

Email address:

[phamnhupham@gmail.com](mailto:phamnhupham@gmail.com)

**Received:** 27/04/2025

**Received in Revised Form:**  
22/05/2025

**Accepted:** 09/06/2025

## Application of YOLOv11 algorithm for detecting license plate of vehicles

Pham Nhu Pham\*, Pham Van Dieu, Nguyen Minh Nhut  
Gia Lai College, Gia Lai, Vietnam

**Abstract:** Artificial intelligence, machine learning and deep learning are increasingly developing, deep learning takes the idea from human neural networks to build intelligent algorithms. Deep learning models of artificial intelligence have the ability to learn from existing data, to process and make decisions quickly and automatically. This article presents the research and application of the YOLOv11 model to recognize license plates. The authors use the YOLOv11n model to retrain a custom dataset, including license plate images and license plate characters. The training results show that the model achieves high accuracy mAP over 99.4%, fast processing time, proving that the model is feasible when applied in practice.

**Keywords:** YOLOv11, license plate recognition, model training, artificial intelligence, deep learning.



**Thông tin bài viết**

**Dạng bài viết:**

Bài báo nghiên cứu

**DOI:**

<https://doi.org/10.58845/jstt.utt.2025.vn.5.2.69-80>

**\*Tác giả liên hệ:**

Địa chỉ Email:

[phanhupham@gmail.com](mailto:phanhupham@gmail.com)

**Ngày nộp bài:** 27/04/2025

**Ngày nộp bài sửa:** 22/05/2025

**Ngày chấp nhận:** 09/06/2025

## Nghiên cứu và ứng dụng thuật toán YOLOv11 để nhận diện biển số xe

Phạm Như Phẩm\*, Phạm Văn Điều, Nguyễn Minh Nhựt  
Trường Cao đẳng Gia Lai, Gia Lai, Việt Nam

**Tóm tắt:** Trí tuệ nhân tạo, học máy và học sâu ngày càng phát triển, học sâu lấy ý tưởng từ mạng nơ ron thần kinh con người để xây dựng các thuật toán thông minh. Các mô hình học sâu của trí tuệ nhân tạo có khả năng học hỏi từ dữ liệu đã có, để xử lý và đưa ra quyết định một cách tự động nhanh chóng. Bài báo này trình bày về nghiên cứu và ứng dụng mô hình YOLOv11 để nhận diện biển số xe. Nhóm tác giả sử dụng mô hình YOLOv11n để huấn luyện lại bộ dữ liệu tùy chỉnh riêng, gồm hình ảnh biển số xe và kí tự biển số. Kết quả huấn luyện cho thấy mô hình đạt được độ chính xác cao mAP trên 99,4%, thời gian xử lý nhanh, chứng tỏ mô hình có tính khả thi khi áp dụng trong thực tế.

**Từ khóa:** YOLOv11, nhận diện biển số xe, huấn luyện mô hình, trí tuệ nhân tạo, học sâu.

### 1. Giới thiệu

Phát hiện đối tượng theo thời gian thực là một chủ đề rất quan trọng trong hệ thống thị giác máy tính. Nó thường được ứng dụng trong các lĩnh vực như: Theo dõi đối tượng, lái xe tự động, robot, xử lý hình ảnh y khoa, hệ thống giám sát giao thông thông minh [1,2]. Phát hiện đối tượng có hai hướng tiếp cận: Hướng tiếp cận truyền thống sử dụng đặc trưng tự thiết kế như Haar-like, HOG (Histogram of Oriented Gradients), SIFT (Scale Invariant Feature Transform) và hướng tiếp cận hiện đại dựa trên mạng học sâu như CNN (Convolutional Neural Networks), R-CNN (Region based Convolutional Neural Networks), Fast R-CNN, YOLO (You Only Look Once) [2]. Thuật toán phát hiện đối tượng của YOLO, sử dụng trí tuệ nhân tạo để phát hiện, phân loại đối tượng một cách nhanh chóng, chính xác. Do đó YOLO đã được nhiều công trình khoa học [3,4,5], nghiên cứu đề xuất, sử dụng mô hình YOLO để nhận diện biển số xe.

Nhận diện biển số xe thông qua công nghệ xử lý ảnh đã thu hút nhiều nhà nghiên cứu và cũng

có nhiều công trình khoa học được công bố. Mặc dù đã có sự tiến bộ đáng kể trong công nghệ nhận diện xử lý hình ảnh, nhưng nhận diện biển số xe vẫn là một bài toán thách thức đối với các hệ thống giám sát giao thông, giám sát các bãi giữ xe thông minh, vì có nhiều yếu tố gây khó khăn trong quá trình nhận diện, chẳng hạn như thay đổi về ánh sáng, góc nhìn, góc nghiêng, độ mờ và kích thước biển số xe [6,7].

Bài nghiên cứu [8] có hạn chế là độ chính xác của kết quả nhận diện phụ thuộc vào ánh sáng môi trường, lý do chương trình xây dựng chỉ dựa trên thuật toán xử lý ảnh bằng OpenCV, mà chưa sử dụng trí tuệ nhân tạo vào mô hình phát hiện biển số. Đối với bài nghiên cứu [9], sử dụng thuật toán trí tuệ nhân tạo YOLOv5 huấn luyện mô hình, bộ dữ liệu gồm 12.500 bức ảnh biển số xe, để phát hiện biển số xe. Kết quả huấn luyện mô hình (Bảng 1) đạt được các chỉ số khá cao. Từ Bảng 1, cho thấy hiệu suất dự đoán của các mô hình truyền thống (Faster R-CNN, SSD300) thấp hơn các biến thể mô hình YOLOv5. Trong đó YOLOv5-LSE vượt

trội hơn hẳn so với các mô hình khác ở 3 chỉ số Recall, Precision và mAP(mean Average Precision).

**Bảng 1.** Kết quả huấn luyện các mô hình dự đoán phát hiện biển số xe [9]

| Thuật toán<br>(Algorithms) | Độ nhạy<br>(Recall) | Độ chính xác<br>(Precision) | Độ chính xác trung bình<br>(mAP) |
|----------------------------|---------------------|-----------------------------|----------------------------------|
| Faster R-CNN               | 86,5                | 89,4                        | 92,3                             |
| SSD300                     | 88,3                | 89,6                        | 93,3                             |
| RPnet                      | 95,2                | 94,8                        | 94,2                             |
| YOLOv5                     | 93,5                | 93,4                        | 94,6                             |
| YOLOv5-1                   | 95,4                | 95,2                        | 95,8                             |
| YOLOv5-LSE                 | 96,5                | 97,4                        | 97,1                             |

Trong bài nghiên cứu [10] cũng sử dụng thuật toán trí tuệ nhân tạo. Ở đây nhóm tác giả sử dụng YOLOv8, để huấn luyện mô hình phát hiện biển số xe và nhận diện kí tự trên biển số. Bộ dữ liệu gồm 2.528 bức ảnh biển số xe ô tô Saudi, được chia theo tỉ lệ huấn luyện (70%, 1.769 ảnh), kiểm định (20%, 506 ảnh) và thử nghiệm (10%, 253 ảnh). Ảnh dữ liệu kí tự biển số gồm 1.849 ảnh, trong đó huấn luyện (70%, 1.295 ảnh), kiểm định (20%, 370 ảnh) và thử nghiệm (10%, 184 ảnh). Kết quả huấn luyện của mô hình thể hiện ở Bảng 2,

cho thấy hiệu suất dự đoán của các mô hình đều đạt độ chính xác rất cao, các mô hình trên hoàn toàn có khả thi ứng dụng vào thực tiễn. Mô hình phát hiện biển số xe tốt nhất là YOLOv5x và YOLOv8x nhận diện kí tự chính xác nhất.

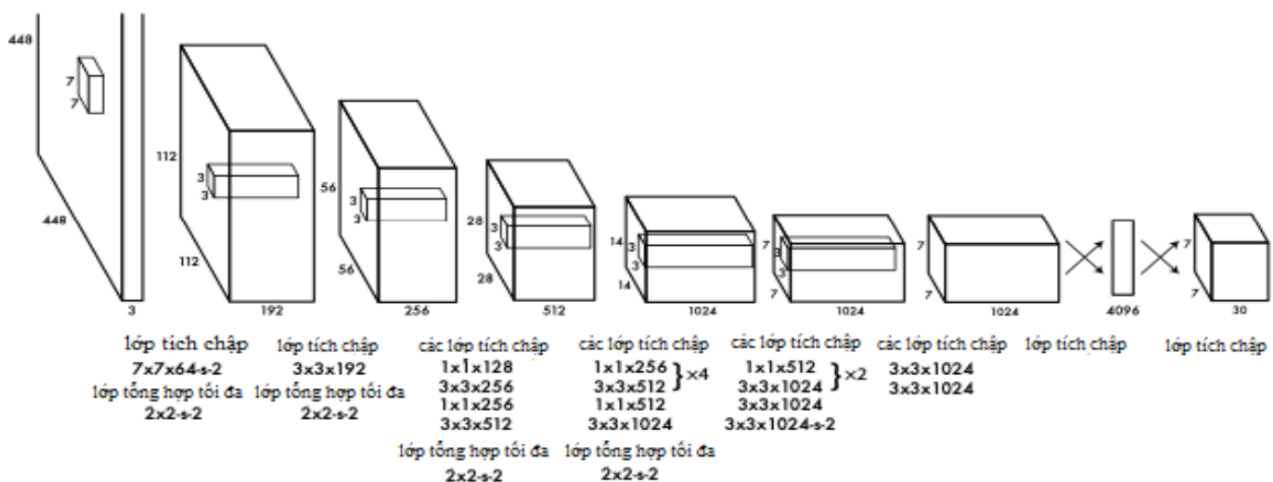
**Bảng 2.** Kết quả huấn luyện các mô hình dự đoán phát hiện biển số và nhận diện kí tự [10]

| Kết quả dự đoán phát hiện biển số |          |           | Kết quả dự đoán nhận diện kí tự |          |           |
|-----------------------------------|----------|-----------|---------------------------------|----------|-----------|
| Mô hình                           | mAP @0.5 | mAP @0.95 | Mô hình                         | mAP @0.5 | mAP @0.95 |
| YOLOv8x                           | 0,973    | 0,844     | YOLOv8x                         | 0,981    | 0,827     |
| YOLOv7x                           | 0,920    | 0,730     | YOLOv7x                         | 0,977    | 0,711     |
| YOLOv5x                           | 0,994    | 0,892     | YOLOv5x                         | 0,978    | 0,819     |

Qua 3 công trình nghiên cứu [8,9,10] trên, nhóm tác giả nhận thấy sử dụng trí tuệ nhận tạo, thuật toán YOLO để nhận diện biển số xe là phương pháp tối ưu nhất. Vì vậy nhóm tác giả tập trung nghiên cứu tìm hiểu và ứng dụng thuật toán YOLOv11 để nhận diện biển số xe. Thuật toán YOLOv11 được huấn luyện lại trên tập dữ liệu riêng của nhóm tác giả. Kết quả huấn luyện đạt được độ chính xác cao, mô hình có khả năng nhận diện biển số xe chính xác và có khả thi áp dụng vào các bãi giữ xe tự động.

## 2. Đối tượng và phương pháp nghiên cứu

### 2.1. Nhận diện vật thể đối với YOLO



**Hình 1.** Kiến trúc của mạng YOLO [1,11,12]

Kiến trúc YOLO (Hình 1) là một mô hình mạng học sâu nơ-ron tích chập. Các lớp tích chập ban đầu của mạng trích xuất các đặc điểm từ hình ảnh, các lớp được kết nối dự đoán xác suất, tọa độ và cho kết quả đầu ra. Mạng YOLO có 24 lớp tích

chập, theo sau là 2 lớp kết nối đầy đủ, thường sử dụng lớp 1x1 thay lớp tích chập 3x3. Với 9 lớp tích chập thay thế cho 24 lớp và một số bộ lọc trong các lớp đó, sẽ được mạng nơ-ron của Fast YOLO [11,12].

Mạng YOLO được ứng dụng để phát hiện, nhận diện đối tượng vật thể, hoạt động như một bài toán hồi quy. Từ hình ảnh đầu vào, qua một mạng gồm các lớp tích chập, tổng hợp và kết nối đầy đủ cho được kết quả đầu ra. Kiến trúc này có thể được tối ưu để chạy trên GPU (Graphics Processing Unit) với một lần chuyển tiếp, vì thế nó đạt được tốc độ rất cao. Thuật toán YOLO có độ chính xác cao và nhanh cho những bài toán nhận diện đối tượng, vô cùng thích hợp cho các ứng dụng thị giác máy [13].

Đầu vào của mô hình YOLO là ảnh (Hình 2), mô hình sẽ nhận dạng ảnh đó gồm có những đối tượng nào, sau đó xác định tọa độ của đối tượng trong bức ảnh. Ảnh đầu vào được chia thành thành  $S \times S$  ô, thường thì sẽ là  $3 \times 3$ ,  $7 \times 7$ ,  $9 \times 9$  .... Đầu ra mô hình là một ma trận 3 chiều có kích thước  $S \times S \times (B*5+C)$  với số lượng tham số mỗi ô là  $(B*5+C)$ . Trong đó, B là số lượng bounding boxes + confidence và C là số class probabilities cho mỗi box cần phải dự đoán. Bounding box gồm 5 thành phần (x, y, w, h, confidence), trong đó (x, y) là tọa độ tâm của bounding box, (w, h) lần lượt là chiều rộng và chiều cao của bounding box và confidence là độ tự tin của dự đoán [1, 11, 13].

## 2.2. Kiến trúc của YOLOv11

YOLOv11 là phiên bản mới của Ultralytics YOLO, được xây dựng trên nền tảng các phiên bản YOLO trước đó [14]. Được công bố tại hội nghị YOLO Vision vào tháng 9 năm 2024. YOLOv11 đánh dấu một bước tiến đáng kể trong công nghệ phát hiện đối tượng theo thời gian thực.

Phần lõi của kiến trúc YOLOv11 bao gồm ba thành phần cơ bản, phần thứ nhất là backbone đóng vai trò là một thành phần quan trọng của kiến trúc YOLO, chịu trách nhiệm trích xuất chính, sử dụng mạng nơ-ron tích chập để chuyển đổi dữ liệu hình ảnh thô đầu vào thành bản đồ tính năng đa tỷ lệ. Thứ hai là thành phần neck hoạt động như một giai đoạn xử lý trung gian, sử dụng các lớp chuyên biệt để tổng hợp và tăng cường các tính năng biểu diễn trên các tỷ lệ khác nhau. Thứ ba, thành phần head hoạt động như cơ chế dự đoán, tạo ra các đầu ra cuối cùng để định vị và phân loại đối tượng

dựa trên các bản đồ tính năng đã tinh chỉnh [15, 16].

Phiên bản mới cải tiến đáng kể cả về kiến trúc lẫn phương pháp huấn luyện, mở rộng giới hạn của độ chính xác và hiệu suất. Thiết kế đổi mới của YOLOv11 kết hợp các kỹ thuật trích xuất đặc trưng tiên tiến, cho phép nắm bắt các chi tiết tinh tế hơn trong khi vẫn giữ số lượng tham số gọn nhẹ. Điều này giúp cải thiện độ chính xác trong nhiều tác vụ thị giác máy tính từ phát hiện đối tượng đến phân loại. Đặc biệt, YOLOv11 đạt được những cải tiến đáng kể về tốc độ xử lý, tăng cường đáng kể khả năng hoạt động theo thời gian thực [14].

YOLOv11 có 5 phiên bản n(nano), s(small), m(medium), l(large) và x(extra-large). Đồ thị so sánh hiệu năng các mô hình huấn luyện của YOLOv11 được mô tả ở Hình 3 và bảng mô tả hiệu suất của các mô hình thể hiện ở Bảng 3. Các mô hình YOLOv11 tạo thành một ranh giới hiệu suất riêng biệt, với mỗi mô hình đều đạt COCO mAP 50-95 cao hơn ở các mức trễ tương ứng các bản YOLO trước đó. Đáng chú ý YOLOv11x đạt được khoảng 54,5% mAP 50-95 ở độ trễ 13ms, vượt qua tất cả các mô hình YOLO trước đó. YOLOv11m thể hiện sự vượt trội hiệu quả, đạt độ chính xác hơn so với các thể hệ YOLO trước đó. Hiệu suất trong chế độ trễ thấp (2-6ms) thì mô hình YOLOv11s duy trì độ chính xác cao khoảng 47% mAP 50-95 [15].

Bảng 4 cho thấy, quy định ảnh trong tập dữ liệu huấn luyện có chiều rộng hay chiều cao tối đa là 640 pixels.

Các thông số: Trung bình của độ chính xác trung bình tại IoU có giá trị từ 50 đến 95 xác định trên tập dữ liệu kiểm định, thời gian xử lý cho mỗi dữ liệu hình ảnh trong tập dữ liệu kiểm định chạy trên các phần cứng CPU, GPU, tham số huấn luyện và số lượng phép tính số thực trong một giây, có giá trị tăng dần từ mô hình YOLOv11n đến mô hình YOLOv11x. Điều này chứng tỏ mô hình YOLOv11n rất nhẹ, số lượng tham số huấn luyện ít, tốc độ nhanh nhất nhưng độ chính xác thấp nhất. Ngược lại thì phiên bản YOLOv11x có độ chính xác cao nhất, nhưng là phiên bản có nhiều tham số nên chạy chậm nhất.

## 2.3. Quy trình nhận diện biên số với YOLOv11

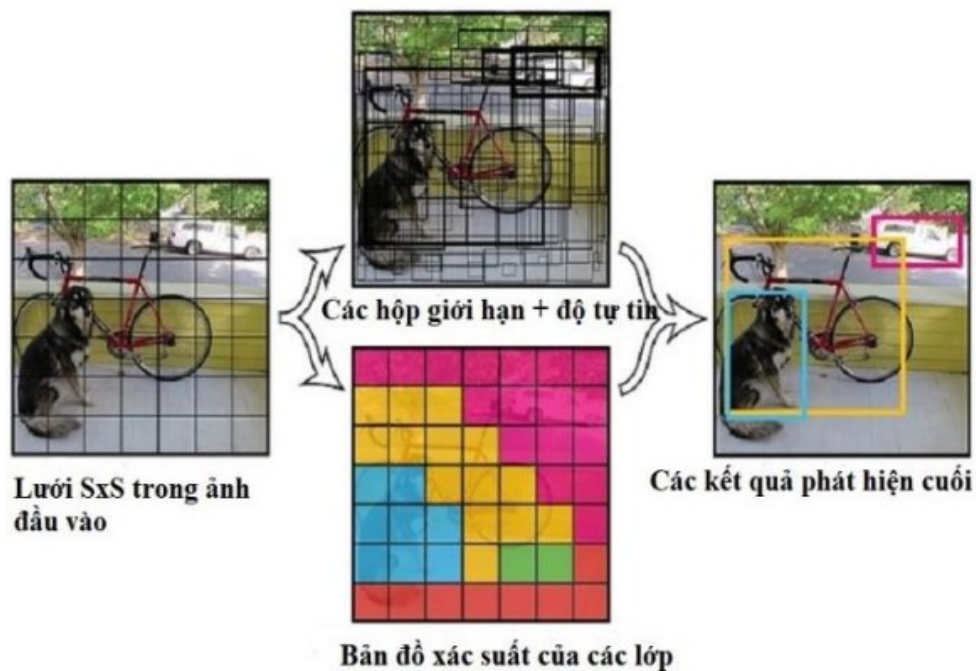
Ứng dụng công nghệ nhận dạng vật thể của YOLOv11, nhóm tác giả xây dựng mô hình bài toán nhận diện biển số xe (Hình 4), gồm có các khối chức năng chính như sau:

Khối thứ nhất: Là khối phát hiện biển số xe, làm nhiệm vụ đọc hình ảnh thông qua video, camera hoặc các ảnh về các phương tiện giao thông có chứa biển số xe, từ đó sử dụng mô hình YOLOv11 đã huấn luyện để phát hiện và khoanh vùng vị trí có chứa biển số trong những bức ảnh đó, sau đó cắt vùng chứa biển số đưa vào khối tiếp

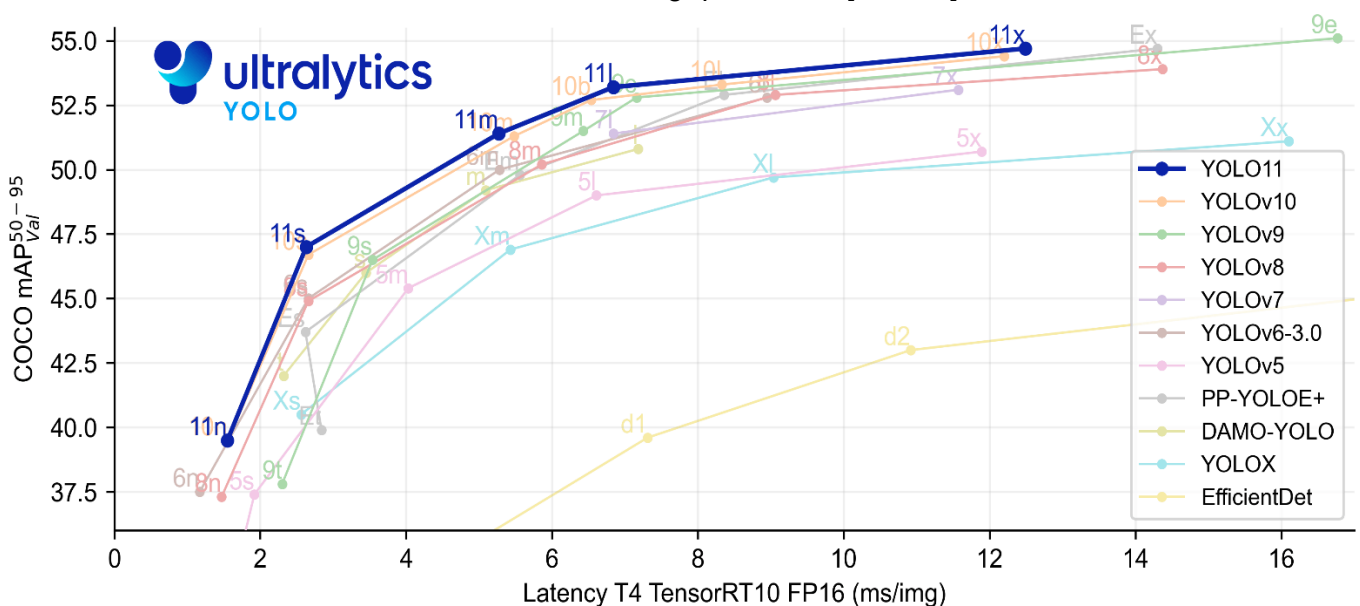
theo để xử lý.

Khối thứ hai: Là khối xoay biến số, làm nhiệm vụ đọc tọa độ của 2 kí tự ngoài cùng dòng dưới (hay dòng trên) của biển số, tính góc nghiêng của biển số so với phương ngang, sau đó xoay biến số theo chiều hợp lý để các kí tự của biển số xe vuông góc với phương ngang nhiều nhất có thể.

Khối thứ ba: Là khối nhận diện kí tự, sử dụng mô hình YOLOv11 đã huấn luyện để nhận dạng từng kí tự trong biển số, trả kết cuối cùng là khoanh vùng kí tự và tên của kí tự đó trên biển số.



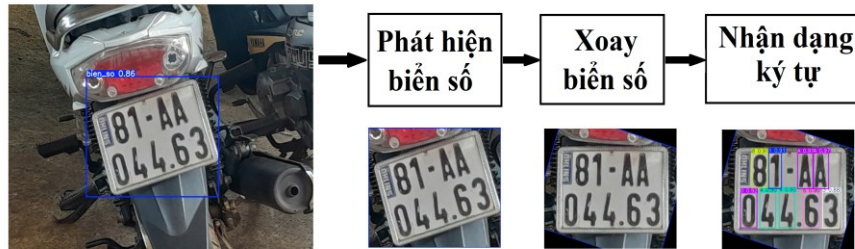
Hình 2. Mô hình tổng quát YOLO [1,11,13]



Hình 3. Biểu đồ so sánh hiệu năng của YOLOv11 với các phiên bản trước [15]

**Bảng 3.** So sánh hiệu suất huấn luyện của các mô hình YOLOv11 [14]

| Mô hình | Kích thước ảnh (pixels) | Độ chính xác trung bình 50-95 (%) | Thời gian xử lý trên CPU (ms) | Thời gian xử lý trên GPU T4 (ms) | Thông số (M) | Số lượng phép tính (B) |
|---------|-------------------------|-----------------------------------|-------------------------------|----------------------------------|--------------|------------------------|
| YOLO11n | 640                     | 39,5                              | 56,1 ± 0,8                    | 1,5 ± 0,0                        | 2,6          | 6,5                    |
| YOLO11s | 640                     | 47,0                              | 90,0 ± 1,2                    | 2,5 ± 0,0                        | 9,4          | 21,5                   |
| YOLO11m | 640                     | 51,5                              | 183,2 ± 2,0                   | 4,7 ± 0,1                        | 20,1         | 68,0                   |
| YOLO11l | 640                     | 53,4                              | 238,6 ± 1,4                   | 6,2 ± 0,1                        | 25,3         | 86,9                   |
| YOLO11x | 640                     | 54,7                              | 462,8 ± 6,7                   | 11,3 ± 0,2                       | 56,9         | 194,9                  |

**Hình 4.** Quy trình các khối nhận diện biển số xe

### 3. Kết quả và thảo luận

#### 3.1. Bộ dữ liệu huấn luyện

Bộ dữ liệu huấn luyện của YOLOv11 có 2 thư mục lớn là images và labels. Trong mỗi thư mục images và labels sẽ chứa 2 thư mục con là train và val. Dữ liệu được chia theo tỉ lệ thư mục train chiếm khoảng 80% và thư mục val chiếm khoảng 20%. Thư mục images chứa những bức ảnh có kích thước 640x640 pixels theo quy định của YOLO,

thư mục labels sẽ chứa nhãn của các ảnh dữ liệu trên. Nó là tọa độ 4 góc của hộp giới hạn đối tượng, được vẽ bởi phần mềm MakeSense. MakeSense là công cụ gán nhãn dữ liệu hình ảnh miễn phí, mã nguồn mở và hoạt động trực tuyến trên trình duyệt mà không cần cài đặt [17]. Giao diện đơn giản, trực quan, dùng chuột để vẽ vùng chứa đối tượng trong ảnh, tạo ra các nhãn. MakeSense có thể dùng AI để hỗ trợ gợi ý tạo nhãn hoặc gán nhãn tự động.

**Hình 5.** Ảnh đại diện của bộ dữ liệu phát hiện biển số và bộ dữ liệu nhận diện ký tự biển số

Trong mô hình nhận diện biển số xe này, tác giả đã xây dựng 2 bộ dữ liệu như Hình 5. Một là, bộ dữ liệu huấn luyện mô hình phát hiện biển số xe, gồm có 8.855 bức ảnh biển số các loại xe (ô tô con, ô tô tải, xe máy, xe điện), trong đó thư mục train chứa 7.050 ảnh và thư mục val chứa 1.805 ảnh. Hai là, bộ dữ liệu huấn luyện mô hình nhận diện ký tự biển số, gồm có 4.025 bức ảnh chứa ký tự biển số (gồm biển số 2 dòng và biển số 1 dòng), trong đó thư mục train chứa 3.210 ảnh và thư mục val chứa 815 ảnh.

Bộ dữ liệu trên, một được tác giả siêu tầm, thu thập trong các dự án của các nhà nghiên cứu trong nước. Một phần tác giả tự chụp trực tiếp hình ảnh các phương tiện giao thông, ở nhiều góc độ, khoảng cách, độ sáng khác nhau, nhằm tạo sự đa dạng bộ dữ liệu. Ngoài ra để tăng thêm độ khó của mô hình phát hiện biển số, tác giả đã sử dụng OpenCV để sao chép tăng thêm các hình ảnh có đường biên, độ sáng, góc nghiêng, độ mờ, phóng to, thu nhỏ khác nhau, nhằm kỳ vọng mô hình nhận diện tốt nhất.

### 3.2. Huấn luyện mô hình

Các mô hình huấn luyện sẵn của YOLOv11 (11n, 11s, 11m và 11x) không phát hiện được đối tượng trong nghiên cứu này là phát hiện vị trí biển số trong ảnh và nhận diện ký tự của biển số. Vì vậy, tác giả cần phải huấn luyện lại mô hình với tập dữ liệu là hình ảnh của biển số và ký tự.

Để việc huấn luyện mô hình diễn ra nhanh

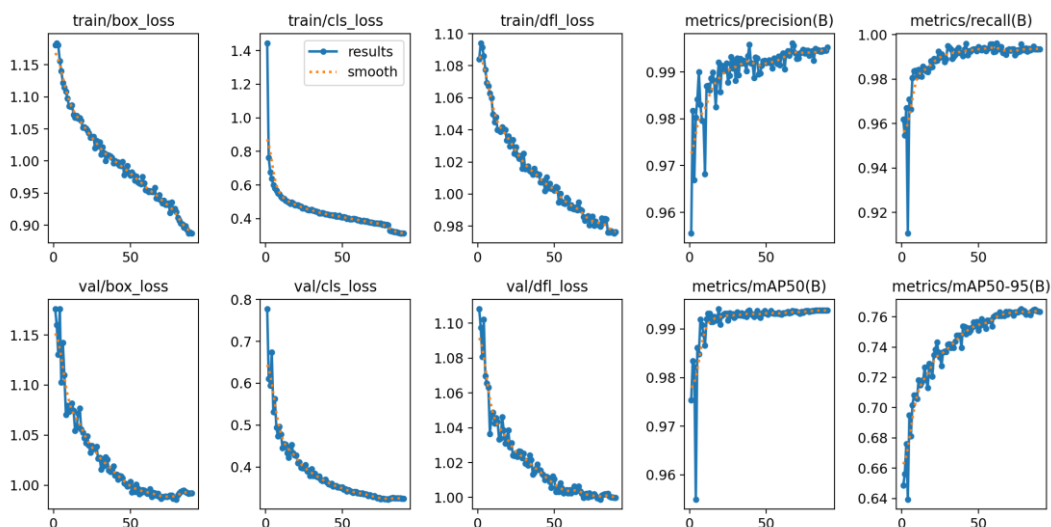
chóng, hiệu quả, đòi hỏi máy tính phải có cấu hình mạnh, đặc biệt có bộ phận xử lý đồ họa GPU để xử lý ảnh. Do đó chúng tôi sử dụng công cụ miễn phí Google Colab của Google và mô hình YOLOv11n của YOLOv11 để huấn luyện dữ liệu. Lệnh thực thi huấn luyện trên Google Colab có cấu trúc `!yolo train model = yolo11n.pt data = coco8.yaml epochs = 90 imgsz = 640`. Các tham số chính model, data, epochs, imgsz được thiết lập phù hợp với mục đích và yêu cầu của YOLO. Ngoài ra, còn có một số tham số khác đã được Google Colab thiết lập mặc định như learning rate ( $lr=0,01$ ); batch size = 16; optimizer = auto (SGD); momentum = 0,937; box = 7,5; cls = 0,5; dfl = 1,5; patience = 100; device = None; weight\_decay = 0.0005;...

### 3.3. Kết quả huấn luyện mô hình

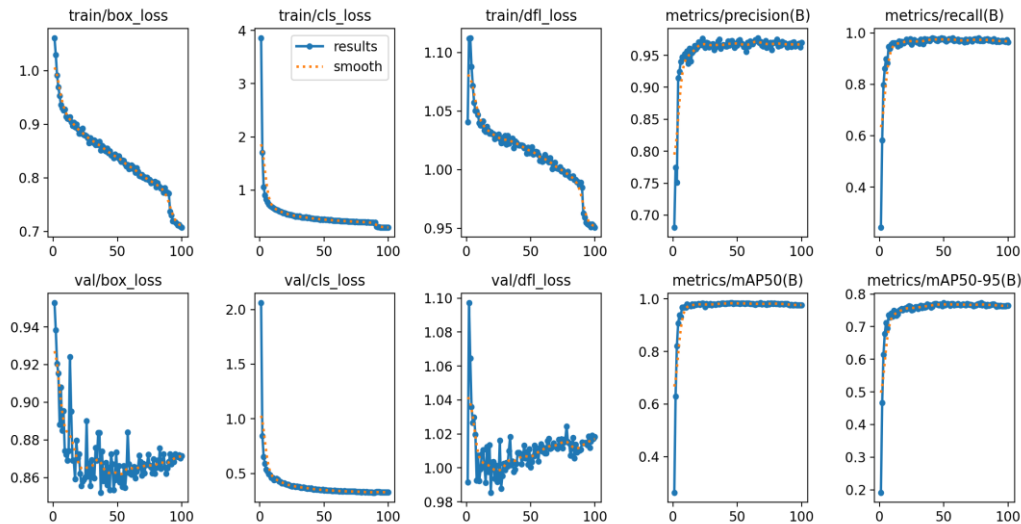
Thực hiện huấn luyện 90 epochs trên bộ dữ liệu phát hiện biển số và 100 epochs trên bộ dữ liệu nhận diện ký tự, kết quả huấn luyện đạt được các thông số ở Bảng 4.

**Bảng 4.** Kết quả huấn luyện bộ dữ liệu phát hiện biển số và nhận diện ký tự

| Bộ dữ liệu        | Precision | Recall | mAP 50 | mAP 50-95 |
|-------------------|-----------|--------|--------|-----------|
| Phát hiện biển số | 0,994     | 0,996  | 0,994  | 0,765     |
| Nhận diện ký tự   | 0,97      | 0,972  | 0,984  | 0,773     |



**Hình 6.** Đường cong hiệu suất mô hình huấn luyện đối với bộ dữ liệu phát hiện biển số



**Hình 7.** Đường cong hiệu suất mô hình huấn luyện đối với bộ dữ liệu nhận diện kí tự biển số

Trong huấn luyện mô hình, chỉ số độ chính xác trung bình mAP@50 luôn cao hơn độ chính xác trung bình mAP@50-95. Vì mAP@50 là trung bình của AP tại ngưỡng IoU=0,5, còn mAP@50-95 là trung bình của AP từ ngưỡng IoU=0,5 đến 0,95. Bảng 4 cho thấy độ chính xác trung bình mAP@50 rất cao, đặc biệt mAP@50 của bộ dữ liệu phát hiện biển số. Điều này khẳng định mô hình hoạt động cực kỳ ổn định, khả năng phát hiện biển số trong điều kiện bình thường là 99,4%.

Đường cong hiệu suất huấn luyện mô hình (Hình 6 và Hình 7) cho thấy ba chỉ số mất mát: mất mát hộp bao vật thể (box\_loss), mất mát phân lớp (cls\_loss) và mất mát tiêu điểm phân phối (df\_l\_loss) của hai mô hình giảm đều, cho thấy kết quả huấn luyện của hai mô hình đạt hiệu quả.

Từ những kết quả trên, cho thể thấy mô hình huấn luyện cho kết quả khá tốt, đặc biệt là kết quả dự đoán precision, recall và mAP@50 có độ chính xác rất cao.

### 3.4. Thục nghiệm mô hình

#### 3.4.1. Lưu đồ thuật toán

Lưu đồ thuật toán quy trình nhận diện biển số (Hình 8), được mô tả gồm các bước thực hiện sau:

- Đọc và hiển thị ảnh gốc: Đọc ảnh đầu vào từ đường dẫn đã được chỉ định hoặc camera; Hiển thị ảnh gốc lên giao diện chương trình Python.
- Dùng mô hình YOLOv11 nhận diện vùng chứa biển số: Sử dụng mô hình YOLOv11n đã

được huấn luyện để phát hiện vùng chứa biển số; Vẽ các hộp giới hạn (bounding boxes) cho vùng chứa biển số trên ảnh gốc.

- Kiểm tra độ tin cậy về nhận diện vùng chứa biển số: Độ tin cậy chính xác phải lớn hơn 50%; Nếu thỏa mãn thì thực hiện bước tiếp theo, nếu không thì kết thúc chu trình.

- Xử lý tách vùng chứa biển số đã nhận diện được: Lặp qua từng vùng chứa biển số; Cắt vùng chứa biển số từ ảnh gốc.

- Tính góc nghiêng và xoay biển số: Tìm tọa độ của 2 kí tự ngoài cùng dòng dưới (hay dòng trên) của biển số; Tìm hệ số góc của đường thẳng đi qua 2 điểm đó; Tính được góc tạo bởi đường thẳng và trục hoành (góc nghiêng); Xoay biển số theo chiều của góc đó.

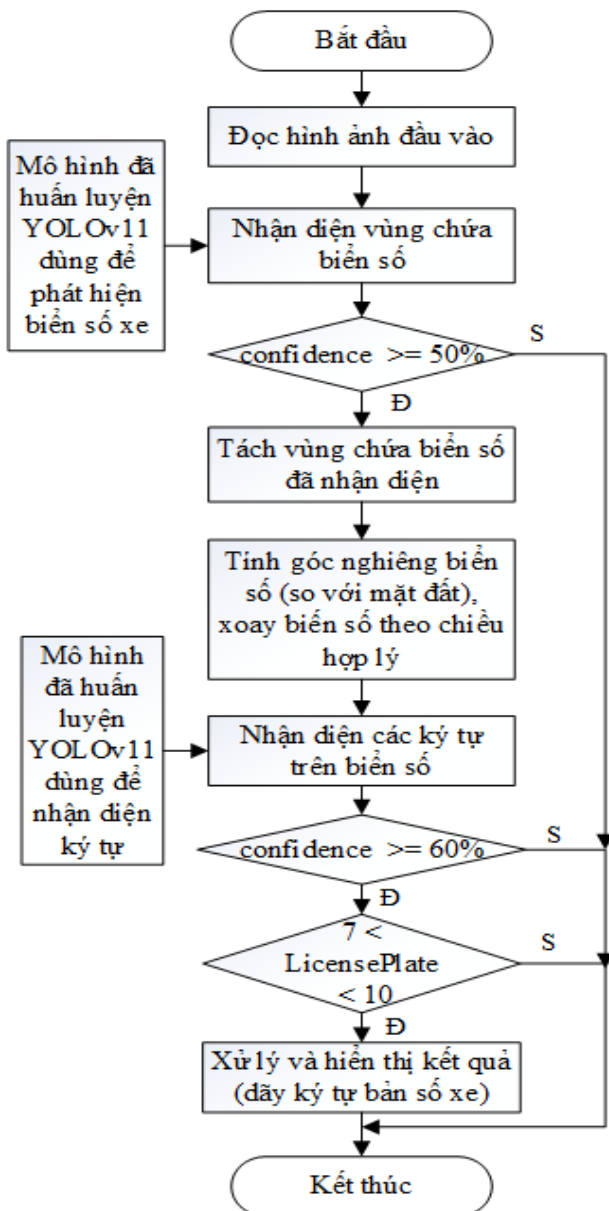
- Sử dụng mô hình YOLOv11 nhận diện các kí tự trong biển số: Sử dụng mô hình YOLOv11n đã huấn luyện để nhận diện các kí tự trong biển số.

- Kiểm tra độ tin cậy về các kí tự đã nhận diện: Độ tin cậy chính xác phải lớn hơn 60%; Nếu thỏa mãn thì thực hiện bước tiếp theo, nếu không thì kết thúc chu trình.

- Kiểm tra số lượng các kí tự đã nhận diện phù hợp với biển số: Độ dài các kí tự trên biển số phải lớn hơn 7 kí tự và nhỏ hơn 10 kí tự; Nếu thỏa mãn thì thực hiện bước tiếp theo, nếu không thì kết thúc chu trình.

- Hiển thị kí tự biển số: Ghép các kí tự vừa nhận diện theo thứ tự để có kết quả biển số đúng

với thực tế; Hiển thị và lưu kí tự.



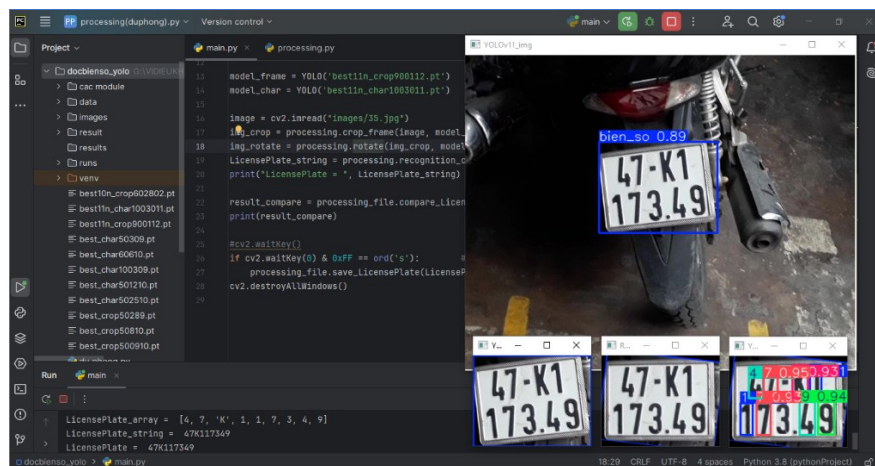
Hình 8. Lưu đồ thuật toán chương trình nhận diện

### 3.4.2. Kết quả thực nghiệm mô hình

Ngôn ngữ sử dụng trong nghiên cứu này là Python phiên bản 3.8 và trình soạn thảo phát triển tích hợp (IDE) Pycharm. Dữ liệu đầu vào mô hình là hình ảnh phần sau của xe gắn máy có chứa biển số, dựa vào thuật toán đã huấn luyện, sau các bước thực hiện, chương trình đã phát hiện, khoanh vùng biển số với tỉ lệ dự đoán là 89%. Tính toán góc nghiêng so với phương ngang là 6.4 độ, xoay biển số vuông góc và nhận dạng từng kí tự với kết quả dự đoán rất cao trên 90% cho các kí tự. Kết quả cuối cùng chương trình nhận diện, đọc biển số xe khá chính xác như Hình 9.

Thử nghiệm nhiều loại biển số khác nhau, nhóm tác giả thấy mô hình phát hiện và nhận diện được hầu hết các kí tự của biển số. Tuy nhiên, đối với một số biển số bị mờ nhiều, chòng chéo nhiều lớp, nhiều cao hoặc góc chụp bị nghiêng nhiều, thì mô hình vẫn phát hiện được biển số nhưng nhận diện kí bị sai lệch hoặc không nhận diện được như Hình 10. Trường hợp trong điều kiện bình thường, biển số có độ nghiêng ít và có kí tự bị che khuất (thiếu kí tự biển số), quá trình đọc tọa độ hai kí tự ngoài cùng (trừ kí tự bị che khuất) của dòng biển số, để tính hệ số góc nghiêng vẫn theo quy trình bình thường, nên không ảnh hưởng đến quá trình xoay biển số, kết quả nhận diện kí tự vẫn đảm bảo Hình 11.

Chương trình hiển thị kí tự biển số, được tác giả lập trình cho hiển thị khi kết quả nhận diện chính xác, đầy đủ kí tự biển số. Nếu kết quả nhận diện kí tự bị thiếu thì chương trình không cho hiển thị và thực hiện thao tác nhận diện lại.



Hình 9. Kết quả nhận diện biển số của mô hình



**Hình 10.** Một số biển số bị mờ, nhiễu và nghiêng nhiều, mô hình vẫn phát hiện được vị trí biển số nhưng nhận diện kí tự bị sai lệch.



**Hình 11.** Quá trình đọc tọa độ, tính hệ số góc nghiêng đối với các trường hợp biển số có độ nghiêng ít và thiếu kí tự ngoài cùng

### 3.5. Đánh giá và bàn luận kết quả

Qua kết quả thực nghiệm mô hình nhận diện biển số xe đối với thuật toán YOLOv11, tác giả nhận thấy mô hình nhận diện khá tốt, khả năng dự đoán chính xác trên 90% cho các loại biển số xe trong điều kiện bình thường.

Nhằm kiểm chứng độ dự đoán chính xác của các phiên bản YOLO, nhóm tác giả đã sử dụng bộ dữ liệu trên (Hình 5), huấn luyện lại mô hình với phiên bản YOLOv9 và YOLOv10. Kết quả huấn luyện được trình bày ở Bảng 5, cho thấy Mô hình YOLOv11n đạt Recall cao nhất (0,996 và 0,972) cho cả hai tác vụ phát hiện biển số và nhận diện kí tự, điều này cho thấy khả năng phát hiện đối tượng của mô hình rất cao, ít khi bỏ sót. YOLOv9t cho kết quả nhận diện kí tự tốt nhất trong ba phiên bản trên, với Precision = 0,983 mAP50-95 = 0,775,

chứng tỏ mô hình nhận diện kí tự có độ chính xác rất cao. Nhìn chung sự chênh lệch các chỉ số huấn luyện giữa các phiên bản YOLOv9, YOLOv10 và YOLOv11 là không đáng kể, nên tại thời điểm hiện tại phiên bản YOLOv11 vẫn có độ chính xác cao.

Tuy nhiên, khi so sánh kết quả trên (Bảng 5) với Bảng 2 của nghiên cứu [10], thì chỉ số mAP50-95 của nhóm tác giả nghiên cứu thấp hơn. Điều này cho thấy kết quả huấn luyện của các mô hình phụ thuộc vào bộ dữ liệu. Một bộ dữ liệu có nhiều mẫu đơn giản, ít nhiễu thì mAP sẽ cao, bộ dữ liệu có nhiều mẫu phức tạp thì mAP thấp hơn. Nghiên cứu của nhóm tác giả có bộ dữ liệu phức tạp, nhiều loại biển số xe (ô tô con, ô tô tải, xe máy và xe điện), trong khi nghiên cứu [10] có bộ dữ liệu chủ yếu là biển số xe ô tô Saudi nên mAP50-95 của nhóm tác giả sẽ thấp hơn.

Tiến hành thực nghiệm với hình ảnh đầu vào là biển số xe (47K117348) Hình 9, nhóm tác giả đã nhận thấy các thông số: thời gian nhận diện xử lý, thời gian tiền xử lý, thời gian suy luận, thời gian hậu xử lý cho mỗi bức ảnh và kết quả dự đoán có sự thay đổi tăng dần hiệu suất ở các phiên bản YOLO sau (Bảng 6). Cụ thể thời gian nhận diện xử lý cũng như tốc độ suy luận của phát hiện biển số và nhận diện kí tự ở phiên bản YOLOv11 là nhanh

nhất (177.9 ms, 164.7ms), trong khi đó ở phiên bản YOLOv10 (209.6 ms, 191.3ms) và YOLOv9 (357.4 ms, 331.1ms) thì chậm hơn.

Vậy, phiên bản YOLOv11 với những kiến trúc thiết kế cải tiến, phương pháp huấn luyện được tối ưu đã mang lại tốc độ xử lý nhanh và độ chính xác cao nhất ở thời điểm hiện tại. Điều này được ứng dụng nhiều trong bài toán phát hiện đối tượng theo thời gian thực.

**Bảng 5.** Kết quả huấn luyện mô hình phát hiện biển số và nhận diện kí tự của các phiên bản YOLO

| Mô hình  | Kết quả dự đoán phát hiện biển số |        |       |          | Kết quả dự đoán nhận diện kí tự |        |       |          |
|----------|-----------------------------------|--------|-------|----------|---------------------------------|--------|-------|----------|
|          | Precision                         | Recall | mAP50 | mAP50-95 | Precision                       | Recall | mAP50 | mAP50-95 |
| YOLOv11n | 0,994                             | 0,996  | 0,994 | 0,765    | 0,970                           | 0,972  | 0,984 | 0,773    |
| YOLOv10n | 0,981                             | 0,984  | 0,994 | 0,768    | 0,932                           | 0,929  | 0,966 | 0,758    |
| YOLOv9t  | 0,996                             | 0,994  | 0,995 | 0,765    | 0,983                           | 0,967  | 0,984 | 0,775    |

**Bảng 6.** Thời gian xử lý (bức ảnh hình 9) của các phiên bản YOLO cho việc phát hiện biển số và nhận diện kí tự (ms)

| Mô hình | Thời gian xử lý phát hiện biển số |          |           | Thời gian xử lý nhận diện kí tự |          |           |
|---------|-----------------------------------|----------|-----------|---------------------------------|----------|-----------|
|         | Tiền xử lý                        | Suy luận | Hậu xử lý | Tiền xử lý                      | Suy luận | Hậu xử lý |
| YOLOv11 | 5.9                               | 177.9    | 1.4       | 6.5                             | 164.7    | 1.4       |
| YOLOv10 | 6.0                               | 209.6    | 0.5       | 3.3                             | 191.3    | 0.4       |
| YOLOv9  | 6.6                               | 357.4    | 1.4       | 3.9                             | 331.1    | 1.5       |

#### 4. Kết luận

Trong bài báo này, nhóm tác giả đã nghiên cứu, phân tích thuật toán YOLOv11, từ đó xây dựng mô hình bài toán nhận diện biển số xe. Cơ sở dữ liệu là 8.855 bức ảnh chứa biển số xe và 4.025 bức ảnh kí tự, được huấn luyện lại theo mô hình YOLOv11n với kết quả dự đoán khá cao: Precision = 0,994 Recall = 0,996 mAP50 = 0,994 mAP50-95 = 0,773. Thực nghiệm với đầu vào là ảnh của biển số xe bất kỳ trong điều kiện bình thường, kết quả nhận diện mô hình (gồm phát hiện vị trí biển số và nhận diện kí tự) khá chính xác với tỉ lệ dự đoán trung bình trên 90%. Tuy nhiên, đối với trường hợp biển số quá nhỏ, mờ nhiều, chồng chéo nhiều lớp, nhiễu cao, bị che khuất hoặc góc chụp bị nghiêng nhiều, thì mô hình vẫn phát hiện được biển số nhưng nhận diện kí bị sai lệch hoặc không nhận diện được. Do vậy mô hình của chúng tôi cần phải cải thiện thêm để phát triển trong tương lai:

- Bổ sung thêm dữ liệu huấn luyện (đặc biệt

là bộ dữ liệu nhận diện kí tự, các kí tự nghiêng, mờ, nhiễu) để kết quả dự đoán tăng thêm độ chính xác. Ứng dụng thêm các thuật toán khác trong xử lý ảnh, xoay biển số nghiêng, để tăng độ chính xác của kết quả thu được.

- Có thể áp dụng ở các bãi gửi xe, các trạm thu phí, các trạm gác cổng, ... hoặc đề tài có thể kết hợp với cảm biến vân tay, tạo thành cổng tự động nhà xe học sinh sinh viên, nhà trọ.

#### Tài liệu tham khảo

- [1]. N.T. Cường, N.V. Mạnh, L.V. An, T.Đ. Lương, N.T. Tú, N.T.T. Huyền, N.T.B. Thúy. (2024). Nghiên cứu và ứng dụng thuật toán YOLOv7 để phân loại cà chua. *Tạp chí Khoa học Tài nguyên và Môi trường*, 50, 102-112. <https://doi.org/10.63064/khtnmt.2024.569>
- [2]. Đ.T. Dung, H.L.N. Dung, T.L. Chương, T.C. Hào, T.V. Phúc. (2024). Nghiên cứu các phiên bản YOLOv8 và YOLO-NAS trong phát hiện biển số xe. *Tạp chí Khoa học và Công nghệ - Đại học Thái Nguyên*, 229(7), 156-167.

- <https://doi.org/10.34238/tnu-jst.10336>
- [3]. A. Tourani, A. Shahbahrami, C.Y. Suen, S. Khazaei, S. Soroori. (2020). A Robust Deep Learning Approach for Automatic Iranian Vehicle License Plate Detection and Recognition for Surveillance Systems. *IEEE Access*, 8, 201317-201330. doi: 10.1109/ACCESS.2020.3035992
- [4]. Y. Zou, Y. Zhang, J. Yan, X. Jiang, T. Huang, H. Fan, Z. Cui. (2020). A Robust License Plate Recognition Model Based on Bi-LSTM. *IEEE Access*, 8, 211630-211641. doi: 10.1109/ACCESS.2020.3040238
- [5]. M.-A. Chung, Y.-J. Lin, C.-W. Lin. (2024). YOLO-SLD: An Attention Mechanism-Improved YOLO for License Plate Detection. *IEEE Access*, 12, 89035-89045. DOI:10.1109/ACCESS.2024.3419587
- [6]. N.T. Lợi, Đ.X. Phúc, N.T.T. Uyên, N.H. Phát. (2023). Đề xuất Mô hình YOLOv5 ứng dụng trong nhận diện biển số xe. *Tạp chí Khoa học Trường Đại học Mở Hà Nội*, 104, 65-77. DOI:10.59266/houjs.2023.277
- [7]. S. Pan, J. Liu, D. Chen. (2022). Research on License Plate Detection and Recognition System based on YOLOv7 and LPRNet. *Academic Journal of Science and Technology*, 4(2), 62-68. <https://doi.org/10.54097/ajst.v4i2.3971>
- [8]. T.T. Hương, N.T.K. Hằng. (2021). Kỹ thuật nhận dạng biển số xe và ứng dụng vào bài toán quản lý bãi giữ xe tại Trường Đại học Đồng Tháp, 10(3), 115-120. <https://doi.org/10.52714/dthu.10.3.2021.875>
- [9]. H. Shi, D. Zhao (2023). License Plate Recognition System Based on Improved YOLOv5 and GRU. *IEEE Access*, 11, 10429-10439. <https://doi.org/10.1109/ACCESS.2023.3240439>
- [10]. M. Safran, A. Alajmi, S. Alfarhood. (2024). Efficient Multistage License Plate Detection and Recognition Using YOLOv8 and CNN for Smart Parking Systems. *Journal of Sensors*, 2024, 4917097. <https://doi.org/10.1155/2024/4917097>
- [11]. J. Redmon, S. Divvala, R. Girshick, A. Farhadi. (2016). You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779-788. <https://doi.org/10.1109/CVPR.2016.91>
- [12]. G. Lavanya, S.D. Pande. (2024). Enhancing Real-time Object Detection with YOLO Algorithm. *EAI Endorsed Transactions on Internet of Things*, 10, 1-9. <https://doi.org/10.4108/eetiot.4541>
- [13]. T.Q. Bảo, N.Đ. Thiện, L.T. Thắng, N.V. Thắng. (2022). Nghiên cứu thuật toán thị giác máy tính để theo dõi việc thực hiện các quy định về phòng ngừa Covid-19 sử dụng kỹ thuật học sâu. *Tạp chí Khoa học Đại học Cần Thơ*, 58(5), 1-13. <https://doi.org/10.22144/ctu.jvn.2022.217>
- [14]. Ultralytics. <https://docs.ultralytics.com/vi/models/yolo11> (truy cập ngày: 22/04/2025)
- [15]. R. Khanam, M. Hussain. (2024). YOLOV11: An overview of the key Architectural Enhancements. <https://doi.org/10.48550/arXiv.2410.17725>
- [16]. M.A.R. Alif. (2024). YOLOV11 For Vehicle Detection: Advancements, Performance and Applications in Intelligent Transportation Systems. <https://doi.org/10.48550/arXiv.2410.22898>
- [17]. VISO. <https://viso-ai.translate.google.computer-vision/image-annotation/? x tr sl=en& x tr tl=vi& x tr hl=vi& x tr pto=wa> (truy cập ngày: 19/05/2025)