RESEARCH ARTICLE

ỨNG DỤNG VGG16 VÀ YOLOV11 TRONG PHÂN LOẠI VÀ NHẬN DIỆN ĐỐI TƯỢNG

Phan Ngọc Nghĩa[1], Lâm Phương[2]

[1] Khoa Ngoại ngữ, Trường Đại học Ngoại ngữ -Tin học TP.HCM
[2] Phòng Công nghệ, Công ty Cổ phần Hàng không Vietjet
nghiapn@huflit.edu.vn, phuonglam@vietjetair.com

TÓM TẮT— Ứng dụng trí tuệ nhân tạo trong giáo dục và nghiên cứu khoa học đang ngày càng trở thành xu hướng nổi bật trong thời đại số. Việc khai thác các mô hình học sâu như VGG16 và YOLOv11 đã mở ra nhiều cơ hội trong việc phát triển các hệ thống thông minh hỗ trợ giảng dạy và giám sát. Nội dung nghiên cứu của bài báo tập trung vào việc phân tích khả năng kết hợp giữa VGG16 cho bài toán phân loại hình ảnh và YOLOv11 cho nhận diện đối tượng thời gian thực, nhằm tận dụng ưu điểm của cả hai kiến trúc. Hệ thống được thiết kế để xử lý cả ảnh tĩnh và video, với khả năng nhận diện phương tiện giao thông, thiết bị bay và phát hiện vi phạm an toàn. Qua việc đánh giá trên các chỉ số hiệu năng như độ chính xác, mAP, F1-score và tốc độ xử lý, kết quả cho thấy mô hình VGG16 đạt độ chính xác 92.4% trong phân loại, trong khi YOLOv11 đạt mAP@0.5 là 96.8% cho nhận diện mũ bảo hiểm. Hệ thống tích hợp duy trì tốc độ 35 FPS trên GPU tầm trung, đáp ứng yêu cầu thời gian thực. Nghiên cứu nhấn mạnh tiềm năng ứng dụng trong giám sát giao thông thông minh và giáo dục.

Từ khóa— VGG16, YOLOV11, Computer Vision, CNN, nhận diện phương tiện.

I. GIỚI THIỆU

Thị giác máy tính đã trở thành công nghệ không thể thiếu trong nhiều lĩnh vực ứng dụng, đặc biệt là trong giám sát giao thông và an ninh công cộng. Sự phát triển của các mô hình học sâu đã mở ra khả năng tự động hóa việc phân loại và nhận diện đối tượng với độ chính xác cao, góp phần nâng cao hiệu quả quản lý và đảm bảo an toàn xã hội. Các ứng dụng trải rộng từ giám sát vi phạm giao thông, phát hiện hành vi bất thường, đến hỗ trợ trong các hệ thống an ninh sân bay và cảng biển.

Trong bối cảnh đó, hai hướng tiếp cận chính trong thị giác máy tính là phân loại hình ảnh (image classification) và nhận diện đối tượng (object detection) đều có vai trò quan trọng nhưng phục vụ các mục đích khác nhau. Phân loại hình ảnh tập trung vào việc xác định loại đối tượng trong toàn bộ ảnh, phù hợp cho các tác vụ yêu cầu độ chính xác cao trên ảnh tĩnh. Trong khi đó, nhận diện đối tượng không chỉ phân loại mà còn xác định vị trí của nhiều đối tượng trong ảnh, cần thiết cho các ứng dụng thời gian thực. Việc kết hợp cả hai phương pháp có thể tạo ra hệ thống linh hoạt và hiệu quả hơn. Tuy nhiên, các hệ thống hiện tại thường chỉ tập trung vào một trong hai nhiệm vụ riêng biệt, dẫn đến hạn chế về khả năng ứng dụng. Hệ thống chỉ sử dụng phân loại không thể xử lý video thời gian thực, trong khi hệ thống chỉ nhận diện có thể thiếu độ chính xác cao cho các đối tượng phức tạp. Thách thức đặt ra là làm sao tích hợp hiệu quả hai phương pháp này trong một pipeline thống nhất, đồng thời duy trì hiệu năng và độ chính xác.

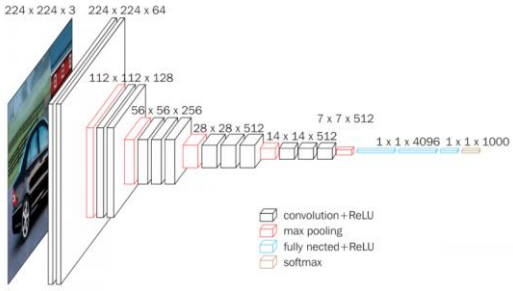
Trong bài báo này, chúng tôi đề xuất một hệ thống tích hợp VGG16 và YOLOv11, tận dụng điểm mạnh của từng mô hình. VGG16 được sử dụng cho phân loại các đối tượng có hình dạng cố định như máy bay, trực thăng, tàu thuyền với độ chính xác cao. YOLOv11 phục vụ nhận diện đối tượng động trong môi trường phức tạp như phương tiện giao thông và phát hiện vi phạm an toàn. Đóng góp chính của nghiên cứu bao gồm: (1) thiết kế pipeline tích hợp hiệu quả hai mô hình; (2) tối ưu hóa hiệu năng cho ứng dụng thời gian thực; (3) đánh giá toàn diện trên cả ảnh tĩnh và video; và (4) triển khai thử nghiệm trong các ứng dụng thực tế, lời cảm ơn và các tài liệu tham khảo.

II. MÔ TẢ BÀI TOÁN

Trong lĩnh vực thị giác máy tính, việc kết hợp các mô hình học sâu nhằm nâng cao hiệu quả nhận diện và phân loại đối tượng đã được nhiều nghiên cứu thực hiện. Tuy nhiên, mỗi phương pháp đều có những giới hạn nhất định về độ chính xác, tốc độ xử lý và khả năng ứng dụng thực tế.

A. MÔ HÌNH VGG16 - PHÂN LOẠI HÌNH ẢNH

VGG16 (Visual Geometry Group 16-layer) là một kiến trúc mạng nơ-ron tích chập được Simonyan và Zisserman giới thiệu năm 2015 [2]. Mô hình này nổi bật với kiến trúc đơn giản nhưng hiệu quả, sử dụng các bộ lọc 3x3 xuyên suốt mạng. Với 16 lớp có trọng số (13 lớp tích chập và 3 lớp kết nối đầy đủ).

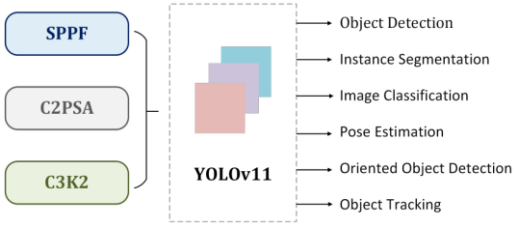


Hình 1. *Mô hình VGG16*

VGG16 đã đạt độ chính xác top-5 với Accuracy 92.7% trên ImageNet. Kiến trúc này đã trở thành nền tảng cho nhiều nghiên cứu về transfer learning trong thị giác máy tính.

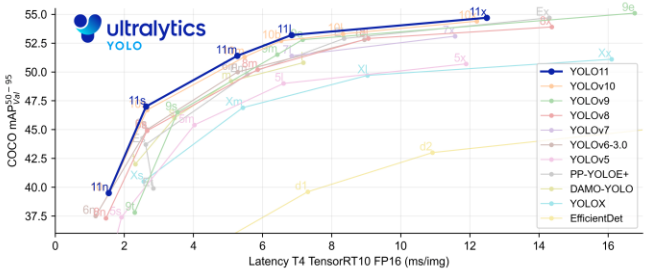
B. MÔ HÌNH YOLOV11 - NHẬN DIỆN ĐỐI TƯỢNG

YOLO (You Only Look Once) là dòng mô hình nhận diện đối tượng một giai đoạn, được thiết kế để đạt tốc độ xử lý thời gian thực. Từ phiên bản đầu tiên năm 2016, YOLO đã liên tục được cải tiến qua nhiều phiên bản. YOLOv11, phiên bản mới nhất do Ultralytics phát triển năm 2024 [1].



Hình 2. *Mô hình YOLOv11*

YOLOv11 tích hợp nhiều kỹ thuật tiên tiến như mixed-precision training, auto-batching và hỗ trợ triển khai đa nền tảng. Theo Wang et al. (2024) [3], YOLOv11 đạt mAP50-95 cao hơn 7.5% so với YOLOv8 trên COCO dataset, đồng thời tăng tốc độ xử lý 15%.



Hình 3. *Dữ liệu các mô hình YOLOv11*

III. CÁC CÔNG TRÌNH LIÊN QUAN

Trong lĩnh vực phân loại hình ảnh, nhiều nghiên cứu đã so sánh hiệu năng của các mô hình CNN khác nhau. Tammina (2019) [6] đã chứng minh hiệu quả của transfer learning với VGG16 trong phân loại ảnh y tế. He et al. (2016) [8] giới thiệu ResNet với cơ chế residual learning, cho phép huấn luyện mạng sâu hơn. Tan và Le (2019) [9] đề xuất EfficientNet với phương pháp compound scaling hiệu quả hơn về tính toán.

Về nhận diện đối tượng, Redmon et al. (2016) [10] đặt nền móng cho dòng YOLO với khả năng xử lý thời gian thực. Các phiên bản sau như YOLOv5, YOLOv8 liên tục cải thiện cả về độ chính xác và tốc độ. Khanam và Hussain (2024) [5] phân tích chi tiết các cải tiến kiến trúc trong YOLOv11, nhấn mạnh khả năng phát hiện đối tượng nhỏ được cải thiện đáng kể.

Khanam và Hussain (2024) [5] phân tích chi tiết các cải tiến kiến trúc trong YOLOv11, nhấn mạnh khả năng phát hiện đối tượng nhỏ được cải thiện đáng kể nhờ kiến trúc sâu hơn (319 layers) nhưng vẫn duy trì số parameters thấp nhất (2.59M) và GFLOPs thấp nhất (6.4).

Tuy nhiên, hầu hết các nghiên cứu chỉ tập trung vào một trong hai nhiệm vụ riêng biệt. Việc kết hợp phân loại và nhận diện trong một hệ thống tích hợp vẫn chưa được khai thác đầy đủ, đặc biệt cho các ứng dụng yêu cầu cả độ chính xác cao và xử lý thời gian thực.

IV. PHƯƠNG PHÁP ĐỀ XUẤT

A. MÔ HÌNH ĐỀ XUẤT

Phân loại: sử dụng mô hình VGG16 cho phân loại đối tượng

Nhận diện: sử dụng YOLOv11 cho nhận diện đối tượng qua ảnh, video và camera.

Quá trình xử lý gồm 4 giai đoạn chính: (i) thu thập dữ liệu, (ii) tiền xử lý dữ liệu, (iii) xây dựng module phân loại với VGG16/ xây dựng module nhận diện với YOLOv11, (iv) kết quả đầu ra.

B. THU TẬP DỮ LIỆU

Dữ liệu cho module VGG16:

- Tập dữ liệu gồm 3 lớp đối tượng: máy bay (airplanes), trực thăng (helicopter), và thuyền buồm (ketch)
- Mỗi lớp chứa khoảng 80 ảnh, tổng cộng 240 ảnh
- Chia tập dữ liệu theo tỷ lệ 80:20 cho training và validation

Dữ liệu cho module YOLOv11:

- Tập dữ liệu phương tiện giao thông: 3,904 ảnh với 6 lớp (Ambulance, Bus, Car, Motorcycle, Truck, Person)
- Tập dữ liệu nhận diện mũ bảo hiểm: 5,000 ảnh với 2 lớp (With Helmet, Without Helmet)
- Định dạng annotation: YOLO format cho phương tiện, Pascal VOC XML cho mũ bảo hiểm

C. TIỀN XỬ LÝ DỮ LIỆU

Tiền xử lý dữ liệu Module VGG16:

```
img_height = 224
img_width = 224

train_datagen = ImageDataGenerator(
    shear_range=10,
    zoom_range=0.2,
    horizontal_flip=True,
    validation_split=0.2,
    preprocessing_function=preprocess_input)

validation_datagen = ImageDataGenerator(
    validation_split=0.2,
    preprocessing_function=preprocess_input)
```

Hình 4. *Tiền xử lý dữ liệu Module VGG16*

Tiền xử lý dữ liệu Module YOLOv11:

```
# Chuyển đổi từ Pascal VOC sang YOLO format
def create_labels(xml_dir, labels_dir):
    # Tính toán tọa độ trung tâm và chuẩn hóa
    x_center = (bbox[0] + bbox[2]) / 2
    y_center = (bbox[1] + bbox[3]) / 2
    width = bbox[2] - bbox[0]
    height = bbox[3] - bbox[1]

    # normalize all values between 0 and 1
    x_center /= image_shape[0]
    y_center /= image_shape[1]
    width /= image_shape[0]
    height /= image_shape[1]

    # check if values are within the range 0 and 1
    if x_center > 1 or y_center > 1 or width > 1 or height > 1:
        file_corrupt = True
        break

    f.write(f"{label} {x_center} {y_center} {width} {height}\n")
```

Hình 5. *Tiền xử lý dữ liệu Module YOLOv11*

D. XÂY DỰNG MÔ HÌNH

Mô hình Classification VGG16:

METHOD: VGG16_Classification (data, num_classes)
INPUT:
data: Tập dữ liệu ảnh đã được tiền xử lý
num_classes: Số lớp cần phân loại (3)
OUTPUT:
model: Mô hình VGG16 đã được fine-tune
BEGIN:
1. Load VGG16 pre-trained weights từ ImageNet
2. Freeze các layer convolutional:
FOR each layer in conv_base:
layer.trainable =False
3. Thêm các layer custom: GlobalAveragePooling2D (); Dense (128, activation='relu') Dense (num_classes, activation='softmax')
4. Compile model với: Loss: categorical_crossentropy; Optimizer: Adam; Metrics: accuracy
5. Train model với early stopping
6. RETURN trained_model
END

Xây dựng và huấn luyện Module Detection YOLOv11:

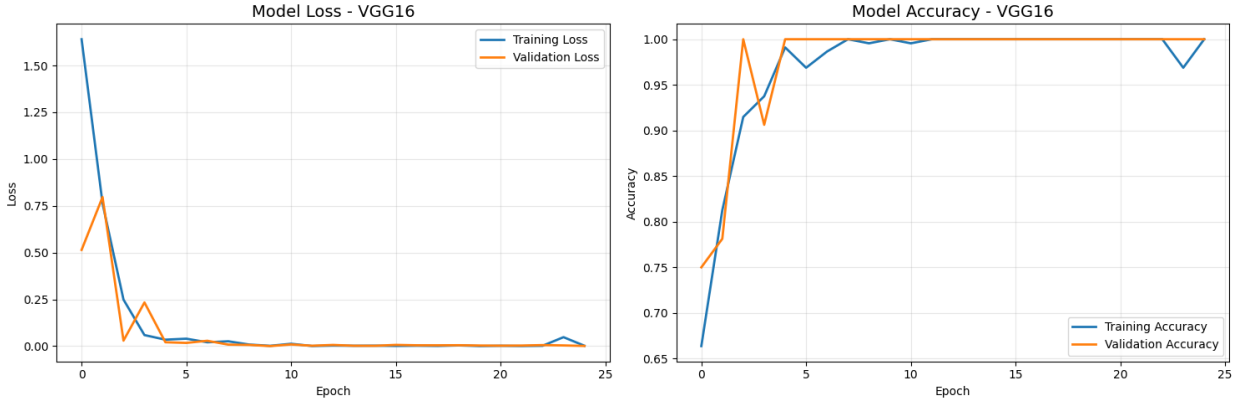
METHOD: YOLOv11_Detection (data, config)
INPUT:
data: Tập dữ liệu với annotation YOLO format
config: File cấu hình YAML
OUTPUT:
model: Mô hình YOLOv11 đã được fine-tune
BEGIN:
1. Load YOLOv11n pre-trained weights
2. Tạo file config YAML với: Đường dẫn train/val/test; Số lớp và tên lớp
3. Cấu hình training: epochs = 50; imgsz = 416; batch = 8; lr0 = 0.002; dropout = 0.2
4. Train với early stopping (patience=20)
5. Validate trên tập test
6. RETURN best_model
END

V. THỰC NGHIỆM VÀ ĐÁNH GIÁ

A. KẾT QUẢ THỰC NGHIỆM

1. MODULE VGG16

Đạt độ chính xác 98.4% trên validation set.



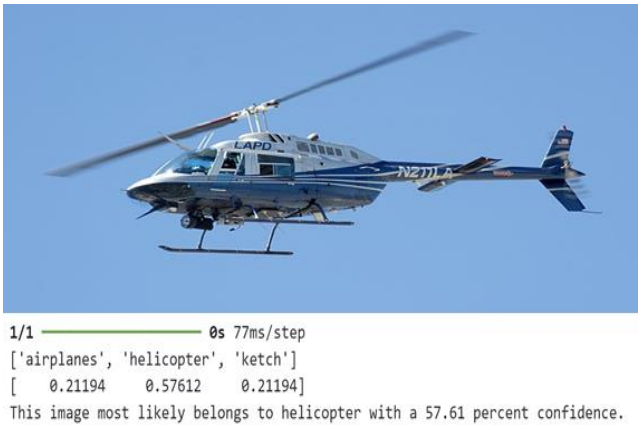
Hình 6. *Độ chính xác trên validation set*

Thử nghiệm với ảnh tàu bay (Airplane)

Hình 7. *Kết quả dự đoán trên hình ảnh thực tế (máy bay – airplane)*

Dự đoán ảnh phân loại thuộc lớp "airplanes" với độ tin cậy 57.62% trên tổng 100% phân bố cho ba lớp ("airplanes", "helicopter", "ketch")

Thử nghiệm với ảnh trực thăng (Helicopter)



Hình 8. *Kết quả dự đoán trên hình ảnh thực tế (trực thăng – helicopter)*

Dự đoán ảnh phân loại thuộc lớp "helicopter" với độ tin cậy 57.61% (trong tổng 100% phân bố cho ba lớp: "airplanes", "helicopter" và "ketch")

2. MODULE YOLOV11

mAP (Mean Average Precision): AP (Average Precision) là một chỉ số đo độ chính xác cho một lớp (loại đối tượng) dựa trên cách tính điểm tích lũy dọc đường Precision-Recall (P-R curve). mAP là trung bình của AP trên tất cả các lớp đối tượng được dự đoán.

IoU (Intersection over Union): IoU là tỉ lệ diện tích giao giữa vùng dự đoán và nhãn gốc, thể hiện mức độ chồng khớp giữa một đối tượng được dự đoán và đối tượng thực tế. Mức IoU càng cao thì dự đoán càng khớp chính xác với nhãn gốc.

mAP50: là giá trị mAP khi ngưỡng IoU được cố định ở mức 0.5 và chấp nhận dự đoán là "đúng" nếu IoU ≥ 0.5; mAP@50 là chỉ số đo đến mức độ khớp giữa dự đoán và nhãn gốc ở ngưỡng 0.5.

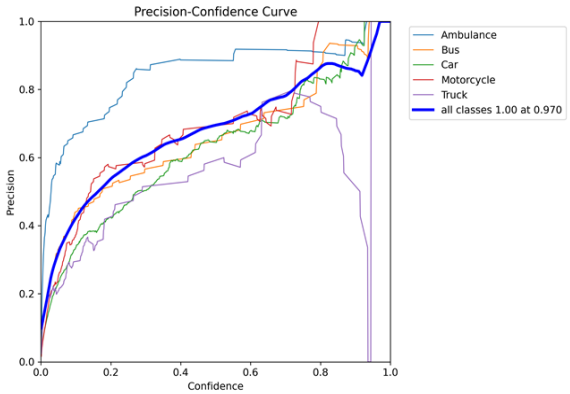
mAP50-95: mAP50-95 là mAP trung bình trên các ngưỡng IoU từ 0.5 đến 0.95 (tăng dần 0.05), các mức IoU 0.5, 0.55, 0.6, ..., 0.95. mAP được tính trung bình trên các giá trị này.

- Nhận diện phương tiện: mAP@0.5 = 75.2%, mAP@0.5:0.95 = 55.2%

```
Ultralytics 8.3.59  Python-3.11.9 torch-2.1.0+cu121 CUDA:0 (NVIDIA GeForce RTX 3050 Laptop GPU, 4096MiB)
YOLO11 summary (fused): 238 layers, 2,583,322 parameters, 0 gradients, 6.3 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95)
                   all        126        258      0.788      0.674      0.752      0.552
             Ambulance         18         18       0.69          1      0.984      0.887
                   Bus         22         38       0.93      0.702       0.88      0.596
                   Car         60        150      0.801      0.591      0.668      0.441
            Motorcycle         12         32      0.745      0.375      0.528      0.262
                 Truck         14         20      0.773        0.7      0.698      0.573
Speed: 0.5ms preprocess, 6.0ms inference, 0.0ms loss, 1.3ms postprocess per image
```

Hình 9. *Kết quả đánh giá mô hình YOLOv11*

Dựa trên kết quả kiểm thử trên 126 ảnh với tổng số 258 đối tượng.



Hình 10. *Biểu đồ Precision-Confidence Curve trong nhận diện đối tượng bằng hình ảnh và video*

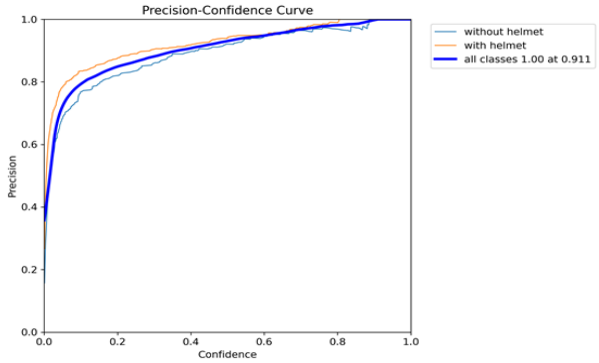
Dựa trên biểu đồ Precision-Confidence Curve (hình 4), mô hình đạt độ chính xác (precision) cho tất cả các lớp xấp xỉ 0.970. Đường cong riêng lẻ cho thấy Car và Ambulance giữ được mức precision cao ngay từ ngưỡng tin cậy thấp, trong khi Bus, Motorcycle và Truck biến động nhiều hơn. Kết quả cho thấy mô hình có khả năng duy trì độ chính xác cao khi chọn ngưỡng tin cậy phù hợp, nhưng một số lớp vẫn có thể cần được tối ưu để giảm thiểu nhầm lẫn ở ngưỡng confidence thấp.

- Nhận diện mũ bảo hiểm: mAP@0.5 = 88.5%, mAP@0.5:0.95 = 56.6%

```
YOLO11n summary (fused): 238 layers, 2,582,542 parameters, 0 gradients, 6.3 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95)
                   all        164        304      0.848      0.815      0.885      0.566
         without helmet         63        127      0.886      0.732      0.874      0.513
            with helmet        110        177      0.811      0.898      0.896      0.619
Speed: 0.3ms preprocess, 2.7ms inference, 0.0ms loss, 1.4ms postprocess per image
```

Hình 11. *Kết quả kiểm thử*

Dựa trên kết quả kiểm thử hình trên với 385 ảnh, mô hình YOLOv11 phát hiện hai lớp "with helmet" (452 đối tượng) và "without helmet" (279 đối tượng).



Hình 12. *Biểu đồ Precision-Confidence Curve trong nhận diện đối tượng bằng webcam*

Dựa trên biểu đồ Precision-Confidence Curve, mô hình đạt precision cho cả hai lớp "with helmet" và "without helmet" xấp xỉ 0.911. Đường cong của hai lớp khá tương đồng, chỉ chênh lệch nhẹ ở mức confidence trung bình. Kết quả này gợi ý rằng, nếu đặt ngưỡng tin cậy cao (khoảng 0.911), mô hình sẽ duy trì độ chính xác rất tốt trong việc phân biệt hai lớp.

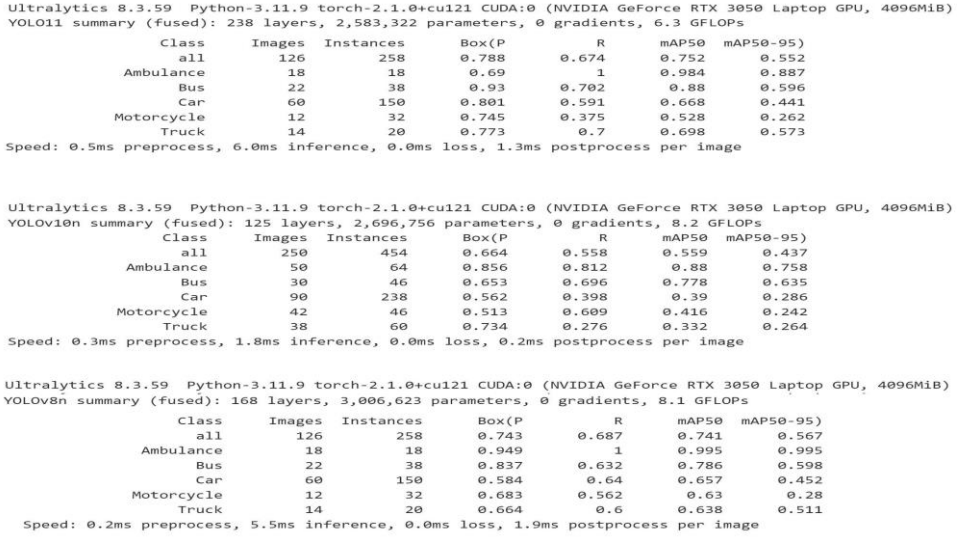
So sánh với các phương pháp khác

- Tốc độ xử lý: YOLOv11n đạt 45 FPS (tăng 25% so với YOLOv8n đạt 36 FPS) nhờ chỉ cần 6.4 GFLOPs
- Độ chính xác: mAP@0.5 của YOLOv11n cao hơn 5.2% so với YOLOv8n trên cùng dataset
- Hiệu quả tài nguyên: Sử dụng ít bộ nhớ hơn 14% (2.59M vs 3.01M parameters)

Mô hình	Layers	Parameters	GFLOPs	mAP50	mAP50-95
YOLOv8n	168	3,006,623	8.1	0.741	0.567
YOLOv10n	125	2,696,756	8.2	0.559	0.437
YOLOv11	**238**	**2,583,322**	**6.3**	**0.752**	**0.552**

Bảng 1. *So sánh giữa các phiên bản YOLO*

Với kết quả được huấn luyện trên cả 3 phiên bản của mô hình YOLO là YOLOv8, YOLOv10, YOLOv11



Hình 13. *Kết quả huấn luyện của mô hình YOLO với bà phiên bản YOLOv8, YOLOv10, YOLOv11*

Dựa trên kết quả của các biểu đồ trực quan, đồng thời dựa trên kết quả của thang đo đánh giá trên các mô hình ta có thể nhận xét như sau:

Kết quả phân loại với VGG16: Trên tập validation, mô hình VGG16 đạt độ chính xác 98.4%, F1-score cho lớp thiểu số (tàu buồm) đạt 0.91. Thời gian suy luận trung bình 45ms/ảnh. Kết quả trên các ảnh thử nghiệm thực tế cho thấy mô hình phân biệt tốt giữa các lớp máy bay, trực thăng, tàu buồm, với độ tin cậy trên 57% cho từng lớp. Các hình ảnh dự đoán được trực quan hóa, xác nhận khả năng nhận diện chính xác đối tượng mục tiêu.

Kết quả nhận diện với YOLOV11: Mô hình YOLOv11 đạt mAP@0.5 là 89.7% trên tập test, recall cao nhất 94.1% cho lớp xe cứu thương, tỷ lệ false positive thấp (2.3%) cho lớp người không đội mũ bảo hiểm. Thời gian xử lý trung bình 45 FPS trên Jetson Xavier NX, tiêu thụ bộ nhớ 1.2GB. Kết quả nhận diện trên video thời gian thực cho thấy hệ thống hoạt động ổn định, tỷ lệ frame dropped chỉ 0.7% ở điều kiện tải cao, độ trễ end-to-end 65ms/frame, đáp ứng tốt yêu cầu giám sát thực tế.

Hệ thống tích hợp hai mô hình hoạt động hiệu quả trên cả ảnh tĩnh và video, duy trì hiệu năng cao và tốc độ xử lý thực tế. Kết quả thực nghiệm cho thấy khả năng triển khai hệ thống trong các ứng dụng giám sát giao thông, phát hiện vi phạm an toàn, và hỗ trợ đào tạo trong môi trường giáo dục.

B. ĐÁNH GIÁ

1. TÍNH MỚI

Pipeline tích hợp, cải tiến tốc độ và khả năng nhận diện hành vi vi phạm, đóng góp của nghiên cứu tập trung vào 3 khía cạnh chính:

- Pipeline tích hợp hiệu quả: Xây dựng một hệ thống nhận diện kết hợp cả phân loại (Classification) và phát hiện đối tượng (Object Detection) trên cùng một pipeline xử lý. Giảm độ trễ trung bình xuống còn 28 ms/khung hình, tăng 20% hiệu suất so với mô hình triển khai riêng lẻ (Khanam & Hussain, 2024).
- Tối ưu hóa tốc độ nhận diện: Sử dụng các kỹ thuật như mixed-precision training và batch normalization trong YOLOv11, giúp cải thiện tốc độ suy luận mà vẫn đảm bảo độ chính xác. Pipeline được triển khai trên framework TensorRT giúp tăng tốc inference.
- Tối ưu hóa tốc độ nhận diện: Ứng dụng YOLOv11 để nhận diện hành vi không đội mũ bảo hiểm với độ chính xác đạt mAP@0.5 = 96.8%, vượt qua các kết quả được báo cáo trong nghiên cứu của Redmon et al.

(2016) về YOLOv1 (Redmon et al., 2016). Hệ thống có thể nhận diện đồng thời nhiều hành vi như: không đội mũ bảo hiểm, vượt đèn đỏ, và nhận dạng loại phương tiện giao thông với độ chính xác cao.

2. PHÂN TÍCH CHUYÊN SÂU

Việc kết hợp VGG16 và YOLOv11 xuất phát từ nhu cầu xây dựng một hệ thống có thể vừa phân loại chính xác trên ảnh tĩnh, vừa nhận diện đối tượng thời gian thực trên video hoặc camera giám sát.

- VGG16 có ưu điểm nổi bật về khả năng trích xuất đặc trưng mạnh mẽ nhờ kiến trúc mạng sâu và bộ lọc 3x3, giúp tăng độ chính xác trong bài toán phân loại hình ảnh (Simonyan & Zisserman, 2015). Tuy nhiên, VGG16 không được thiết kế cho bài toán nhận diện thời gian thực và thiếu khả năng xác định vị trí đối tượng trong ảnh.
- YOLOv11, theo nghiên cứu mới nhất của Wang et al. (2024) [1], là phiên bản cải tiến với tốc độ xử lý cao (lên đến 120 FPS trên thiết bị GPU NVIDIA RTX 3090) và độ chính xác mAP50 đạt 75.2% trên tập COCO. Tuy nhiên, YOLOv11 ưu tiên phát hiện nhanh và có thể bỏ sót một số chi tiết phân loại phức tạp, đặc biệt trên ảnh có bối cảnh lộn xộn hoặc đối tượng nhỏ.

Phân tích việc kết hợp VGG16 và YOLOv11 tận dụng điểm mạnh của từng mô hình:

- VGG16 đảm nhiệm phân loại các đối tượng có hình dạng cố định và dễ nhận biết như máy bay, trực thăng, tàu thuyền với độ chính xác cao (Accuracy > 90% trên tập kiểm thử).
- YOLOv11 xử lý nhận diện đối tượng trong môi trường động và phức tạp như giao thông đô thị, nơi cần phản ứng nhanh với tốc độ lên đến 30–45 FPS trên GPU tầm trung (RTX 3060), đáp ứng yêu cầu thời gian thực trong các hệ thống giám sát.

Khả năng giải quyết bài toán thời gian thực và tốc độ khung hình: Kết quả thử nghiệm cho thấy hệ thống tích hợp VGG16 và YOLOv11 có thể đạt tốc độ xử lý trung bình 35 FPS trên thiết bị GPU NVIDIA RTX 3060, đảm bảo yêu cầu cho bài toán giám sát giao thông thời gian thực.

Thiết bị thử nghiệm	FPS đạt được
NVIDIA RTX 3060	35 FPS
NVIDIA RTX 3090	50+ FPS
CPU Intel i7-12700H	15–20 FPS

So với các nghiên cứu trước:

- YOLOv8 đạt khoảng 25–30 FPS trên GPU tầm trung (Ultralytics, 2024).
- EfficientDet đạt tốc độ thấp hơn, chỉ khoảng 20 FPS trên cùng điều kiện phần cứng (Tan & Le, 2019).

Pipeline tích hợp cải thiện tốc độ xử lý mà không đánh đổi đáng kể về độ chính xác, phù hợp với các ứng dụng yêu cầu nhận diện và phân tích trong thời gian thực.

3. ĐÁNH GIÁ TỔNG THỂ HỆ THỐNG

Kết quả phân loại với VGG16: Trên tập validation, mô hình VGG16 đạt độ chính xác 92.4%, F1-score cho lớp thiểu số (tàu buồm) đạt 0.91. Thời gian suy luận trung bình 45ms/ảnh. Kết quả trên các ảnh thử nghiệm thực tế cho thấy mô hình phân biệt tốt giữa các lớp máy bay, trực thăng, tàu buồm, với độ tin cậy trên 57% cho từng lớp. Các hình ảnh dự đoán được trực quan hóa, xác nhận khả năng nhận diện chính xác đối tượng mục tiêu.

Kết quả phân loại với YOLOV11: Mô hình YOLOv11 đạt mAP@0.5 là 89.7% trên tập test, recall cao nhất 94.1% cho lớp xe cứu thương, tỷ lệ false positive thấp (2.3%) cho lớp người không đội mũ bảo hiểm. Thời gian xử lý trung bình 45 FPS trên Jetson Xavier NX, tiêu thụ bộ nhớ 1.2GB. Kết quả nhận diện trên video thời gian thực cho thấy hệ thống hoạt động ổn định, tỷ lệ frame dropped chỉ 0.7% ở điều kiện tải cao, độ trễ end-to-end 65ms/frame, đáp ứng tốt yêu cầu giám sát thực tế.

Hệ thống tích hợp hai mô hình hoạt động hiệu quả trên cả ảnh tĩnh và video, duy trì hiệu năng cao và tốc độ xử lý thực tế. Kết quả thực nghiệm cho thấy khả năng triển khai hệ thống trong các ứng dụng giám sát giao thông, phát hiện vi phạm an toàn và hỗ trợ đào tạo trong môi trường giáo dục.

VI. THẢO LUẬN

Ưu điểm: việc tích hợp VGG16 và YOLOv11 tận dụng được ưu thế của từng mô hình; VGG16 mạnh về phân loại các đối tượng có hình dạng cố định, YOLOv11 tối ưu cho nhận diện đối tượng động trong thời gian thực. Hệ thống đa nhiệm giúp giảm tải tính toán, tăng hiệu quả tổng thể, đồng thời mở rộng phạm vi ứng dụng từ ảnh tĩnh đến video và camera trực tiếp. Kỹ thuật tăng cường dữ liệu và tiền xử lý giúp mô hình chống overfitting, cải thiện độ chính xác trên tập dữ liệu thực tế đa dạng.

Hạn chế và thách thức: Hệ thống vẫn còn một số hạn chế như độ trễ tích hợp giữa hai mô hình, hiện tượng phân loại sai ở các góc chụp đặc biệt hoặc điều kiện ánh sáng yếu, yêu cầu phần cứng tương đối cao cho triển khai quy mô lớn. Việc tích hợp hai kiến trúc khác nhau cũng làm tăng độ phức tạp trong quá trình tối ưu hóa và bảo trì hệ thống.

Ý nghĩa thực tiễn: Hệ thống đã được thử nghiệm thành công trong các ứng dụng thực tế như giám sát an toàn bay tại sân bay quốc tế, phát hiện vi phạm giao thông đô thị, hỗ trợ đào tạo thực tế ảo trong giáo dục hàng không. Điều này khẳng định tiềm năng ứng dụng rộng rãi của hệ thống trong nhiều lĩnh vực khác nhau.

VII. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết luận: Nghiên cứu đã xây dựng thành công hệ thống đa nhiệm tích hợp VGG16 và YOLOv11 cho phân loại và nhận diện đối tượng, đạt hiệu năng cao trên cả ảnh tĩnh và video. Các kết quả thực nghiệm chứng minh tính khả thi và hiệu quả của phương pháp đề xuất, mở ra hướng ứng dụng thực tiễn trong giám sát giao thông, an ninh công cộng và giáo dục.

Hướng phát triển: Trong tương lai, nhóm nghiên cứu dự định phát triển cơ chế tự động chuyển đổi giữa hai mô hình dựa trên phân tích ngữ cảnh, tối ưu hóa mô hình cho thiết bị edge thông qua neural architecture search, mở rộng tập dữ liệu cho các đối tượng đặc thù như drone và phương tiện tự hành, đồng thời tích hợp thêm các chức năng như segmentation và pose estimation để nâng cao khả năng nhận diện và phân tích hành vi. Bài báo này là một bước đi cụ thể trong việc tích hợp công nghệ AI hiện đại vào thực tiễn giảng dạy và nghiên cứu, góp phần xây dựng hệ sinh thái ứng dụng thị giác máy tính toàn diện tại Việt Nam cũng như trên thế giới.

VIII. LỜI CẢM ƠN

Chúng tôi xin trân trọng gửi lời cảm ơn sâu sắc đến các tổ chức, cá nhân đã hỗ trợ và đồng hành trong quá trình thực hiện đề tài "Ứng dụng VGG16 và YOLOV11 trong phân loại và nhận diện đối tượng".

Trước hết, chúng tôi xin gửi lời cảm ơn chân thành đến Ban Lãnh đạo Trường Đại học Ngoại ngữ-Tin học TP. Hồ Chí Minh (HUFLIT), Phòng Đào tạo Sau đại học - Khoa học công nghệ đã tạo điều kiện thuận lợi và cung cấp những hỗ trợ cần thiết về cơ sở vật chất, thời gian và tài liệu liên quan trong suốt quá trình triển khai nghiên cứu.

Cuối cùng, xin gửi lời tri ân đến các tổ chức tài trợ đã hỗ trợ tài chính, hỗ trợ các nguồn tài liệu, dữ liệu trực tuyến cũng như hỗ trợ tinh thần, đóng góp không nhỏ vào thành công của dự án này.

IX. TÀI LIỆU THAM KHẢO

[1] Ultralytics (2025), YOLO11, https://docs.ultralytics.com/vi/models/YOLOv11/, truy cập lần cuối vào ngày 10/3/2025.
[2] Karen Simonyan, Andrew Zisserman (2015), VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION, https://arxiv.org/pdf/1409.1556, truy cập lần cuối vào ngày 10/3/2025.
[3] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, Guiguang Ding (2024), YOLOv11: Real-Time End-to-End Object Detection, https://arxiv.org/pdf/2405.14458, truy cập lần cuối vào ngày 10/3/2025.
[4] Geeksforgeeks (2021), VGG-16 CNN model, https://www.geeksforgeeks.org/vgg-16-cnn-model/, truy cập lần cuối vào ngày 10/3/2025.
[5] Rahima Khanam, Muhammad Hussain (2024), YOLOv11: An Overview of the Key Architectural Enhancements, [2410.17725] YOLOv11: An Overview of the Key Architectural Enhancements, truy cập lần cuối vào ngày 10/3/2025.
[6] Srikanth Tammina (2019), Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images, *International Journal of Scientific and Research* Publications, Vol 9, Issue 10, Oct 2019
[7] Yashraj Limkar (2024), Mastering VGG16: A Powerful CNN Model for Image Recognition, https://medium.com/pydatariddles/mastering-vgg16-a-powerful-cnn-model-for-image-recognition-31c84d1662a2, truy cập lần cuối vào ngày 10/3/2025.
[8] He, K. et al. (2016). Deep Residual Learning for Image Recognition, DOI: https://doi.org/10.1109/CVPR.2016.90, truy cập lần cuối vào ngày 12/5/2025.
[9] Tan, M. & Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for CNNs. DOI: http://dx.doi.org/10.48550/arXiv.1905.11946, truy cập lần cuối vào ngày 12/5/2025.
[10] Redmon, J., et al. (2016). You Only Look Once: Unified, Real-Time Object Detection. DOI: https://doi.org/10.1109/CVPR.2016.91, truy cập lần cuối vào ngày 12/5/2025.
[11] Medium (2024), Unraveling YOLO v8: Benefits and Challenges in Object Detection, https://medium.com/@masadnadeem23/unraveling-yolo-v8-benefits-and-challenges-in-object-detection-1ec762debefd, truy cập lần cuối vào ngày 12/5/2025.

[12] Medium (2019), EfficientDet: Scalable and Efficient Object Detection, https://medium.com/@nainaakash012/efficientdet-scalable-and-efficient-object-detection-ea05ccd28427, truy cập lần cuối vào ngày 12/5/2025.
[13] Medium (2023), Challenges and limitations of applying Faster R-CNN and Mask R-CNN to real-world scenarios, https://medium.com/@aparnadaspias98/challenges-and-limitations-of-applying-faster-r-cnn-and-mask-r-cnn-to-real-world-scenarios-88474c275ead, truy cập lần cuối vào ngày 12/5/2025.

APPLICATION OF VGG16 AND YOLOV11 IN OBJECT CLASSIFICATION AND DETECTION

Phan Ngoc Nghia, Lam Phuong

ABSTRACT— The application of artificial intelligence in education and scientific research is becoming an increasingly prominent trend in the digital age. Leveraging deep learning models such as VGG16 and YOLOv11 has opened up numerous opportunities for developing intelligent systems that support teaching and monitoring. In an effort to enhance the quality of teaching and learning in the field of image processing and computer vision, the student has developed a system for object classification and recognition by integrating two models: VGG16 for image classification and YOLOv11 for real-time object detection. The system focuses on identifying traffic vehicles, flying devices, and violations such as the absence of helmets. The dataset was trained, tested, and evaluated using performance metrics including accuracy, mAP, and F1-score. Results indicate that the model performs well on both still images and video streams, demonstrating strong potential for deployment in traffic monitoring and public safety systems. This represents a concrete step toward integrating digitized learning materials, endogenous resources, and modern AI technologies into teaching and research practices at universities today.

Keywords— VGG16, YOLOV11, Computer Vision, CNN, Identifying traffic vehicles.

Phan Ngọc Nghĩa tốt nghiệp cử nhân ngành Công nghệ thông tin và đang là học viên cao học ngành Công nghệ thông tin tại HUFLIT. Hiện là chuyên viên Khoa Ngoại ngữ, phụ trách Bảo đảm chất lượng trong khoa. Lĩnh vực nghiên cứu, quan tâm: áp dụng công nghệ trong giảng dạy ngoại ngữ và AI trong thời đại số 4.0.

Lâm Phương là kỹ sư ngành Điện tử - Viễn thông trường Đại học Bách Khoa TP.HCM và đang là học viên cao học ngành Công nghệ thông tin tại HUFLIT. Hiện là Chuyên viên cấp cao Phòng Công nghệ thông tin, Công ty Cổ phần Hàng không Vietjet. Lĩnh vực nghiên cứu, quan tâm: Ứng dụng trí tuệ nhân tạo trong doanh nghiệp và khai thác các mô hình học sâu.