

# ỨNG DỤNG MẠNG YOLOV8 TRONG PHÁT HIỆN HƯ HỎNG TRÊN MẶT ĐƯỜNG BỘ

## TOWARDS ROAD DAMAGE DETECTION BY USING YOLOV8

NGUYỄN HỮU TUÂN\*, NGUYỄN DUY TRƯỜNG GIANG

Khoa Công nghệ Thông tin, Trường Đại học Hàng hải Việt Nam

\*Email liên hệ: huu-tuan.nguyen@vimaru.edu.vn

### Tóm tắt

Các hư hỏng trên mặt đường bộ (ổ gà, ổ trâu, các vết nứt, hỏng - tiếng Anh gọi chung là potholes) là một trong các nguyên nhân chủ yếu gây ra tai nạn giao thông, thiệt hại đối với con đường và các phương tiện giao thông. Việc phát hiện kịp thời và chính xác để từ đó có các biện pháp tu sửa, nâng cấp các hư hỏng này sẽ góp phần giảm tai nạn giao thông và thiệt hại đối với hệ thống đường bộ và phương tiện giao thông. Điều này cũng giúp các nhà quản lý lập kế hoạch bảo dưỡng, bảo trì đường hợp lý và tối ưu hơn. Tuy nhiên, việc phát hiện các hư hỏng trên mặt đường bộ một cách thủ công là một công việc nhàm chán, mất thời gian và không phải lúc nào cũng khả thi, kịp thời. Trong bài báo này, nhóm tác giả đề xuất một giải pháp cho bài toán phát hiện hư hỏng trên mặt đường bộ bằng cách áp dụng mô hình mạng học sâu YOLOv8 trên dữ liệu video thu được từ camera giám sát của con đường cần đánh giá. Kết quả huấn luyện trên một tập dữ liệu hình ảnh gồm 19074 ảnh cho thấy hệ thống đề xuất có độ chính xác huấn luyện tốt (độ chính xác trung bình - mean Average Precision  $mAP@0.5$  là 87,5% với mô hình YOLOv8x sau 200 epoch huấn luyện). Kết quả thực nghiệm khi sử dụng mô hình đã huấn luyện với dữ liệu mặt đường thu thập từ thực tế cũng cho thấy cách tiếp cận sử dụng mạng YOLOv8 cho độ chính xác cao và có khả năng áp dụng trong thực tế.

**Từ khóa:** Phát hiện hư hỏng trên mặt đường bộ, YOLOv8, pothole detection.

### Abstract

Damages on roads (small holes, large holes, and cracks etc. - potholes in general) are one of the main cause to traffic accidents, disasters and vehicles' devastations. Timely and accurate detection of these potholes and then repair them will help to reduce accidents and damages to roads and vehicles together. This also provides

good supports for managers to plan road maintenance and repair more appropriately and optimally. However, manually detecting and measuring road surface damage is a tedious, time-consuming task that is not always feasible and timely. In this article, we propose a solution to the problem of detecting damage on road surfaces (potholes) by applying the YOLOv8 deep learning network model on video data obtained from surveillance cameras. Training results on an image dataset of 19074 images show that the proposed system has good training accuracy ( $mAP@0.5$  is 87.5% with the YOLOv8x model after 200 training epochs). Experimental results of the trained model upon real-life road videos also show that the approach using the YOLOv8 network has high accuracy and is applicable in practice.

**Keywords:** Road damage detection, YOLOv8, pothole detection.

### 1. Mở đầu

Các hư hỏng trên mặt đường bộ (potholes) có thể là hậu quả của việc thi công kém chất lượng, ảnh hưởng của thời tiết (mưa), sự phá hoại hoặc đơn giản là hậu quả của một quá trình sử dụng lâu dài mà không có sự sửa chữa bảo trì mặt đường. Việc sửa chữa các hư hỏng này là một nhiệm vụ cấp thiết đối với các cơ quan quản lý và bảo trì đường bộ để có thể giảm sự hư hỏng của các con đường, giảm tai nạn giao thông và các thiệt hại có thể xảy ra đối với các phương tiện tham gia giao thông. Để có thể xây dựng các kế hoạch sửa chữa, bảo trì và nâng cấp chất lượng mặt đường một cách hợp lý và tối ưu các nguồn lực sử dụng thì việc phát hiện chính xác các hư hỏng trên mặt đường là một bài toán cần phải giải quyết một cách chính xác và hiệu quả. Cách tiếp cận thủ công sử dụng con người di chuyển và quan sát bằng mắt thường là một quá trình tốn thời gian, tốn kém nhân lực và không phải luôn khả thi và có thể gây nguy hiểm cho nhân lực thực hiện. Công việc quan sát thủ công này không thể

đáp ứng yêu cầu phát hiện kịp thời và nhanh chóng các hư hỏng trên một con đường để cơ quan quản lý có thể đưa ra các quyết định và kế hoạch kịp thời. Do đó bài toán phát hiện hư hỏng trên mặt đường bộ cần những cách tiếp cận tự động sử dụng các giải pháp Công nghệ Thông tin hiện đại.

Để giải quyết bài toán phát hiện hư hỏng trên mặt đường bộ ở Ấn Độ, Dharneshkar .J và các cộng sự [1] đã sử dụng mô hình YOLOv3-tiny [2] huấn luyện trên 1500 ảnh và đạt được độ chính xác khá khiêm tốn (mAP@0.5 là 49,71%). Mô hình YOLOv4 [3] với kiến trúc Darknet đã được sử dụng trong [4] và đạt được độ chính xác mAP@0.5 là 81,82%. Tuy nhiên số ảnh huấn luyện còn hạn chế (1087 ảnh). YOLOv3 cũng là một cách tiếp cận được Rohitaa R. và các cộng sự áp dụng trong [5] và đạt được tốc độ real-time (40ms/1 frame). Phiên bản tiếp theo của cách tiếp cận YOLO là YOLOv5 [6] tiếp tục được áp dụng và cũng có kết quả khá tốt (mAP@0.50 = 90% và FPS = 31,76) [7]. Một cách tiếp cận khác là sử dụng các phương pháp phân lớp hình ảnh (mô hình VGG19 [8]) đã được các tác giả sử dụng trong [9] trên 1157 ảnh chụp vệ tinh và đạt được độ chính xác khoảng 96%. Srikanth H. và các cộng sự áp dụng mô hình YOLOv5 trên khoảng 1100 ảnh huấn luyện và đạt được độ chính xác khoảng 86,4%. Có thể thấy điểm chung của một số cách tiếp cận đã được đề cập ở trên là sử dụng các mô hình phát hiện đối tượng (chủ yếu là họ các mô hình YOLO) và các mạng phân lớp học sâu trên các tập cơ sở dữ liệu hình ảnh còn khá hạn chế (nhiều nhất là 1500 ảnh). Điều này dẫn tới độ chính xác đạt được còn chưa thực sự tin cậy.

Gần đây, phiên bản tiếp theo của các mô hình YOLO là YOLOv8 [10] đã được đưa ra và đạt được độ chính xác và tốc độ thực hiện cao hơn so với các mô hình YOLO trước đó. Trong bài báo này, chúng tôi đề xuất áp dụng mô hình phát hiện đối tượng YOLOv8 trên tập dữ liệu hình ảnh có số lượng ảnh lớn, kết hợp với các kỹ thuật tăng cường dữ liệu hình ảnh để giải quyết bài toán phát hiện các hư hỏng trên mặt đường bộ. Bố cục của bài báo này sẽ được tổ chức như sau: phần 2 sẽ trình bày về mô hình kiến trúc của mạng YOLOv8, phần 3 là áp dụng mô hình YOLOv8 để huấn luyện mô hình phát hiện hư hỏng trên mặt đường bộ và phần cuối cùng là Kết luận.

## 2. Mạng YOLOv8

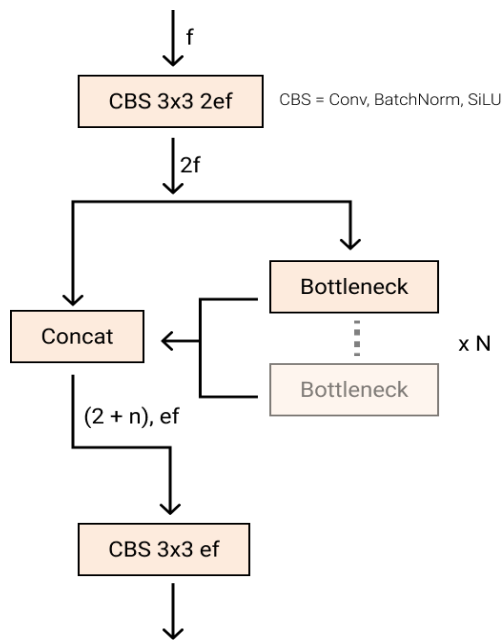
Kể từ phiên bản đầu tiên [11], các phiên bản mới của mô hình YOLO (tên gọi YOLO xuất phát từ các chữ cái đầu của cụm từ **You Only Look Once**, tạm dịch là bạn chỉ nhìn 1 lần), một mô hình phát hiện đối

tượng bằng 1 mô hình mạng học sâu duy nhất, đã liên tục được đưa ra và ngày càng có độ chính xác cao hơn, tốc độ nhanh hơn. Các phiên bản sau đều kế thừa kiến trúc của phiên bản trước và bổ sung thêm các cải tiến về mặt thuật toán và kiến trúc. Kiến trúc của mô hình mạng YOLOv8 (xem chi tiết trong hình vẽ tại địa chỉ <https://github.com/ultralytics/ultralytics/issues/189>) cũng được xây dựng theo nguyên lý đó và chủ yếu dựa trên mô hình YOLOv5 [6]. YOLOv8 sử dụng một mô hình mạng CNNs (Convolutional Neural Networks) [12] đơn được chia thành hai phần: Phần xương sống (backbone) và phần đầu (head). Phần xương sống của YOLOv8 là một phiên bản sửa đổi của kiến trúc CPSPDarknet53 có 53 lớp nhân chập và các kết nối một phần giữa các lớp để cải thiện luồng thông tin truyền giữa các lớp khác nhau. Phần đầu của mạng YOLOv8 bao gồm một loạt các lớp nhân chập và cuối cùng là một loạt các lớp kết nối đầy đủ có nhiệm vụ dự đoán các bounding boxes, độ tin cậy của các đối tượng và lớp mà nó cần dự đoán. Một đặc trưng quan trọng khác của mô hình YOLOv8 là sử dụng cơ chế phát hiện đối tượng multi-scale bằng cách áp dụng một mạng đặc trưng kiểu kim tự tháp cho phép nó phát hiện các đối tượng có kích thước và tỉ lệ khác nhau trong một bức ảnh.

So với phiên bản YOLOv5, YOLOv8 có 3 điểm khác biệt lớn là: Cơ chế phát hiện không phụ thuộc vào các điểm neo (anchor free detection), các phép nhân chập mới, cơ chế dừng kỹ thuật tăng cường dữ liệu. Chúng ta sẽ làm rõ các điểm mới mới này ở dưới đây.

**Cơ chế phát hiện đối tượng không phụ thuộc các điểm neo:** Cơ chế này cho phép YOLOv8 có thể thực hiện các dự đoán về các điểm trung tâm của các bounding boxes thay vì các giá trị khoảng cách (offset) từ một điểm neo như các phiên bản YOLO trước đây. Cơ chế này sẽ giảm số lượng dự đoán, tăng tốc quá trình loại bỏ các dự đoán sai và giảm đáng kể thời gian tính toán cho cả mô hình.

**Phép nhân chập mới:** Khối nhân chập đầu tiên 6x6 được thay bằng phép nhân chập có nhân 3x3 và khối nhân chập chính C3 được thay bằng khối C2f (xem chi tiết trong Hình 1). Khối có tên là Cổ chai (Bottleneck) trong Hình 1 hoàn toàn giống với YOLOv5 nhưng kích thước của nhân của phép nhân chập đã đổi từ 1x1 thành 3x3. Ở phần cổ (neck) nằm giữa phần xương sống và đầu, các đặc trưng được ghép lại với nhau một cách trực tiếp mà không cần qua bước chuẩn hóa để cùng số chiều để giảm kích thước tham số cần thiết.



Hình 1. Phép nhân chập mới của YOLOv8

**Cơ chế dừng kỹ thuật tăng cường dữ liệu:** Đối với YOLOv8, việc huấn luyện mạng cũng là một yếu tố quan trọng. Quá trình huấn luyện của YOLOv8 áp dụng các thuật toán tăng cường dữ liệu một cách linh hoạt, có điều chỉnh và thay đổi trong các epoch huấn luyện liên tiếp của một lần huấn luyện và một trong các kỹ thuật hiệu quả là tăng cường dữ liệu kiểu khảm ảnh (mosaic augmentation): Ghép các ảnh với nhau để mô hình phải học việc dự đoán các vị trí mới của các đối tượng trong bức ảnh ghép. Với phiên bản YOLOv8, khoảng 10 epoch cuối sẽ không dừng áp

dụng kỹ thuật này để tránh suy thoái về độ chính xác.

**Độ chính xác của mô hình YOLOv8:** Theo tài liệu chính thức được đưa ra [10] thì YOLOv8 chính xác hơn và nhanh hơn tất cả các mô hình đã công bố trước đó.

### 3. Áp dụng YOLOv8 cho bài toán phát hiện hư hỏng trên mặt đường bộ

Nhằm mục đích xây dựng một ứng dụng có khả năng phát hiện chính xác và nhanh chóng các hư hỏng trên mặt đường bộ, các tác giả đã lựa chọn mô hình YOLOv8 vì đây là một mô hình mạng mới, có độ chính xác cao trong phát hiện đối tượng và có tốc độ nhanh so với các mô hình phát hiện đối tượng khác.

Để xây dựng mô hình phát hiện hư hỏng trên mặt đường bộ, chúng tôi đã khảo sát, đánh giá, phân tích và lựa chọn một tập dữ liệu công cộng gồm hình ảnh các hư hỏng trên mặt đường bộ có số lượng ảnh lớn, cụ thể tập huấn luyện gồm 6359 ảnh (<https://www.kaggle.com/datasets/mersico/road-damage-tracking-dataset-rdtd-v10/data>), tập validation gồm 871 ảnh có độ phân giải 1280x1280 (xem chi tiết trong Hình 2). Tập dữ liệu này là các hình ảnh đường bộ và hư hỏng trên mặt đường của các nước Ấn Độ, Nhật Bản, Trung Quốc,... được các tác giả của nó thu thập từ Internet và đánh nhãn. Các hình ảnh sau khi áp dụng các kỹ thuật tăng cường dữ liệu thì tổng số ảnh huấn luyện là 19074 ảnh. Các thuật toán xử lý ảnh khác nhau đã được áp dụng để tạo ra các ảnh mới nhằm tăng độ chính xác của mô hình huấn luyện bao gồm: Ghép ảnh, tạo nhiễu muối tiêu, xoay ngang, xoay ảnh ngẫu nhiên theo các góc trong khoảng -15 đến +15 độ, chỉnh độ sáng trong khoảng từ -25 đến +25 phần trăm, làm mờ bằng bộ lọc Gaussian.

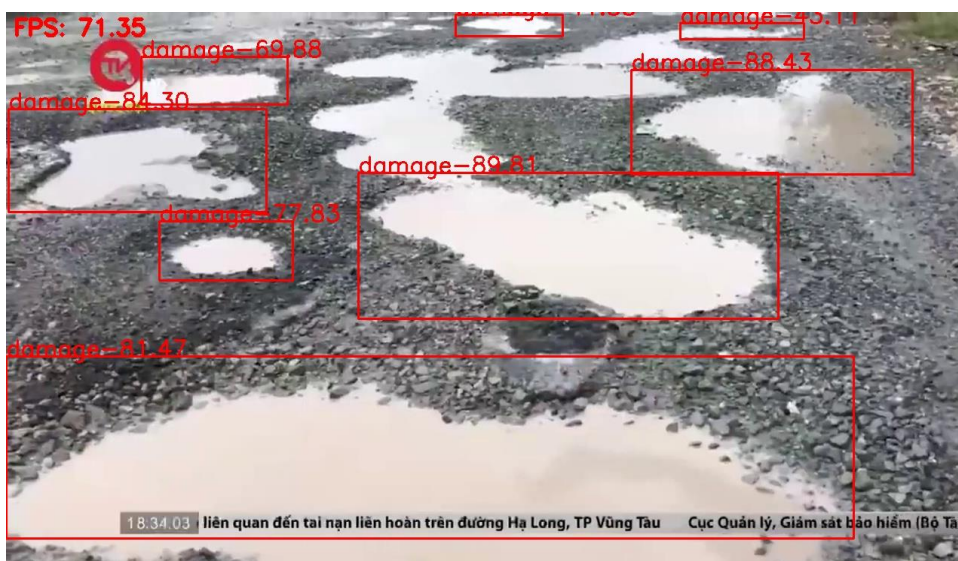


Hình 2. Ảnh hư hỏng trên mặt đường bộ trong tập huấn luyện



**Bảng 1. Kết quả huấn luyện mô hình YOLOv8 phát hiện hư hỏng trên mặt đường bộ**

TT	Mô hình	Độ chính xác trên tập huấn luyện (%)		Độ chính xác trên tập validation (%)	
		mAP@0.50	mAP@0.50-0.95	mAP@0.50	mAP@0.5-0.95
1	YOLOv8n	82.9	52.5	70.9	36.2
2	YOLOv8s	85.9	58.1	80.4	42.8
3	YOLOv8m	87.4	60.6	81.7	44.8
4	YOLOv8l	87.4	61.6	82.1	45.4
5	YOLOv8x	87.5	62.4	82.7	46.6



**Hình 3. Kết quả phát hiện hư hỏng trên mặt đường bộ với dữ liệu test thực tế**

YOLOv8 có 5 phiên bản khác nhau là Nano (n), Small (s), Medium (m), Large (l) và Extra Large (x). Các phiên bản này khác nhau về độ phức tạp (số tham số), kích thước mô hình, độ chính xác và thời gian thực hiện trong đó mô hình Nano là đơn giản nhất, có kích thước nhỏ nhất, độ chính xác thấp nhất, tốc độ thực hiện nhanh nhất còn mô hình Extra Large phức tạp nhất, kích thước lớn nhất, độ chính xác cao nhất và thời gian thực hiện lâu nhất. Trong bài báo này, để đảm bảo các kết quả huấn luyện có độ tin cậy và các mô hình sau huấn luyện có độ chính xác cao chúng tôi đã thực hiện huấn luyện cả 5 mô hình trên một máy trạm cài đặt thư viện Pytorch 2.2.1 với phiên bản CUDA 11.8 được trang bị 1 card GPU RTX 3060 12 GB RAM với kích thước ảnh huấn luyện là 640x640 với 200 epochs. Kết quả huấn luyện được thể hiện trong Bảng 1 cho thấy cách tiếp cận sử dụng mạng YOLOv8 là đúng đắn và hiệu quả khi kết quả độ chính xác mAP@0.5 của cả 5 phiên bản YOLOv8 đều khá cao: Thấp nhất là 82,9% với mô hình YOLOv8n và cao nhất là 87,5% với mô hình YOLOv8x. Các kết quả này đối với giá trị đánh giá mAP@0.5-0.95 trên tập

huấn luyện cũng rất hứa hẹn khi đạt được 52,5% với mô hình Nano và tăng dần tới 62,4% với mô hình Extra Large. Tuy nhiên trên tập Validation thì độ chính xác này cũng bị suy giảm đáng kể: Chỉ còn từ 70,9% đến 82,7% với mAP@0.5 và tương ứng là 36,2% tới 46,6% với mAP@0.5-0.95.

Để kiểm thử mô hình đã huấn luyện, chúng tôi đã thu thập 5 video test từ thực tế từ Internet và qua camera giám sát hành trình của 1 xe ô tô và kết quả cho thấy các mô hình đã phát hiện tương đối tốt các hư hỏng trên mặt đường bộ của các video này (xem chi tiết trong Hình 3). Tốc độ phát hiện cũng đáp ứng khả năng triển khai thực tế khi đạt tới 71,35 frame ảnh trong 1 giây (mô hình Small) trên 1 laptop có trang bị card RTX 3070 (xem chi tiết trong Hình 3). Nhóm tác giả cũng đã chuyển định dạng của mô hình huấn luyện từ định dạng file của thư viện Pytorch sang định dạng để triển khai thực tế là TensorRT và thử nghiệm và đạt được tốc độ lên đến 110 frame ảnh trong 1 giây. Điều này cho thấy khả năng ứng dụng vào thực tế của cách tiếp cận này là hoàn toàn khả thi về mặt tốc độ.

#### 4. Kết luận

Để giải quyết bài toán phát hiện hư hỏng trên mặt đường bộ, trong bài báo này chúng tôi đã đề xuất áp dụng mô hình mạng học sâu YOLOv8. Các phiên bản khác nhau của YOLOv8 đã được huấn luyện trên một tập dữ liệu hình ảnh cùng với các kỹ thuật tăng cường dữ liệu. Kết quả huấn luyện và thử nghiệm trên dữ liệu thực tế (độ chính xác và tốc độ) cho thấy hướng tiếp cận sử dụng YOLOv8 cho bài toán phát hiện hư hỏng trên mặt đường bộ là một hướng tiếp cận đúng đắn và có thể đưa vào áp dụng trên thực tế. Theo thông tin khảo sát của nhóm tác giả, đây là một trong số bài báo đầu tiên ở Việt Nam giải quyết bài toán phát hiện hư hỏng trên mặt đường bộ theo cách tiếp cận này. Bên cạnh đó, các kết quả trong bài báo cũng đã được thử nghiệm với 1 tập dữ liệu khá lớn (so với các công trình đã đề cập trong phần đầu của bài báo) và thử nghiệm với dữ liệu thực tế, mô hình sau khi huấn luyện cũng đã được chuyển thành định dạng để có thể triển khai thực tế với độ chính xác khá cao và tốc độ theo thời gian thực. Tương lai chúng tôi sẽ hướng tới việc thu thập thêm dữ liệu thực tế để huấn luyện, thử nghiệm và triển khai trên các thiết bị biên để nâng cao độ chính xác trong việc nhận dạng và hướng tới ứng dụng vào cuộc sống.

#### Lời cảm ơn

Nghiên cứu này được tài trợ bởi Trường Đại học Hàng hải Việt Nam trong đề tài mã số: **DT23-24.64**.

#### TÀI LIỆU THAM KHẢO

- [1] D. J, S. D. V, A. S A, K. R, and L. Parameswaran (2020), *Deep Learning based Detection of potholes in Indian roads using YOLO*, in 2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India: IEEE, Feb. 2020, pp.381-385.  
doi: 10.1109/ICICT48043.2020.9112424.
- [2] J. Redmon and A. Farhadi (2018), *YOLOv3: An Incremental Improvement*. arXiv, Apr. 08.  
doi: 10.48550/arXiv.1804.02767.
- [3] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao (2020), *YOLOv4: Optimal Speed and Accuracy of Object Detection*. arXiv, Apr. 22.  
doi: 10.48550/arXiv.2004.10934.
- [4] A. Al-Shaghouri, R. Alkhatib, and S. Berjaoui (2021), *Real-Time Pothole Detection Using Deep Learning*.
- [5] R. R, S. Shreya, and A. R (2021), *Intelligent Deep Learning based Pothole Detection and Reporting System*, in 2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT), Erode, India: IEEE, pp.1-5.  
doi: 10.1109/ICECCT52121.2021.9616703.
- [6] Ultralytics (2021), *YOLOv5: A state-of-the-art real-time object detection system*.  
[Online]. Available: <https://docs.ultralytics.com>
- [7] M. H. Asad, S. Khaliq, M. H. Yousaf, M. O. Ullah, and A. Ahmad (2022), *Pothole Detection Using Deep Learning: A Real-Time and AI-on-the-Edge Perspective*, Adv. Civ. Eng., Vol.2022, pp.1-13.  
doi: 10.1155/2022/9221211.
- [8] K. Simonyan and A. Zisserman (2015), *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv, Apr. 10.  
doi: 10.48550/arXiv.1409.1556.
- [9] C. Saisree and Dr. K. U (2023), *Pothole Detection Using Deep Learning Classification Method*, Procedia Comput. Sci., Vol.218, pp.2143-2152.  
doi: 10.1016/j.procs.2023.01.190.
- [10] J. Glenn, C. Ayush, and Q. Jing (2023), *Ultralytics YOLOv8*.  
[Online]. Available: <https://github.com/ultralytics/ultralytics>
- [11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi (2016), *You Only Look Once: Unified, Real-Time Object Detection*. arXiv.  
doi: 10.48550/arXiv.1506.02640.
- [12] A. Krizhevsky, I. Sutskever, and G. E. Hinton (2012), *ImageNet classification with deep convolutional neural networks*, in Proceedings of the 25th International Conference on Neural Information Processing Systems - Vol. 1, in NIPS'12. Red Hook, NY, USA: Curran Associates Inc., pp.1097-1105.

Ngày nhận bài:	08/03/2024
Ngày nhận bản sửa lần 01:	18/03/2024
Ngày nhận bản sửa lần 02:	02/04/2024
Ngày duyệt đăng:	13/04/2024