

BÁO CÁO ĐỒ ÁN

1. Link Source Code và báo cáo
<https://github.com/buingochung1983/VRA.BuiNgocHung/tree/master/DO%20AN>
2. Link Video Kết quả chạy demo
<https://github.com/buingochung1983/VRA.BuiNgocHung/blob/master/DO%20AN/D%20EMO.rar>
3. Link Data Train
 - Chứa trongCODE\result\data\obj\
4. Link Video chạy demo
<https://github.com/buingochung1983/VRA.BuiNgocHung/blob/master/DO%20AN/T%20EST.rar>
5. Mục tiêu, nội dung và cách thực hiện
 - 5.1 . Mục tiêu
 - Detect được bảng số xuất hiện trong video
 - 5.2 . Nội dung
 - Dùng machineLearning để detect bảng số xuất hiện trong video
 - 5.3 . Cách thực hiện
 - 5.3.1 . Chuẩn bị Data train
 - Là bộ hình có chứa bảng số
 - film chạy demo
 - 5.3.2 . Thực hiện

5.3.2.1 Mô tả mạng YOLO

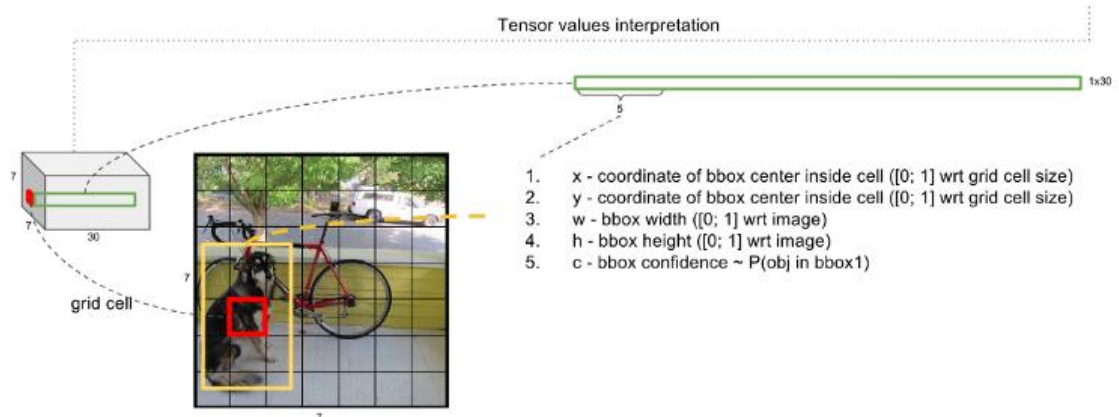
Kiến trúc mạng YOLO:

YOLO sử dụng VGG16 cho tầng rút trích đặc trưng:

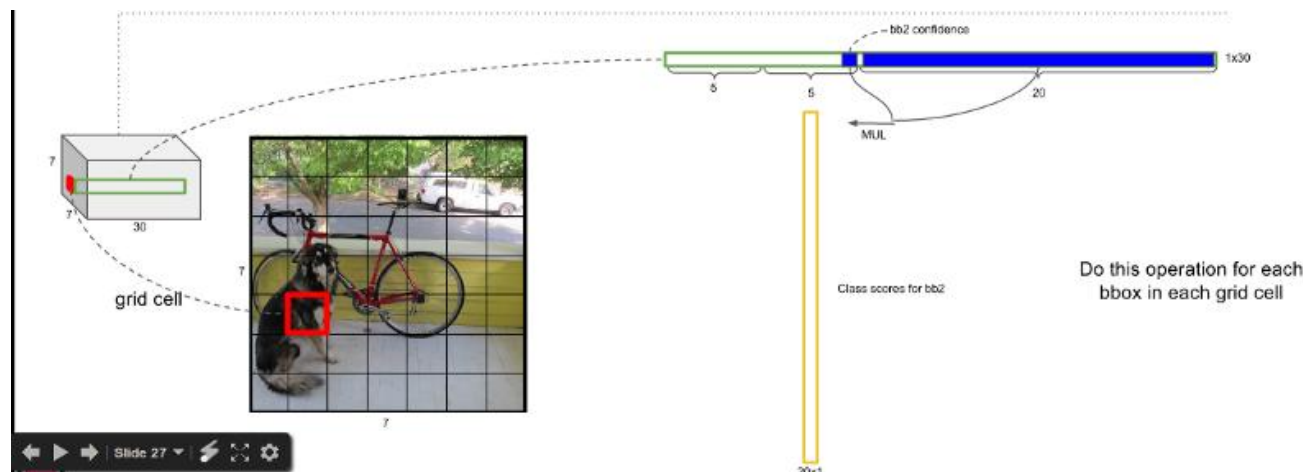


Input của mạng YOLO là: ảnh 3 kênh màu, cùng với file anotation, dùng để label cho đối tượng trong ảnh.

Tầng output (boxes) của YOLO chứa thông tin như sau:



Output của YOLO trả về một vector 1x30, trong đó 10 giá trị đầu là tọa độ x, y, chiều dài chiều rộng của đối cell (tương ứng 5 hệ số cho mỗi bounding box, bởi vì mỗi grid cell sẽ có 2 bounding box) và hệ số tin tưởng, hệ số tin tưởng này tính toán dựa trên độ overlap của detect được và vùng thực sự của đối tượng, 20 giá trị còn lại chính là score của class.



5.3.2.2 Train YOLO với custom data:

tạo file train.txt trong thư mục build\darknet\x64\data\, ví dụ nội dung file train.txt

```
data/obj/img1.jpg
data/obj/img2.jpg
data/obj/img3.jpg
```

tải pretrained model về, làm init cho các layers.

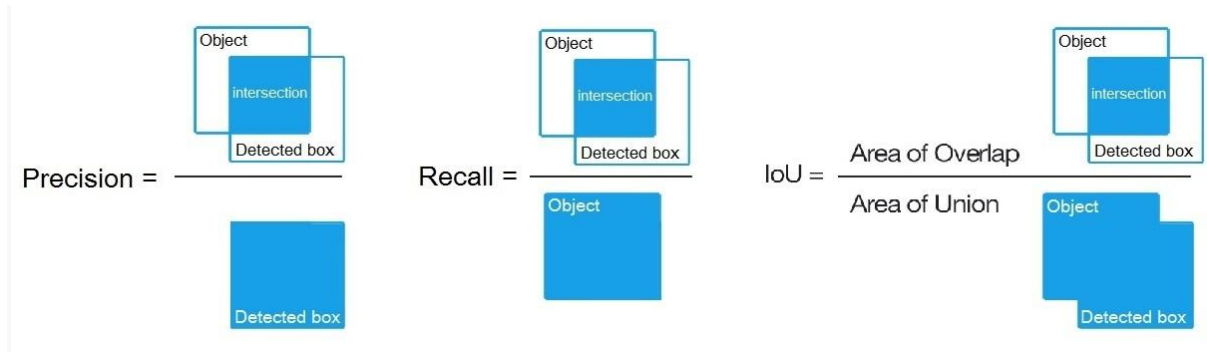
Tập validate dùng để verify chất lượng training, trong quá trình train.

Tập validate tổ chức giống như tập training.

Các chỉ số của YOLO

IOU: càng lớn càng tốt, chỉ về độ chính xác.

Recall: lớn, cũng nói về độ chính xác.



Hàm chính của YOLO:

File main là file src/yolo.c, chương trình bắt đầu từ file này, trong YOLO có hỗ trợ sử dụng cuda trong quá trình training,

Hàm main: void run_yolo(yolo.c) -> train_network(network.c) -

>train_network_datum(network.c), tại đây sẽ có hàm forward và backward cho mạng

Download pretrained mode của YOLO đã train sẵn để làm file init weight cho mạng, chép file này chứa trong thư mục result/data. Link download file pretrained model:

http://www.mediafire.com/file/qh8mjb7hs6o2o5g/darknet19_448.conv.23

Command line để train:

Train:

```
darknet.exe detector train data/obj.data yolo_2.cfg data/darknet19_448.conv.23
```

test với video:

download file yolo_2_5000.weights, đã train với 5000 vòng, để test model, chép file này vào trong thư mục result/data

link download yolo_2_5000.weights:

http://www.mediafire.com/file/12k9ff4kxixmzh9/yolo_2_5000.weights.

```
darknet.exe detector demo data/obj.data yolo_2.cfg data/yolo_2_5000.weights
test.mp4 -i 0 -out_filename res.avi
```

Tạo file cfg để config cho mạng, bao gồm các tham số sử dụng như filter cho convolution, số lượng class đầu vào, đường dẫn đến data và một số tham số khác.

Ví dụ:

```
[convolutional]
filters=35

[region]
classes=1
```

tạo file obj.names để mapping tên của object, mỗi object là một dòng, file này đặt trong build\darknet\x64\data\,

Tạo file obj.data để chứa thông tin về file txt cho training, đặt trong build\darknet\x64\data\, **classes = số lượng đối tượng**

```
classes= 1
train  = data/train.txt
valid  = data/test.txt
names = data/obj.names
backup = backup/ // nơi chứa file trọng số - weight
đặt file ảnh(data) trong build\darknet\x64\data\obj\
```

tạo file train.txt,

dùng đoạn code python list.py để liệt kê các file jpg trong thư mục data obj.

chỉnh lại đường dẫn trong đến thư mục chứa data, và đường dẫn đến thư mục lưu file train.txt trong source code:

```
path = "C:\\path_To_data \\darknet-master\\build\\darknet\\x64\\data\\obj"
```

```
train = "C:\\ path_To_data \\darknet-master\\build\\darknet\\x64\\data\\train.txt"
```

tạo file txt cùng tên và cùng thư mục với file image, với thông tin <object-class> <x>
<y> <width> <height>

<object-class> - số mapping của class, là số nguyên

<x> <y> <width> <height> - giá trị thực, có thể mang giá trị từ 0-1.0

Ví dụ: <x> = <absolute_x> / <image_width> hoặc <height> = <absolute_height> / <image_height>

attention: <x> <y> - tâm của window

sử dụng /BBox-Label-Tool để tạo file anotaion cho object trong ảnh,
(<https://github.com/puzzledqs/BBox-Label-Tool>)

sử dụng file python convert.py để convert file anotation sang chuẩn của YOLO.

*** tất cả nguồn ảnh được lấy từ: <https://medium.com/diaryofawannapreneur/yolo-you-only-look-once-for-object-detection-explained-6f80ea7aaa1e>