YOLO V8

INPUT – BACKBONE – NECK – HEAD – OUTPUT

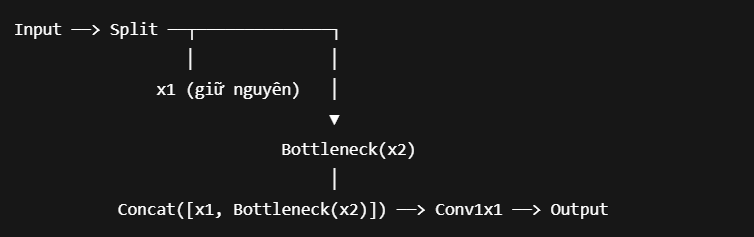
Trước khi đưa vào model – thì YOLOV8 sẽ tự độ resize ảnh (Auto-reshape) theo tỷ lệ phù hợp với model. Và padding (tạo viền đen) để ảnh có kích thước vuông, giúp model hoạt động chính xác khi nào các ảnh có tỷ lệ khác nhau.

**BACKBONE**: Dùng C2f block, cụ thể: (Fusion)

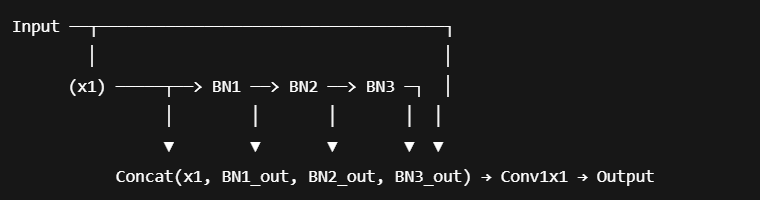
Đầu tiên phải nói tới C2. C2 trong C2f là viết tắt CSP(Cross Stage Partial) Bottleneck với 2 Conv.

C2f sẽ triển khai nhanh hơn C2. Cải thiện tốc độ thực hiện trong khi hiệu suất vẫn duy trì ở mức tương đương. Được sử đổi một số thành phần từ C2.

C2 CŨ:



C2f:



**NECK**: Sử dụng BiFPN (Bidirectional Feature Pỷamid Network)

Giải thích đơn giản thì Neck này sẽ gộp đặc trưng của các tầng (multi scale), giống như nối thông tin lại từ nhiều góc độ để có thể nhìn rõ hơn. Ngoài ra BiFPN còn giúp xử lý nhanh hơn.

**HEAD** dự đoán

Dùng Anchor-free: không cần xác định anchor boxes trước

Các pixel trên feature map sẽ trực tiếp dự đoán

**OUTPUT YOLOV8:**

Output bao gồm 3 thành phần: class: loại vật thể dự đoán được (ví dụ: 0 – con chó, 1 – con mèo)

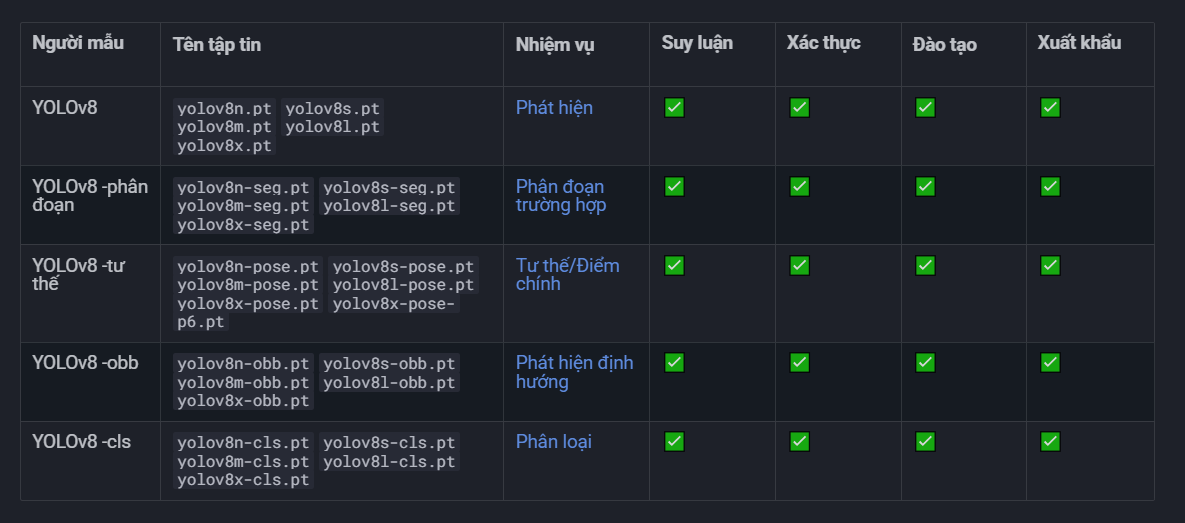
Confidence: xác suất xác định vật thể

Toạ độ [x, y, w, h]

**CÁC CẢI TIẾN CHÍNH**

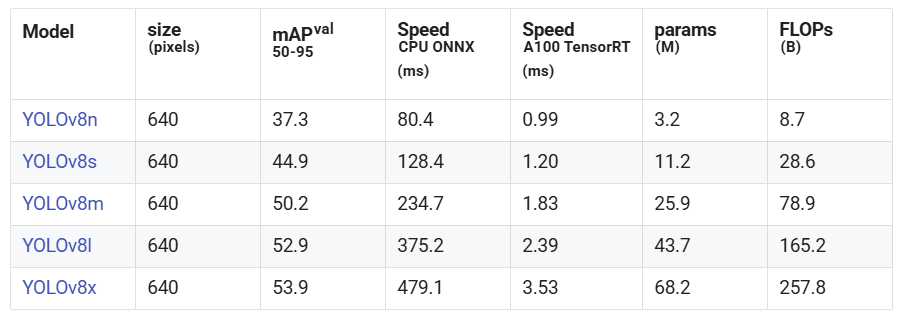
Chuyển sang anchor free

Hỗ trợ nhiều công việc hơn, cụ thể là: object detection, segmentation, classification



* Các biến thể

**VỀ ĐỘ CHÍNH XÁC**



**FPS**

|  |  |
| --- | --- |
| ~1000 FPS (YOLOv8n) | Benchmark tối ưu (A100, TensorRT, không hiển thị ảnh) |

|  |  |
| --- | --- |
| ~140 FPS (YOLOv8x) | Sử dụng trên máy tính cá nhân hoặc không dùng TensorRT |

**PHÁT HIỆN VẬT THỂ NHỎ**

Theo COCO định nghĩa: vật thể nhỏ có kích thước <32x32 pixel (trên ảnh 640)

Ví dụ như biển số xe, người ở xa,vv

Phát hiện tốt, tuy nhiên phụ thuộc nhiều yếu tố. Ví dụ: ảnh có độ phân giải thấp vì đã bị nén, thì dễ bị mất đặc trưng. Rồi vật thể bị mờ, thiếu ánh sáng thì mô hình khó học.

**TRƯỜNG HỢP MÔI TRƯỜNG BỊ ẢNH HƯỞNG**

Bản chất của YOLOV8 vẫn là xử lý từng Dframe, kể cả video. Tuy nhiên, có thể cải thiện

Về TH sương mù, mờ: có thể làm mất đặc trưng hình học khiến mô hình khó phân biệt.

Thiếu sáng -> Constrast thấp, khó bắt được vật

Ngược sáng -> Mất chi tiết ảnh

Trong video, nếu vật thể di chuyên quá nhanh thì dẫn đến mờ chuyển động, kh bắt được kịp

Cải thiện:

Về video, có thể dùng tracking/ensemble để ổn định Dframe. Hoặc frame fusion.

Mờ: Trong YOLO có thuật toán xử lý ảnh mờ:

Fine-tune YOLOV8 với ảnh mờ: Có thể sử dụng Roboflow Augmentation

Trường hợp thiéu sáng:

Dùng OpenCV tăng sáng ảnh (histogram stretch)

Fine-tune thêm ảnh đêm thật, COCO hơi hạn chế ảnh đen.

Có thể sử dụng camera hồng ngoại