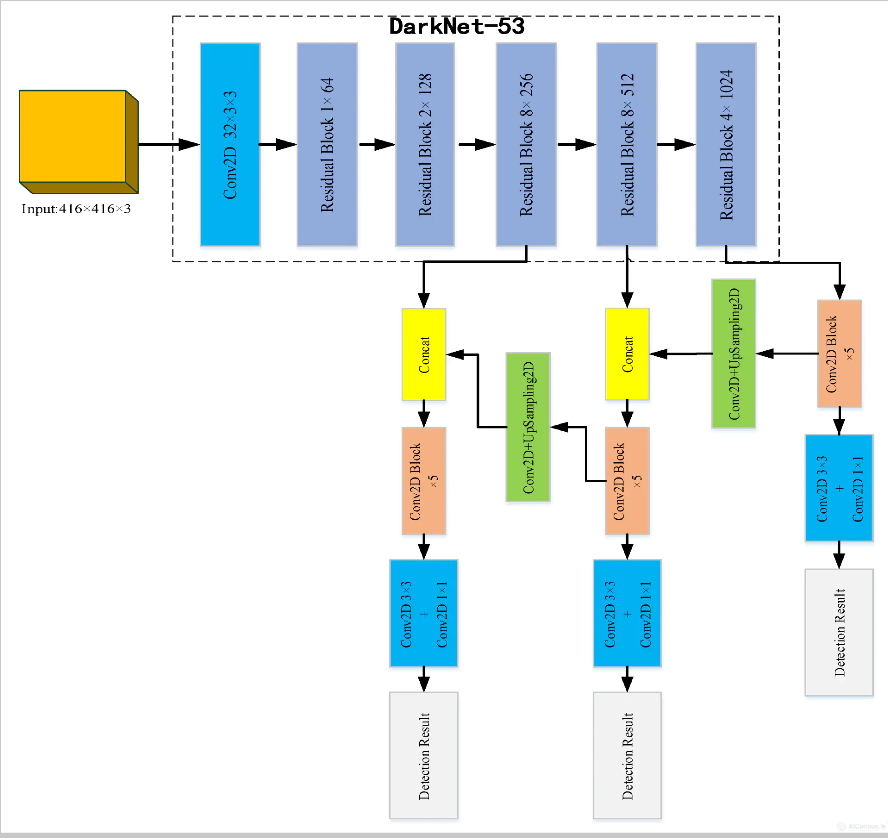
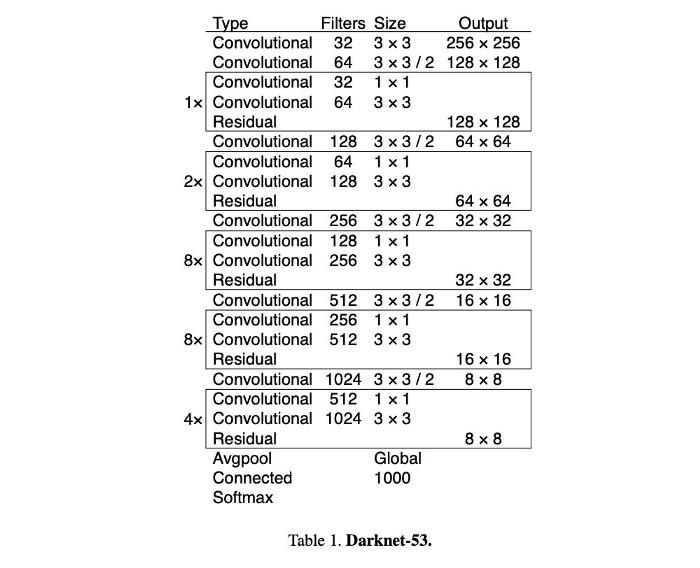
1.Kiến trúc

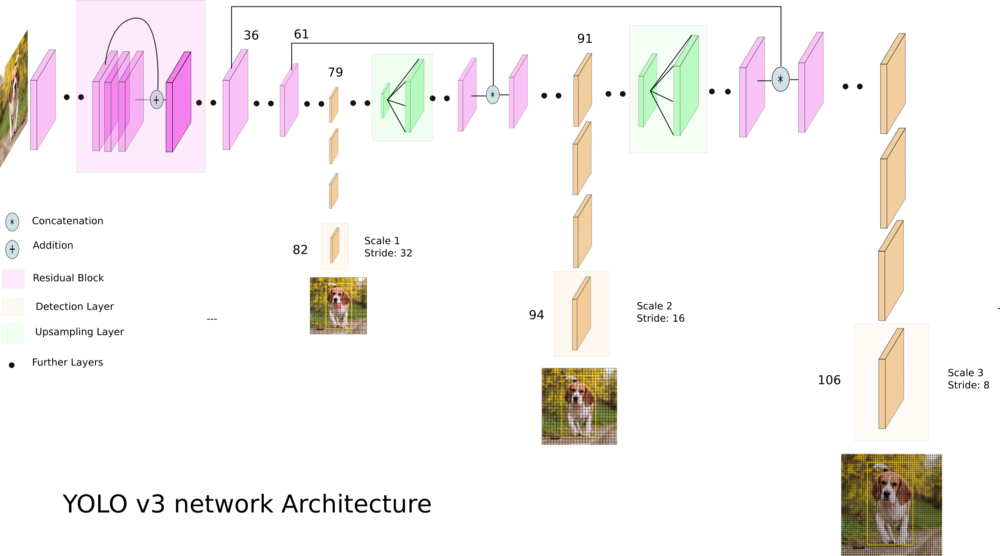


* **Logistic regression cho confidence score:** predict độ tự tin của bounding box (có chứa vật hay không) sử dụng logistic regression
* **Thay softmax bằng các logistic classifier rời rạc:** sử dụng các logistic classifier thay vì softmax cho việc classify đối tượng. Việc này cho hiệu quả tốt hơn nếu các label không "mutually exclusive", tức là có thể có đối tượng cùng thuộc 2 hay nhiều class khác nhau. Ví dụ với bài toán cần phát hiện 2 đối tượng là "chó" và "chó Phú Quốc". Rõ ràng nếu đối tượng là "chó Phú Quốc" thì nó cũng thuộc class "chó", và việc sử dụng softmax là không phù hợp trong trường hợp này.
* **Backbone mới - Darknet-53:** Backbone được thiết kế lại với việc thêm các residual blocks (kiến trúc sử dụng trong ResNet).
* **Multi-scale prediction:** sử dụng kiến trúc Feature Pyramid Networks (FPN) để đưa ra các dự đoán từ nhiều scale khác nhau của feature map. Việc này giúp tận dụng các feature map với độ thô - tinh khác nhau cho việc dự đoán.
* **Skip-layer concatenation:** cũng thêm các liên kết giữa các lớp dự đoán. Mô hình upsample các lớp dự đoán ở các tầng sau và sau đó concatenate với các lớp dự đoán ở các tầng trước đó. Phương pháp này giúp tăng độ chính xác khi predict các object nhỏ.

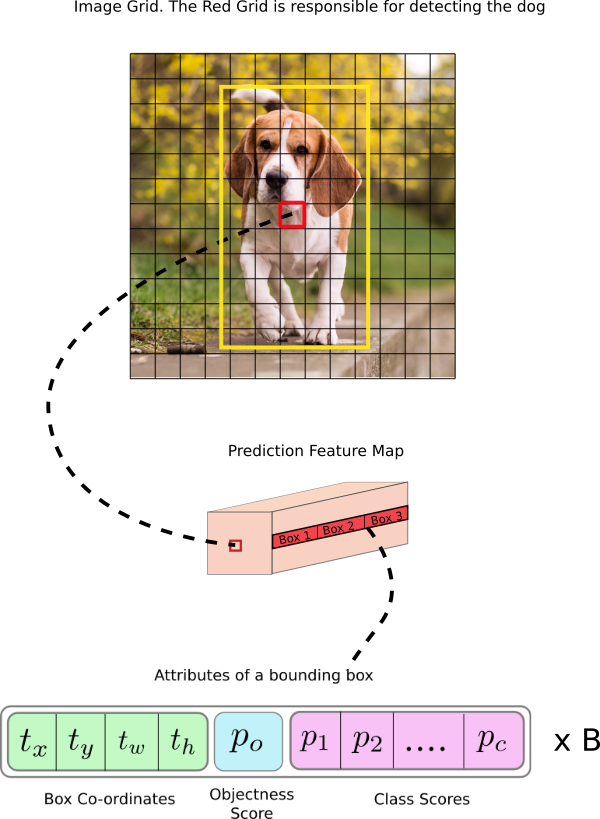


YOLO v3 sử dụng một biến thể của Darknet thêm các residual blocks (kiến trúc sử dụng trong ResNet), ban đầu có mạng 53 lớp được đào tạo trên Imagenet. Để có nhiệm vụ phát hiện, 53 lớp nữa được xếp chồng lên nhau, cho chúng ta một kiến trúc gồm 106 lớp convolutional cho YOLO v3. Đây là lý do đằng sau sự chậm chạp của YOLO v3 so với YOLO v2.

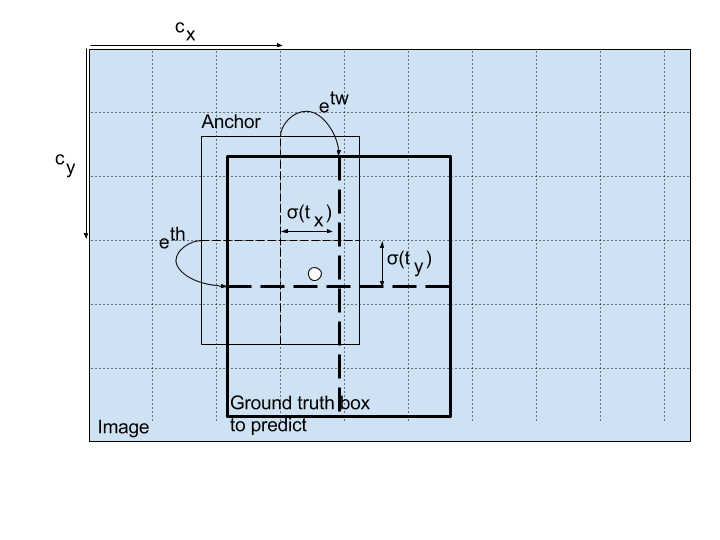
3.Quá trình phát hiện vật thể

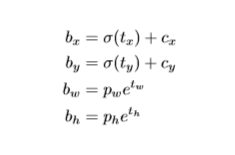


* Để phát hiện được các vật thể nhỏ trên Images thì Yolo v3 đã xử lý bằng cách nhận dạng 3 lần trên 1 Image
* Giả sử với kích thước ảnh đầu vào 416 x 416:
  + Tại lớp chập thứ 82: Bức ảnh được chia thành các ô lưới với kích thước 13×13 ( bức ảnh đã được chia với 32). Tại đây, các ô lưới sẽ có trách nhiệm tìm các vật thể có kích thước lớn trong bức hình.
  + Tại lớp chập thứ 94: Bức ảnh được chia thành các ô lưới với kích thước 26 x 26 ( bức ảnh được chia với 16). Và có trách nhiệm tìm các vật thể có kích thước trung bình.
  + Tương tự, tại lớp chập 106, bức ảnh được chia thành các ô lưới với kích thước 52×52( ảnh được chia với 8) và có trách nhiệm tìm các vật thể có kích thước bé.
  + Bức ảnh sau khi được chia thành các ô lưới, thì được gọi là Downsample images( ảnh giảm mẫu)
* **Ví dụ :**
  + Lấy ví dụ bằng ảnh có kích thước 416 x 416. Lần phát hiện đầu tiên được thực hiện bởi lớp thứ 82. Đối với 81 lớp đầu tiên, hình ảnh được mạng downsampling sao cho lớp thứ 81 có stride là 32. Lúc này, biểu đồ đặc trưng sẽ có kích thước 13 x 13. Một lần phát hiện được tạo ở đây bằng cách sử dụng kernel phát hiện 1 x 1, tạo nên biểu đồ đặc trưng phát hiện có shape là 13 x 13 x 255.
  + Sau đó, ô (trên hình ảnh đầu vào) chứa tâm của hộp ground truth của một đối tượng được chọn làm ô chịu trách nhiệm dự đoán đối tượng. Trong ảnh trên, nó là ô được đánh dấu màu đỏ, chứa trung tâm của hộp ground truth (được đánh dấu màu vàng). Bây giờ, ô này có thể dự đoán ba hộp giới hạn. Cái nào sẽ được gán cho nhãn ground truth? Để hiểu được điều đó, chúng ta phải tìm hiểu khái niệm neo (anchors) ở phần sau.
  + Tiếp theo, feature map từ lớp 79 phải được tích chập một vài lớp trước khi upsampling 2 lần đến kích thước 26 x 26. Feature map này sau đó được ghép độ sâu (depth) với feature map từ lớp 61. Sau đó, feature map mới được ghép nối phải được tích chập một vài lớp 1 x 1 để hợp nhất các đặc trưng từ lớp trước đó. Sau đó, phát hiện thứ hai được thực hiện bởi lớp thứ 94, tạo nên feature map có shape là 26 x 26 x 255.
  + Cuối cùng, quy trình tương tự lại được thực hiện một lần nữa, trong đó feature map từ lớp 91 phải được tích chập một vài lớp trước khi được nối theo độ sâu với feature map từ lớp 36. Giống như trên, một vài lớp tích chập 1 x 1 theo sau để kết hợp thông tin từ trước lớp (36). Sau đó, phát hiện cuối cùng ở lớp thứ 106, tạo ra feature map có kích shape là 52 x 52 x 255.
  + Kích thước 13 x 13 chịu trách nhiệm phát hiện các vật thể lớn, trong khi kích thước 52 x 52 phát hiện các vật thể nhỏ, với kích thước 26 x 26 phát hiện các vật thể trung bình. Khắc phục nhược điểm của các phiên bản YOLO trước đó.



* Kích thước của đầu ra là :S x S x 255
  + S lần lượt được gắn với các giá trị:13,26,52. Và
  + Giá trị 255 = B x (5+C) với B = 3 là số lượng bounding box, và C =80 là số lớp vật thể.
  + Mỗi bounding box chứa 5 giá trị bao gồm: x , y , w , h và p( xác suất ô lưới có chứa vật thể)
  + Để tính ra kết quả cuối cùng thì mỗi bounding box sẽ nhân giá trị P lần lượt với P1,P2,…Pc ( là xác suất ô lưới đó chứa các vật thể cụ thể) và lựa chọn ra kết quả cao nhất.
  + Với mỗi bức ảnh được nhận dạng, thì YOLOv3 sẽ sinh ra (13 x 13 + 26 x 26 + 52 x 52) x3 = 10,647 bounding box. Từ đó sẽ tính xác suất và xuất ra kết quả là những vật thể có xác suất **cao nhất**
* **Bounding box predict :**





* + , , , lần lượt là tọa độ tâm (x, y) và chiều rộng , chiều cao bounding box dự đoán .
  + , , , là đầu ra của mạng
  + , là tọa độ trên cùng bên trái của lưới ảnh . , là kích thước hộp neo .
  + Yolo v3 dự đoán tâm thông qua một hàm **sigmoid =>** Bắt buộc giá trị đầu ra phải nằm trong khoảng (0 , 1) (để nằm trong phạm vi 1 ô) . Sau đó ta thêm các tọa độ trên cùng bên trái , để dự đoán tọa độ thực của đối tượng
  + Kích thước của hộp giới hạn được dự đoán bằng cách áp dụng phép biến đổi không gian logarit cho đầu ra sau đó **nhân** với kích thước anchor box . Trong kết quả của dự đoán , được chuẩn hóa theo chiều cao và chiều rộng của hình ảnh
    - Đối với objectness score : Trong yolo v3 đưa ra điểm cho các đối tượng cho mỗi hộp giới hạn . Nó sử dụng hồi quy logistic để dự đoán điểm đối tượng để mang tính khách quan . Nó sử dụng hàm kích hoạt **leaky ReLU**
    - Đối với class score :Yolo v3 không sử dụng **softmax** thay vào đó sử dụng  **logistic classifier** rời rạc (trong lúc huấn luyện sử dụng hàm mất mát binary cross - entropy) => dùng để phân biệt các đối tượng bao nhau ví dụ như con người và đàn ông **ngược lại** nếu sử dụng **softmax** cho dự đoán chỉ giả định rằng mỗi hộp có đúng 1 lớp , điều này thường không đúng .
* **Anchor box :**
  + Anchor box thực ra là các bounding box nhưng được tạo sẵn ( chứ không phải kết quả của quá trình nhận dạng – prediction).
  + Với mỗi ô lưới sẽ 9 anchor box khác nhau với kích thước:
    - Ô lưới 13×13: (116×90) , (156 × 198) , (373 × 32)
    - Ô lưới 26×26: (30×61) , (62×45) , (59 × 119).
    - Ô lưới 52×52: (10×13) , (16×30) , (33 x 23).
  + Hộp giới hạn chịu trách nhiệm phát hiện đối tượng sẽ là hộp neo có **IoU** cao nhất với hộp ground truth. Trong quá trình đào tạo sẽ kết hợp với thuật toán **Kmean cluster** (phân cụm). Cùng với ground truth để tính ra sai sót giữa ground truth và anchor box bằng cách điều chỉnh các giá trị x,y,w,h, từ đó học được các đặc điểm của vật thể.