# **Football Player Detection**

Võ Duy Khang 20521441

Nguyễn Huy Khánh 20521451

Trần Đăng Khoa 20521475

1. Mô tả bài toán:

Input: Một tấm ảnh màu hoặc 1 video cần nhận diện cầu thủ.

Output: Ảnh màu có bouding box quanh cầu thủ cần nhận diện với tên của cầu thủ (nếu cận cảnh) và với quốc gia cầu thủ (nếu từ xa).

Ngữ cảnh:

Nhận diện cầu thủ là một công nghệ được sử dụng trong bóng đá để phân loại và xác định các cầu thủ trên sân. Nó có thể được sử dụng để ghi lại và phân tích các hoạt động của cầu thủ trong trận đấu, bao gồm cả vị trí, tốc độ, và các hoạt động khác như đánh đầu, chạy, và tấn công.

Công nghệ nhận diện cầu thủ cũng có thể được sử dụng để giúp cho các đội bóng và tổ chức thi đấu có thể quản lý và giám sát trận đấu một cách chính xác hơn. Nó cũng có thể giúp cho các chuyên gia phân tích và đánh giá hiệu suất của các cầu thủ trong trận đấu để cải thiện kỹ thuật và phong cách chơi.

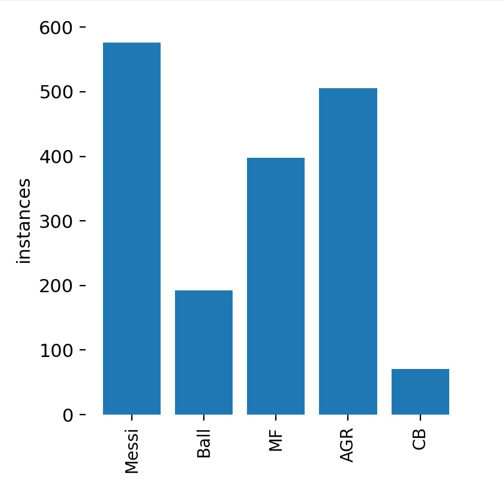
Tổng thể, nhận diện cầu thủ là một công nghệ quan trọng và cần thiết đối với sự phát triển và nâng cao chất lượng của bóng đá hiện đại.

1. Mô tả dữ liệu:

Cách thức xây dựng bộ dữ liệu: Dữ liệu bao gồm 1500 tấm ảnh của các cầu thủ AGR trên sân cỏ tự tìm và tải về trên google. Và các hình ảnh cắt ra từ video (youtube).

Các thao tác tiền xử lý dữ liệu: Tiến hành gán nhãn dữ liệu bằng Tool <https://www.makesense.ai>

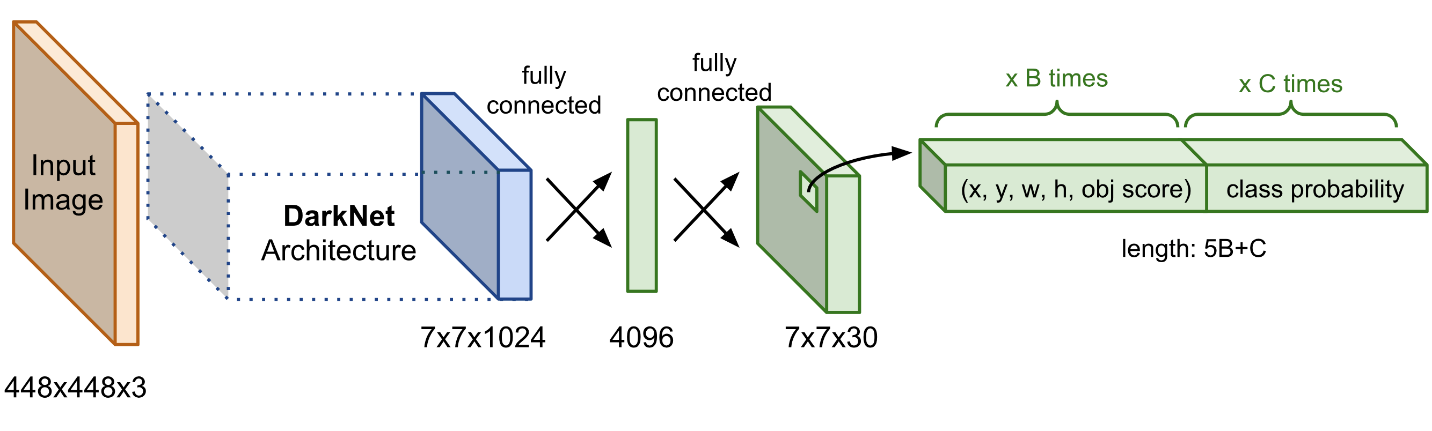
Bộ dữ liệu bao gồm 1200 image cho quá trình train và 300 image cho quá trình test.



1. Mô tả đặc trưng:

Feature engineering ( kỹ thuật trích xuất đặc trưng ) :

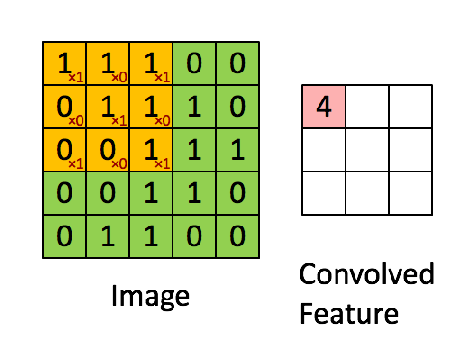
Yolo là 1 mô hình object detection . vì vậy không chỉ gãn nhãn được đối tượng mà nó còn xác định được vị trí của chúng bằng bouding box



Sơ đồ kiến trúc mạng YOLO. Thành phần Darknet Architechture được gọi là base network có tác dụng trích suất đặc trưng. Output của base network là một feature map có kích thước 7x7x1024 sẽ được sử dụng làm input cho các Extra layers có tác dụng dự đoán nhãn và tọa độ bounding box của vật thể.

Base network của YOLO sử dụng chủ yếu là các convolutional layer và các fully conntected layer. Các kiến trúc YOLO cũng khá đa dạng và có thể tùy biến thành các version cho nhiều input shape khác nhau.

Convolutional layer là một cửa sổ trượt (Sliding Windows) trên một ma trận như mô tả hình dưới:



Các convolutional layer có các parameter(kernel) đã được học để tự điều chỉnh lấy ra những thông tin chính xác nhất mà không cần chọn các feature.

Trong hình ảnh ví dụ trên, ma trận bên trái là một hình ảnh trắng đen được số hóa. Ma trận có kích thước **5×5** và mỗi điểm ảnh có giá trị 1 hoặc 0 là giao điểm của dòng và cột.

Convolution hay tích chập là nhân từng phần tử trong ma trận 3. Sliding Window hay còn gọi là kernel, filter hoặc feature detect là một ma trận có kích thước nhỏ như trong ví dụ trên là 3×3.

Convolution hay tích chập là nhân từng phần tử bên trong ma trận 3×3 với ma trận bên trái. Kết quả được một ma trận gọi là Convoled feature được sinh ra từ việc nhận ma trận Filter với ma trận ảnh 5×5 bên trái.

Extra layer (dùng để phát hiện vật thể trên feature map trên base network).

1. Data processing pipeline:

Gồm 1500 tấm ảnh có hình messi và các đồng đội trên sân cỏ được chụp cận cảnh hoặc từ xa

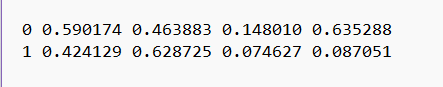
Chia 1500 tấm thành 2 fouder : fouder 1 : 1200 tấm , fouder 2- 300 tấm.

Gán nhãn cho toàn bộ ảnh : sử dụng Tool <https://www.makesense.ai> để gán nhãn dối tượng 🡪 có được 1500 tấm ảnh với 1500 file txt tương ứng. Các file txt chứa các thông số là các nhãn đối tượng và tọa độ bouding box tương ứng.

Tiến hành train dữ liệu.

VD :



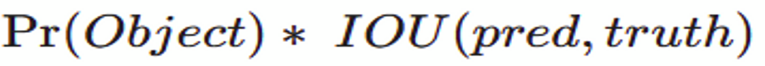


1. Mô tả thuật toán:

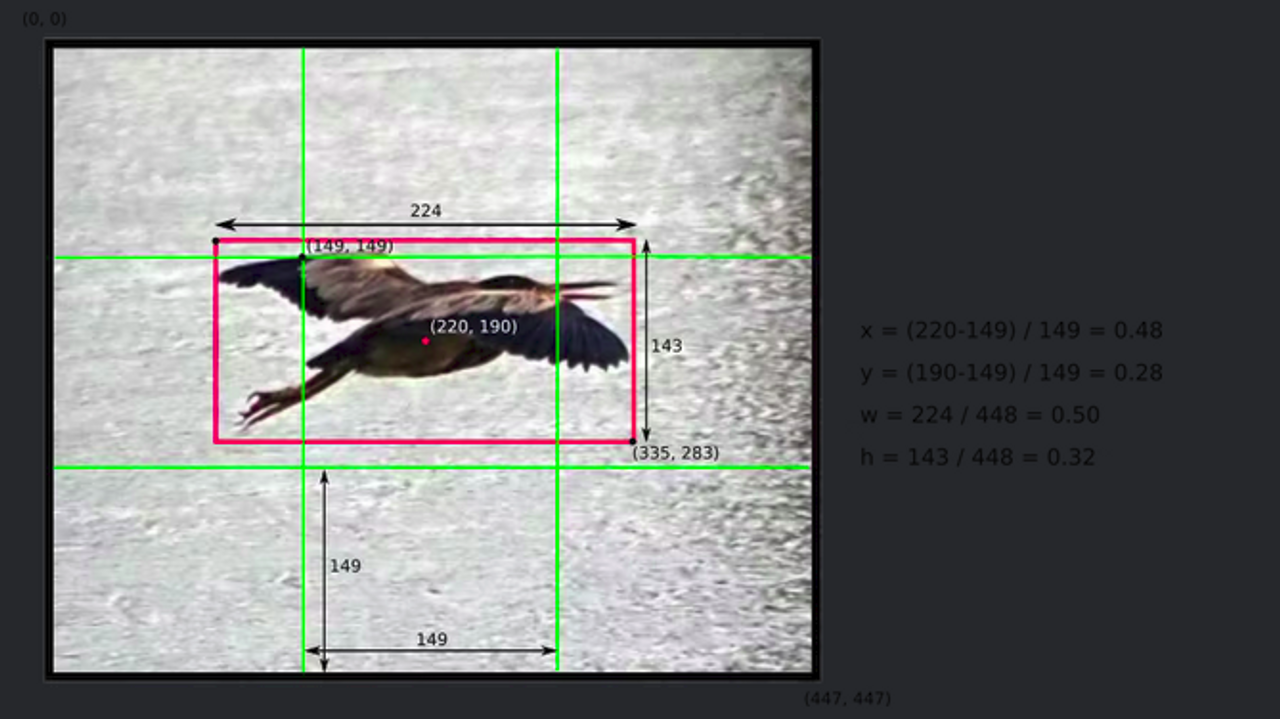
## Bounding box

Với input của mạng Yolo là một hình ảnh, Yolo sẽ chia hình ảnh thành các ô lướt có kích thước SxS ( thường S = 3,5,7) sau đó ta nhận lại được một vector bouding box và dự đoán của các class có trong ảnh đó. Nói cụ thể là một vector có 5 thành phần lần lượt là (x,y,w,h,p)

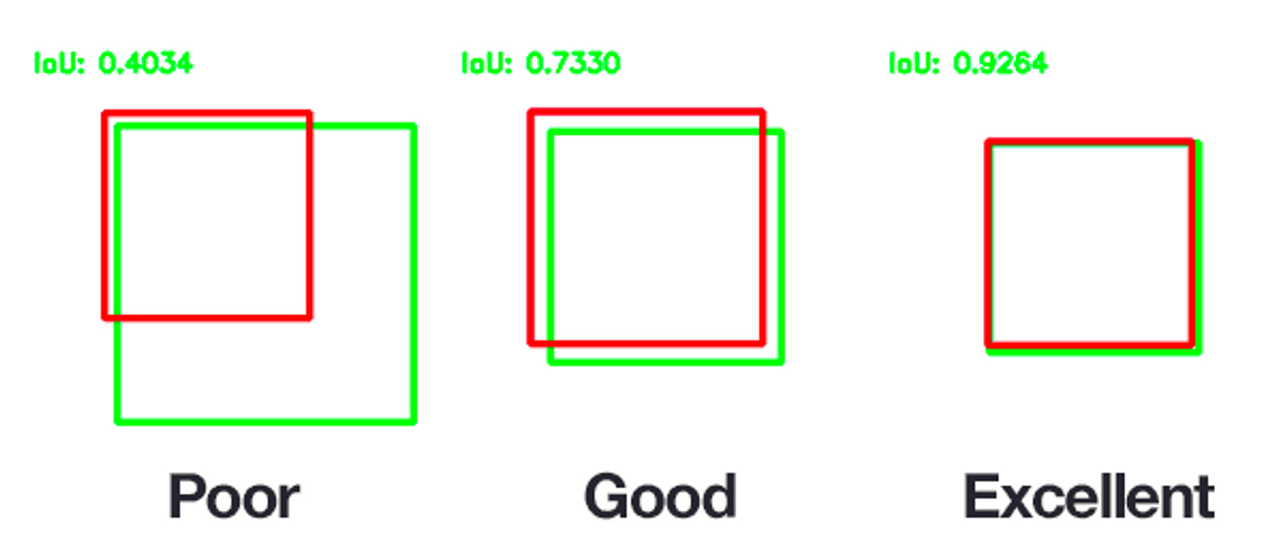
Lần lượt : x,y là toạ độ tâm của bounding box; w,h: là chiều rộng (weight) và chiều cao ( height) của bounding box; p(prediction) được định nghĩa là:



Thông thường các toạ độ này sẽ được chuẩn hoá về các giá trị nằm trong khoảng [0,1] thông qua tỉ lệ với kích thước với size của ảnh đầu vào. Dưới đây là một ví dụ :



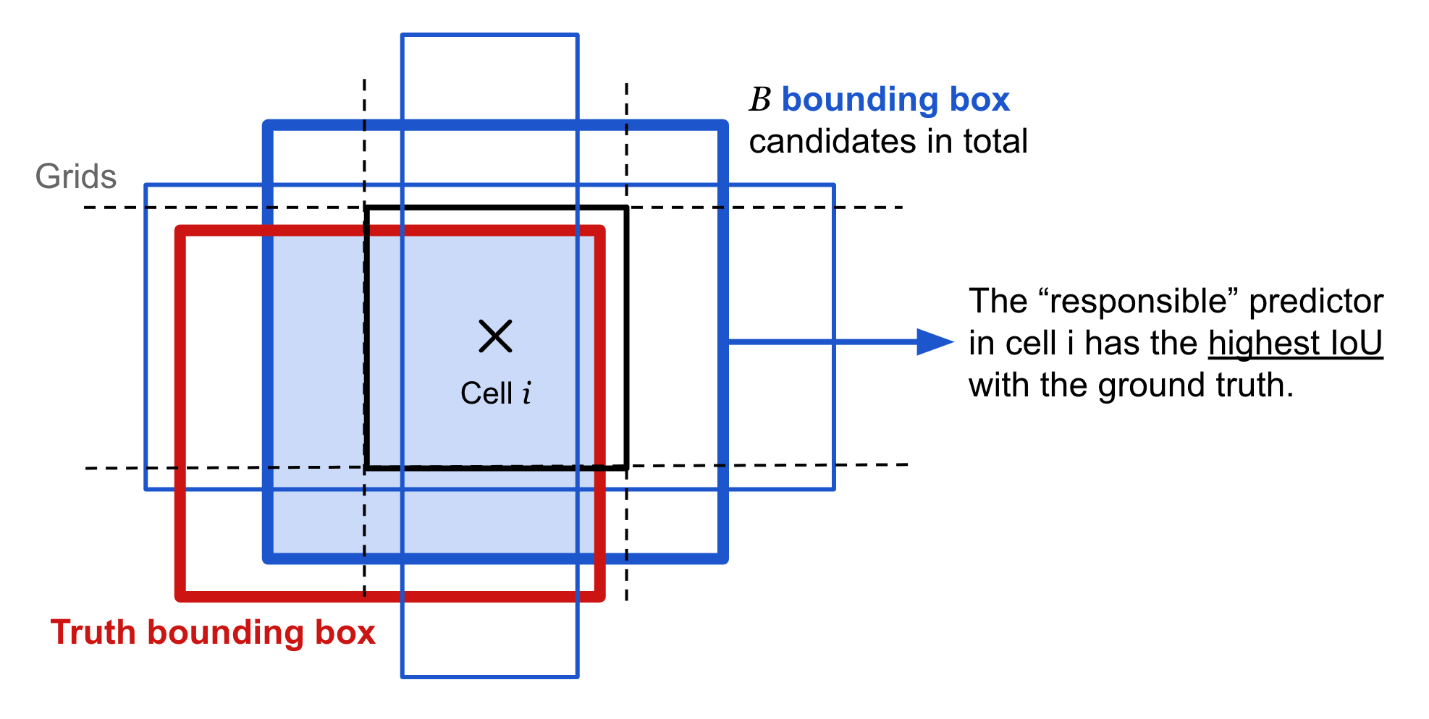
IoU hay Intersection over Union là một phương pháp để đo độ chính xác của một bộ nhận diện đối tượng. Đây là một chỉ số mà ta sẽ thường xuyên gặp ở nhiều thuật toán khác nhau có thể kể ra như : HOG và Linear SVM và các mạng CNN trong đó có cả Yolo.



Anchor box

Để tìm được bounding box cho vật thể, YOLO sẽ cần các anchor box làm cơ sở ước lượng. Những anchor box này sẽ được xác định trước và sẽ bao quanh vật thể một cách tương đối chính xác. Sau này thuật toán regression bounding box sẽ tinh chỉnh lại anchor box để tạo ra bounding box dự đoán cho vật thể. Trong một mô hình YOLO:

* Mỗi một vật thể trong hình ảnh huấn luyện được phân bố về một anchor box. Trong trường hợp có từ 2 anchor boxes trở lên cùng bao quanh vật thể thì ta sẽ xác định anchor box mà có IoU với ground truth bounding box là cao nhất.

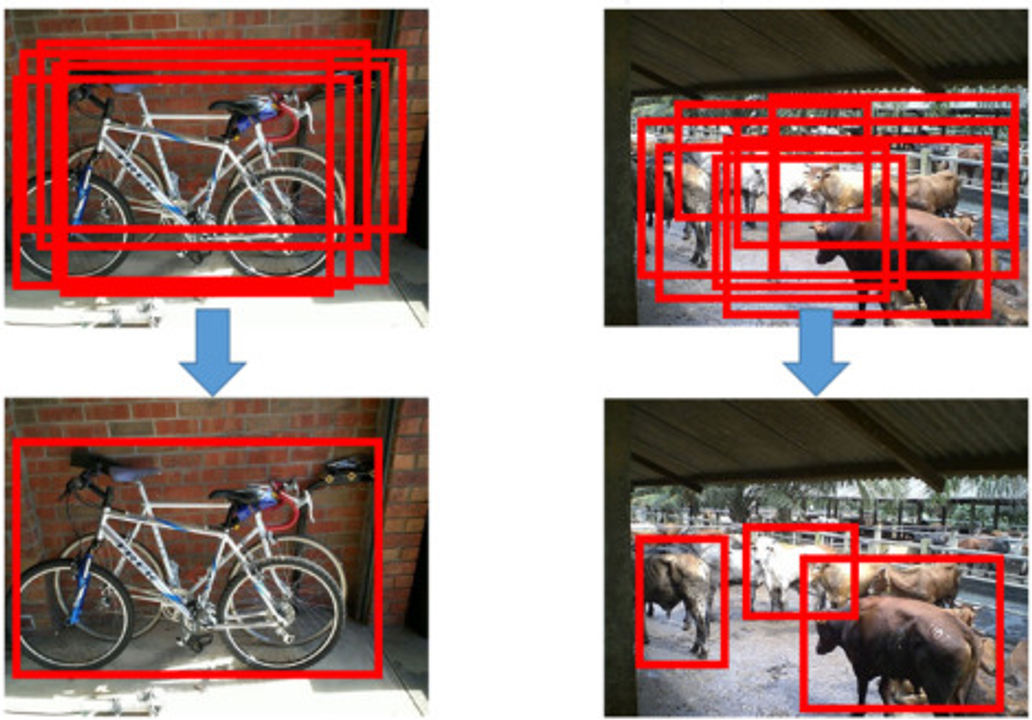


Xác định anchor box cho một vật thể. Từ Cell i ta xác định được 3 anchor boxes viền xanh như trong hình. Cả 3 anchor boxes này đều giao nhau với bounding box của vật thể. Tuy nhiên chỉ anchor box có đường viền dày nhất màu xanh được lựa chọn làm anchor box cho vật thể bởi nó có IoU so với ground truth bounding box là cao nhất.

* Mỗi một vật thể trong hình ảnh huấn luyện được phân bố về một cell trên feature map mà chứa điểm mid point của vật thể. Chẳng hạn như hình chú chó trong hình 3 sẽ được phân về cho cell màu đỏ vì điểm mid point của ảnh chú chó rơi vào đúng cell này. Từ cell ta sẽ xác định các anchor boxes bao quanh hình ảnh chú chó.
* Như vậy khi xác định một vật thể ta sẽ cần xác định 2 thành phần gắn liền với nó là (cell, anchor box). Không chỉ riêng mình cell hoặc chỉ mình anchor box.
* Một số trường hợp 2 vật thể bị trùng mid point, mặc dù rất hiếm khi xảy ra, thuật toán sẽ rất khó xác định được class cho chúng.

## Non Max supperession

Là một kĩ thuật làm giảm cái bounding box được dự đoán chồng tâm lên nhau. Hình ảnh bên dưới sẽ là ví dụ cho bạn dễ hiểu vấn đề đang được để cập



Ta có thể nhận ra ngay với những object lớn thì các bounding box xác định tâm tức là x,y sẽ bị rơi vào nhiều ô lưới ( grid S x S) dẫn đến tình trạng các Bounding box đè chồng lên nhau. Và NMS đã được triển khai như sau để loại bỏ tình trạng trên.

*1. Loại bỏ những đối tượng có confidence C < C-threshold ( ngưỡng này chúng ta có thể re setup).*

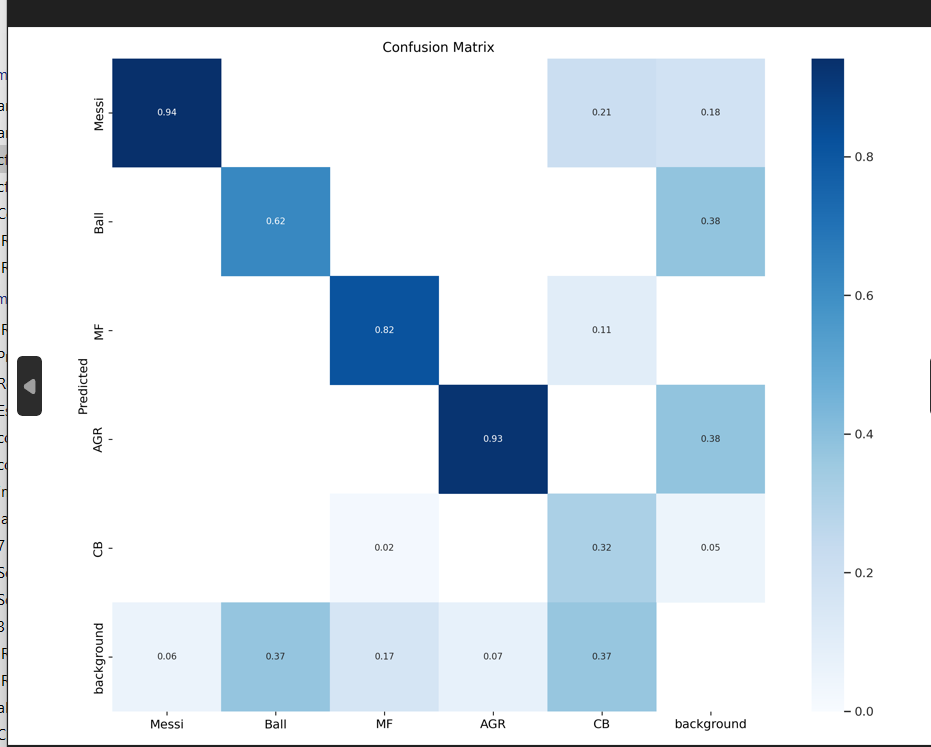
*2. Sắp xếp những đối tượng có confidence C theo thứ tự giảm dần.*

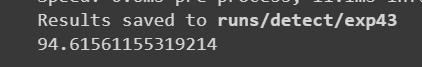
*3. Chọn bounding box của những vật thể có confidence C cao nhất và xuất ra kết quả.*

*4. Loại bỏ những bounding box có IOU < IOU-threshold (ngưỡng này chúng ta có thể re setup)*

*5. Quay lại bước 3 cho tới khi mọi kết quả được kiểm tra.*

1. Đánh giá kết quả và kết luận:





1. Source code

[https://colab.research.google.com/drive/1a2etwWdgpvfSpRB-TxJ4ucdjBfEWxqHi?authuser=2&fbclid=IwAR0sW4vXTnUxH5jthDA4VPSB9XXY-4hzcf9uG5Vrp79fUQN7lbpeXsBxP1k#scrollTo=VOUXpevEorjf](https://colab.research.google.com/drive/1a2etwWdgpvfSpRB-TxJ4ucdjBfEWxqHi?authuser=2&fbclid=IwAR0sW4vXTnUxH5jthDA4VPSB9XXY-4hzcf9uG5Vrp79fUQN7lbpeXsBxP1k%23scrollTo=VOUXpevEorjf)