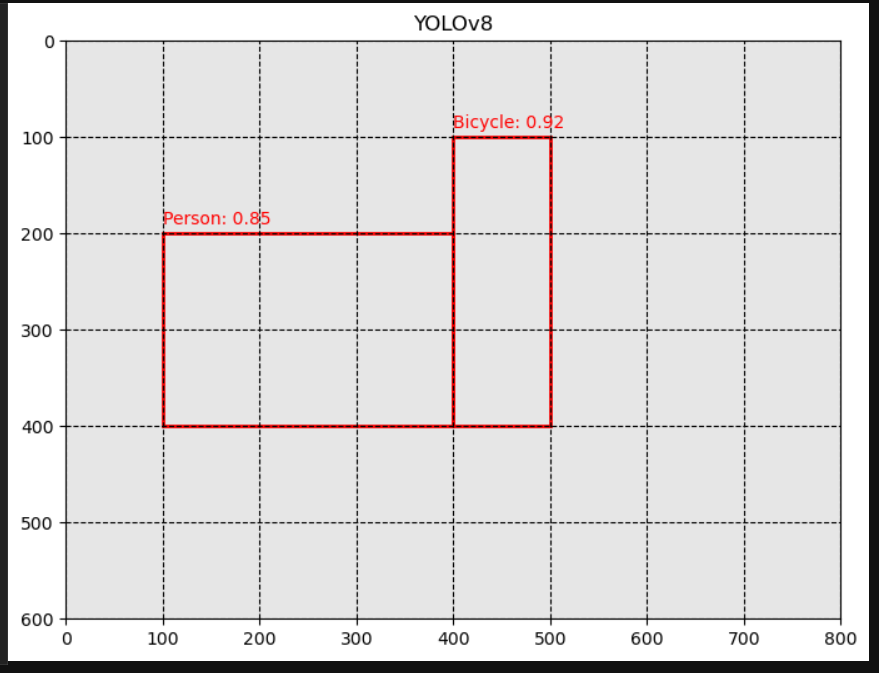
**YOU ONLY LOOK ONCE**

**Dựa vào hình ảnh, mô tả chi tiết YOLO**

****

Lấy hình bên trên là hình ảnh, một hình ảnh được gọi là lưới với nhiều ô nhỏ, gọi là grid cells. Mỗi grid cell có thể chứa nhiều đối tượng, tuỳ thuộc vào vị trị của vật đó.

Sau đó, bên trong mỗi ô sẽ chứa một hoặc nhiều đối tượng (bounding box) – là hình màu đỏ được vẽ xung quanh các vật thể.

Và bounding box bên trong lưới đó sẽ chứa một số thông tin của vật ở bên trong. Đó chính là lớp (class) và độ tin cậy(confidence score).

Class: là thứ mà bounding box phát hiện, ví dụ như con người, chiếc xe

Confidence: độ tin cậy là mức độ chính xác của model yolo khi nhận diện được hình ảnh trên.

**Cách tính các tham số bên trong yolo**

Toạ độ Center của BB:

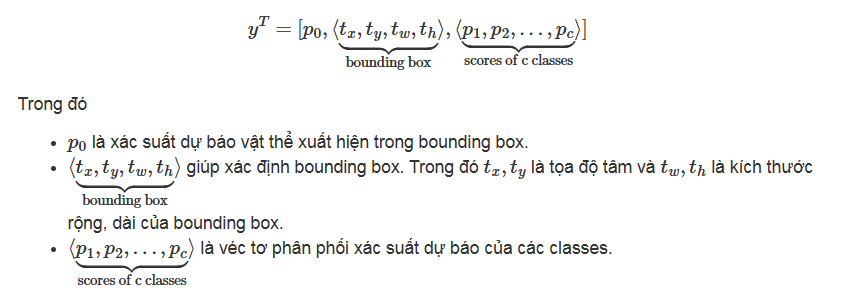
**Tx​=σ(tx​)+cx​, Ty​=σ(ty​)+cy​**

Tx, Ty là toạ độ của grid cell nơi mà BB thuộc vào, σlà sigmoid

**w=tw \* ​e^tw​, h=th​ \* e^th​**

Trong đó tw,tht\_w, t\_htw​,th​ là các giá trị chiều rộng và chiều cao gốc của grid cell

**OUTPUT CỦA YOLO**



Bounding box đã giải thích ở trên

Còn p được tính bằng p(class\_i) = σt(i)

CÁC PHIÊN BẢN

V1: 2016, cho ra độ chính xác cao hơn các model OD trong thời điểm đó.

Chia ảnh đầu vào thành lưới 7x7. Mỗi ô dự đoán Bouding Box và Label

V2: 2017,

Cải tiến kiến trúc mạng (Darknet-19). Thay cho CNN trong V1.

THêm Anchor Box -> có thể dự đoán chính xác hơn với nhiều kích cỡ

Học cùng lúc trên COCO + ImageNet -> Nhận diện 9000 lớp

*AnchorBox: Dự đoán nhiều Bounding Box cho mỗi ô lưới (grid cell)*

*Joint Training: P2 học đồng thời 2 tập data lớn là ImageNet & COCO*

V3: 2018, DarkNet-53: Kết hợp kỹ thuật residual connections: giải quyết vđ mất thtin trong các nơ ron, giúp đọc được đặc trưng phức tạp hơn, bottleneck layers giảm sl tham số và các tính toán k cần thiết mà k giảm accuracy.

Multi-scale predic: dự đoán 3 độ phân giải (kích thước small,medium,large) khác nhau giúp nhận diện vật thể nhỏ tốt hơn.

Sử dụng Sigmoid thay cho Softmax: mỗi dự đoán ko phụ thuộc vào các class khác, cải thiện sự linh hoạt khi nhận diện chồng lấn (phát hiện nhiều đối tượng trong một bức ảnh).

YOLO sử dụng IoU (Intersection over Union). Kết hợp với CloU (Complete IoU Loss) giúp tối ưu hoá IoU, cụ thể:

1.Nếu 2 BB có tỷ lệ khác nhau (1 dài 1 rộng), CloU sẽ giảm điểm số IoU để phản ánh việc trên.

2.Center BB: CloU trừng phạt việc dự đoán sai vị trí center của BB.

V4: 2020, Alexey Bochkovskiy: Ko phải nhóm tác giả cũ Redmon đã rút

Dùng CSPDarknet-53 làm backbone: chia mạng nơ ron thành các phần nhỏ hơn và thực hiện nhiều phép tính hơn, cải thiện độ 9 xác.

Neck: sd PANet. Kết nối các đặc trưng từ các tầng khác nhau -> nhận diện tốt hơn vs các vật thể kích thước khác nhau.

Thêm các cải tiến: Mosaic data augmentation Gộp bốn bức ảnh ngẫu nhiên lại, giúp model học được sự biến đổi của các đối tượng khi chúng xuất hiện trong các cảnh nền khác nhau. Giúp đa dạng tình huống.

MixUp Aug: Trộn các bức ảnh với label, giúp model học được sự kết hợp của nhiều đối tượng trong một cảnh.

DropBlock Regularization: Giống Dropout