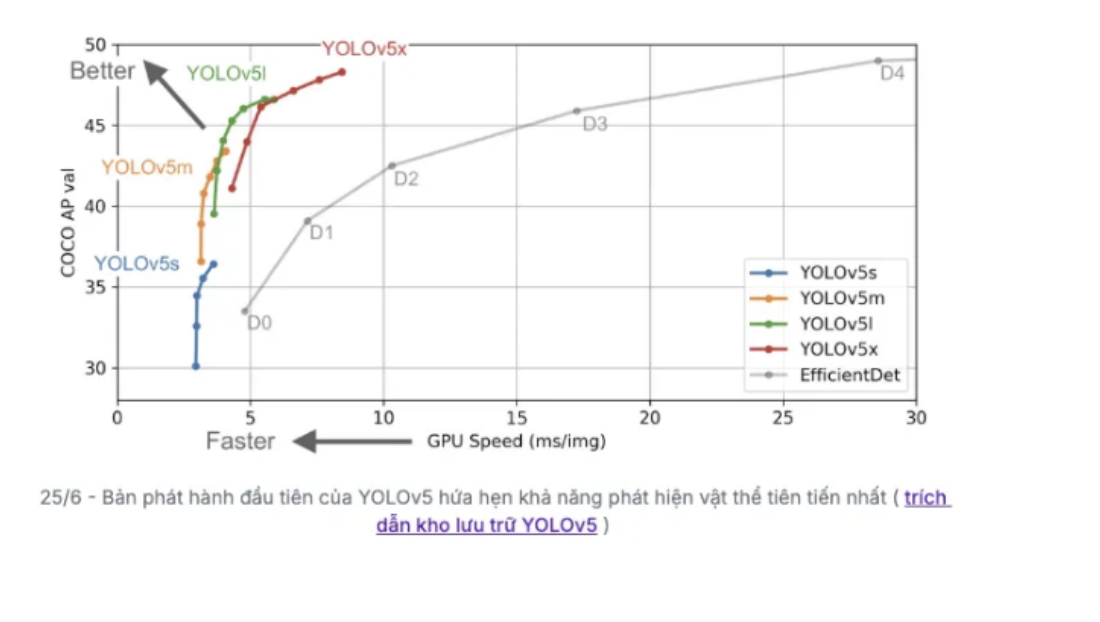
Yolov5

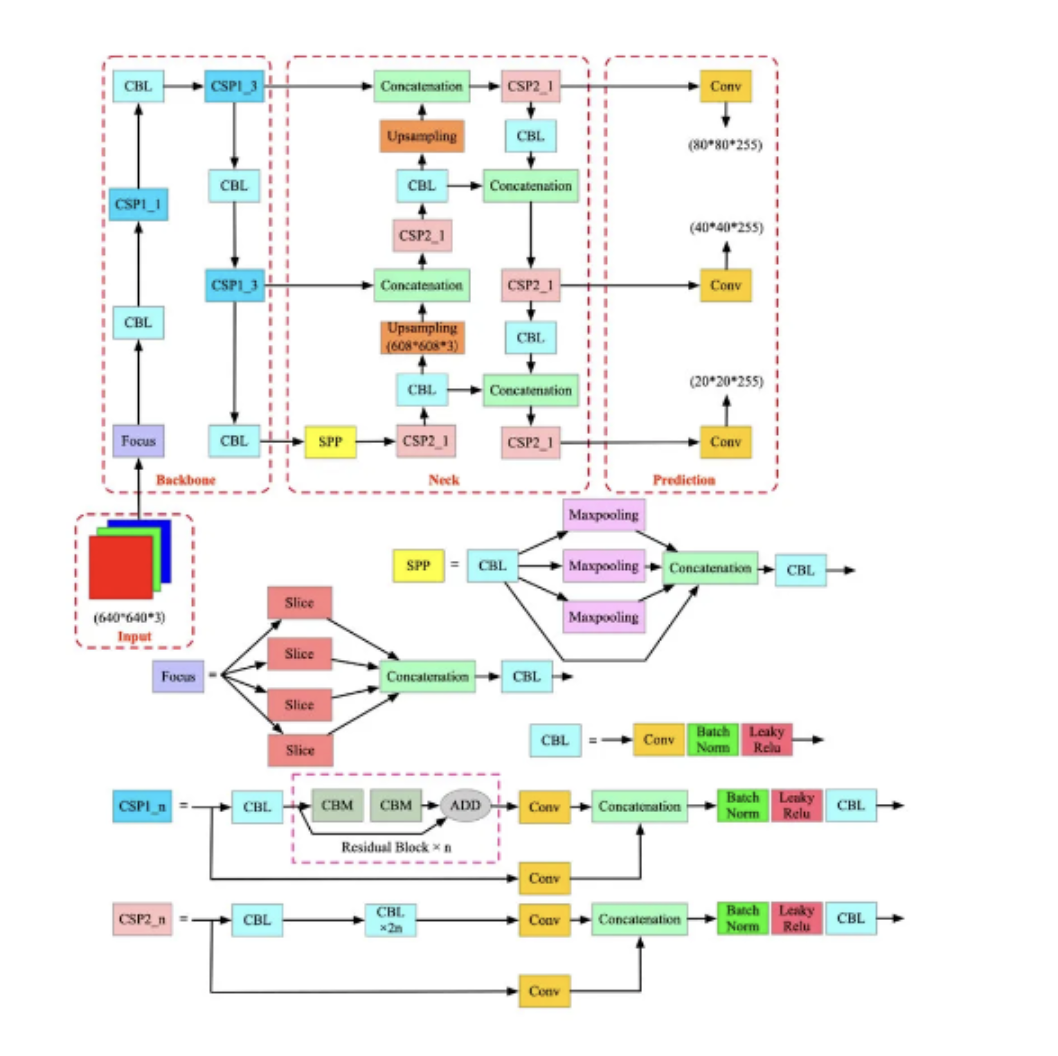
YOLOv5 (You Only Look Once version 5) là một trong những mô hình phổ biến nhất để phát hiện đối tượng (object detection), kế thừa và phát triển từ dòng mô hình YOLO ban đầu của Joseph Redmon.

Đơn giản là một mô hình trong họ mô hình thị giác máy You Only Look Once

Yolov5 có bốn phiên bản chính : nhỏ , trung bình , lớn , cực lớn

mỗi phiên bản đều có tỷ lệ chính xác cao hơn dần dần. Mỗi biến thể cũng mất một khoảng thời gian khác nhau để đào tạo.



Kiến trúc

Các thành phần của YOLOv5

YOLOv5 gồm các phần chính:

1. Backbone: Dùng để trích xuất các đặc trưng từ hình ảnh, YOLOv5 sử dụng kiến trúc CSPDarknet, giúp cải thiện hiệu quả và tốc độ tính toán.  
     
    Backbone là phần đầu tiên của mô hình YOLOv5, chịu trách nhiệm trích xuất các đặc trưng từ ảnh đầu vào. YOLOv5 sử dụng kiến trúc CSPDarknet làm backbone, được tối ưu hóa để nâng cao khả năng phát hiện đối tượng trong khi giảm thiểu chi phí tính toán.  
     
   * CSPDarknet (Cross Stage Partial Darknet): Đây là một phiên bản cải tiến của Darknet, vốn là backbone ban đầu của YOLO. CSPDarknet giúp mô hình hiệu quả hơn khi xử lý các đặc trưng bằng cách chia sẻ thông tin giữa các tầng. Các điểm nổi bật của CSPDarknet:
     + Khối CSP (Cross Stage Partial): Khối này giúp giảm lượng tính toán bằng cách chia luồng dữ liệu thành hai phần và chỉ áp dụng phép biến đổi phức tạp trên một phần. Sau đó, hai luồng này sẽ được kết hợp lại, giữ được đầy đủ thông tin nhưng với chi phí thấp hơn.
     + Residual Block (Khối dư): Residual block trong CSPDarknet giúp giữ lại các thông tin quan trọng từ các lớp trước đó, tránh mất mát thông tin và giúp mô hình hội tụ nhanh hơn.
     + Focus Layer: Lớp Focus là một cải tiến đặc biệt trong YOLOv5, dùng để chuyển đổi ảnh đầu vào thành dạng đặc trưng nhiều kênh (channel) ngay từ đầu. Thay vì chỉ giảm kích thước ảnh bằng phép pooling hoặc convolution, lớp này lấy các đặc trưng từ các vị trí khác nhau, tăng độ chi tiết và giảm độ phức tạp.
2. Neck: Neck bao gồm FPN (Feature Pyramid Network) và PANet (Path Aggregation Network), giúp kết hợp các đặc trưng từ nhiều tầng để tăng khả năng phát hiện đối tượng ở các kích thước khác nhau.  
     
    Neck là phần trung gian giữa backbone và head, chịu trách nhiệm kết hợp và xử lý các đặc trưng từ backbone để tăng cường khả năng phát hiện đối tượng ở các kích thước và vị trí khác nhau trong ảnh. YOLOv5 sử dụng FPN (Feature Pyramid Network) và PANet (Path Aggregation Network) trong thành phần Neck.  
     
   * Feature Pyramid Network (FPN): FPN được dùng để xử lý các đặc trưng ở các mức phân giải khác nhau, từ đó giúp mô hình nhận diện tốt cả các đối tượng lớn và nhỏ trong ảnh. FPN giúp tổng hợp thông tin từ các tầng sâu (high-level) và nông (low-level) trong mô hình để có cái nhìn tổng quan hơn về hình ảnh.
     + Các đặc trưng từ các tầng thấp cung cấp thông tin chi tiết, hữu ích cho việc phát hiện các đối tượng nhỏ.
     + Các đặc trưng từ các tầng cao giúp nhận diện các đối tượng lớn hơn và chứa nhiều ngữ cảnh.
   * Path Aggregation Network (PANet): PANet bổ sung vào FPN bằng cách tăng cường việc kết nối giữa các tầng, cho phép mô hình kết hợp đặc trưng từ nhiều mức độ khác nhau (từ tầng thấp đến cao và ngược lại). Điều này giúp mô hình dễ dàng phát hiện các đối tượng ở nhiều tỷ lệ kích thước khác nhau.
     + PANet tạo một con đường kết nối từ các tầng cao đến các tầng thấp, giúp mô hình không chỉ thu thập thông tin từ các tầng trên mà còn truyền thông tin quan trọng từ các tầng dưới ngược lên.
3. Head: Đầu ra của YOLOv5 bao gồm các lớp phân loại và định vị đối tượng trong hình ảnh .  
     
    Head là thành phần cuối cùng của YOLOv5, chịu trách nhiệm đưa ra dự đoán cuối cùng về vị trí và loại đối tượng. Head có nhiệm vụ tạo ra các bounding box (hộp giới hạn) và xác suất (probability) cho từng lớp đối tượng.  
     
   * Bounding Box Prediction: Head sẽ dự đoán tọa độ và kích thước của các bounding box. YOLOv5 chia ảnh thành một lưới và dự đoán các bounding box tại mỗi ô lưới. Đối với mỗi bounding box, Head sẽ dự đoán:
     + Tọa độ tâm (x, y): Vị trí tương đối của tâm đối tượng trong ô lưới.
     + Chiều rộng và chiều cao (width, height): Được chuẩn hóa dựa trên kích thước của ảnh đầu vào.
     + Confidence Score: Xác suất của mỗi bounding box chứa đối tượng.
   * Classification (Phân loại): Với mỗi bounding box, Head dự đoán lớp đối tượng mà nó thuộc về (ví dụ: người, xe, động vật). Mỗi bounding box sẽ đi kèm với một xác suất cho từng lớp đối tượng trong tập dữ liệu.
   * Anchor Boxes: YOLOv5 sử dụng các anchor boxes, là các kích thước bounding box tham chiếu được xác định trước dựa trên các kích thước đối tượng phổ biến trong tập huấn luyện. Anchor boxes giúp mô hình nhanh chóng học được vị trí và kích thước của các đối tượng có kích thước và hình dạng khác nhau.

Cách huấn luyện YOLOv5

Huấn luyện YOLOv5 có thể thực hiện thông qua mã nguồn của Ultralytics trên GitHub với các bước:

Cài đặt môi trường và các thư viện cần thiết.  
  
 Bước 1: Clone mã nguồn YOLOv5 từ kho lưu trữ GitHub của Ultralytics:  
  
 git clone <https://github.com/ultralytics/yolov5>

cd yolov5

Bước 2: Cài đặt các thư viện yêu cầu, bao gồm PyTorch, OpenCV, và các thư viện khác:

pip install -r requirements.txt

2 .Chuẩn bị dữ liệu và cài đặt cấu trúc thư mục

Bước 3 : cấu trúc thư mục dữ liệu

data/

├── images/

│   ├── train/

│   └── val/

└── labels/

    ├── train/

    └── val/

Bước 4 : Định dạng nhãn: File nhãn .txt của mỗi ảnh bao gồm:

<class\_id> <x\_center> <y\_center> <width> <height>

Bước 4 : Tệp cấu hình dữ liệu: Tạo một file .yaml (ví dụ: data.yaml) để định cấu hình dữ liệu với nội dung như sau:

path: ./data  # Đường dẫn đến thư mục dữ liệu

train: images/train  # Thư mục chứa ảnh huấn luyện

val: images/val  # Thư mục chứa ảnh kiểm thử

nc: 2  # Số lớp đối tượng

names: ['class1', 'class2']  # Tên các lớp đối tượng

3 . Cấu hình mô hình

4 . Huấn luyện mô hình

1. Theo dõi quá trình huấn luyện
2. Đánh giá và triển khai
3. Tinh chỉnh và tối ưu hoá

YOLOv6:

Tối ưu cho thiết bị di động và nhúng, phù hợp với môi trường có tài nguyên hạn chế.

Sử dụng EfficientRep Backbone và cải tiến cấu trúc, giúp tăng tốc độ và giảm số lượng tham số.

YOLOv7:

Được tối ưu về độ chính xác và tốc độ, vượt trội hơn YOLOv5 và YOLOv6 trên nhiều chỉ số.

Giới thiệu ELAN và các cải tiến trong thiết kế Head để tăng hiệu suất cho đối tượng nhỏ hoặc chồng chéo.

YOLOv8:

Kiến trúc hoàn toàn mới, với giao diện dễ sử dụng nhất, phù hợp cho các ứng dụng thực tế.

Tối ưu hóa cho phần cứng hiện đại và đạt độ chính xác cao nhất, dễ triển khai và sử dụng.