THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

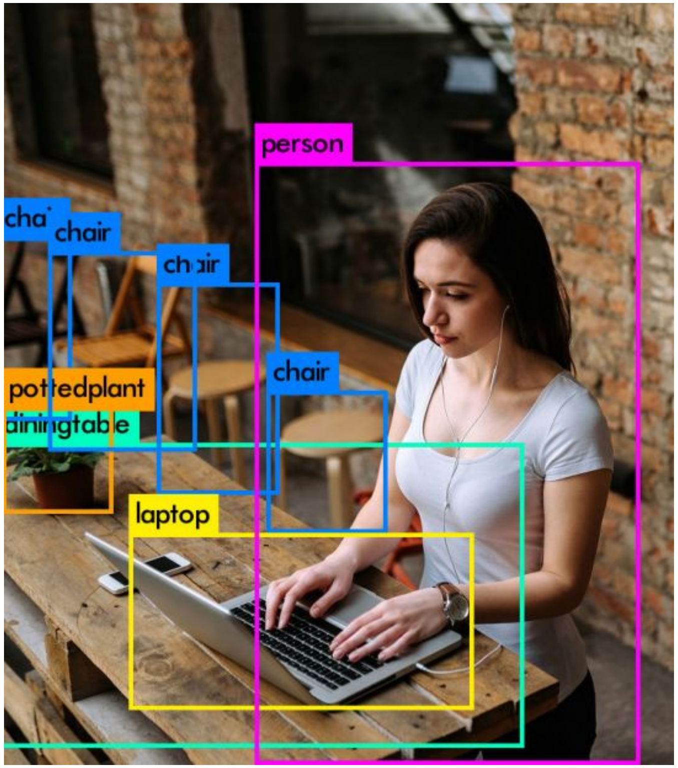
Final Project Group 7 Object Detection Using YOLO

Hao Heng, Ze Gong, Junhe Zhang

Introduction

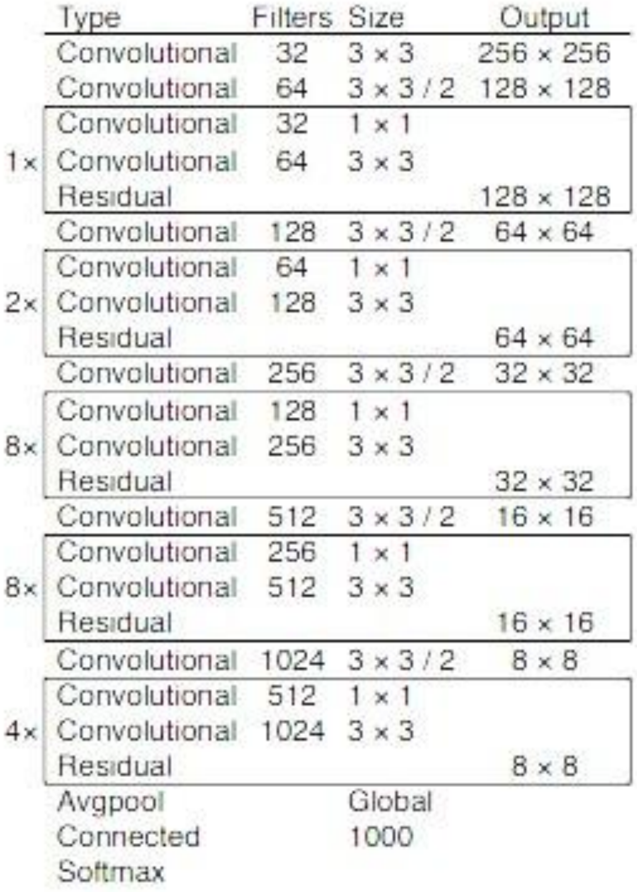
- YOLO
- Create custom dataset
- Train YOLO network
- Result and Conclusion

What is Yolo?



- You only look once (YOLO) is a state-of-the-art, real-time object detection system

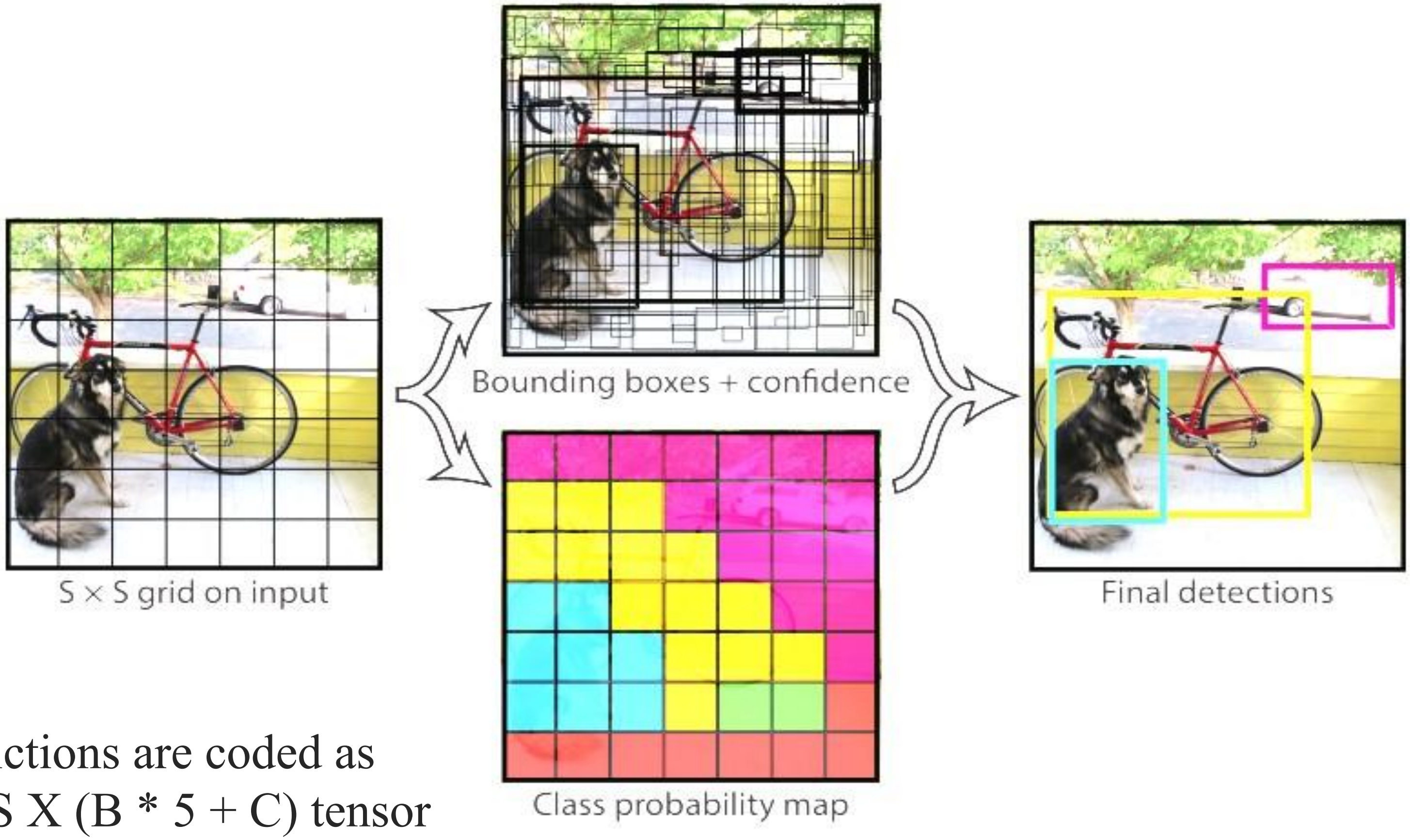
YOLO Network



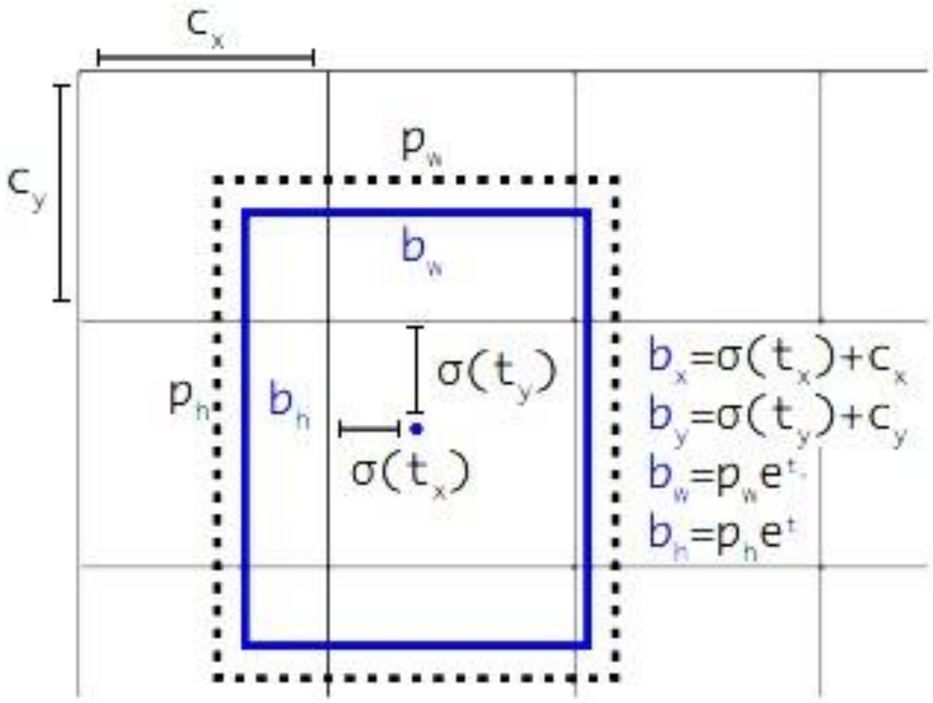
	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

In this project we use YOLO (You Only Look Once) v3.0 for the purpose of Object Detection.

YOLO



Predictions are coded as
S X S X (B * 5 + C) tensor



Calculation of the bounding box coordinates.

THE GEORGE
WASHINGTON
UNIVERSITY

WASHINGTON, DC

Create custom dataset

Dataset

Source: Google Open Image Dataset

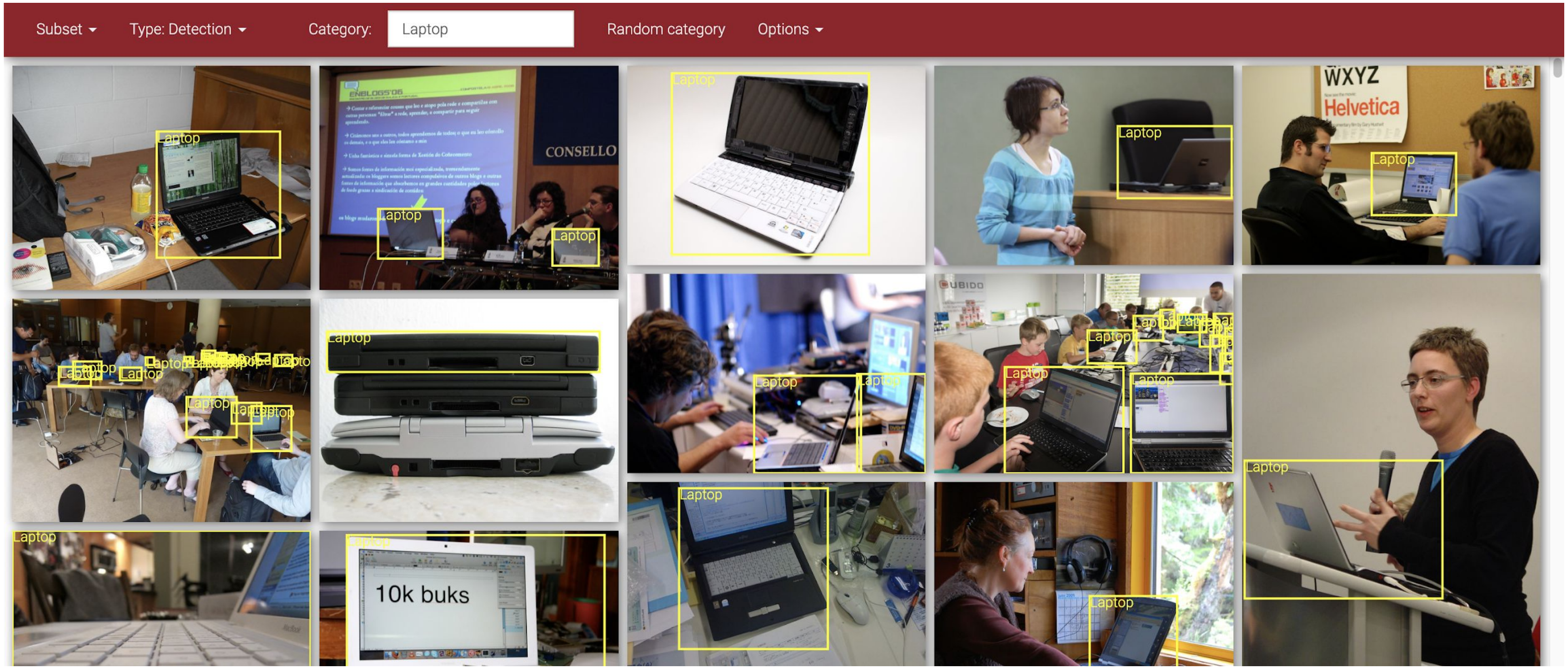
link:https://storage.googleapis.com/openimages/web/index.html

Downloader: OIDv4_ToolKit

Category: Car, Bicycle wheel, Traffic light, Person, Jeans, Laptop, Bus

Instance of each category: 1000

Dataset



Type: Detection

Category: Laptop

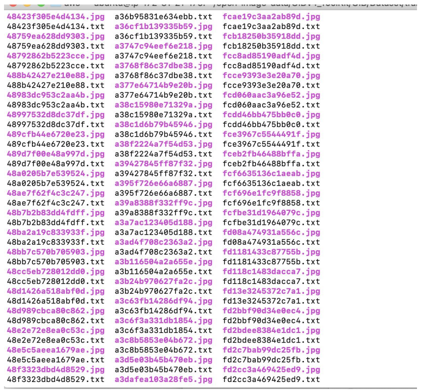
Data preprocessing

Needed files for darknet framework:

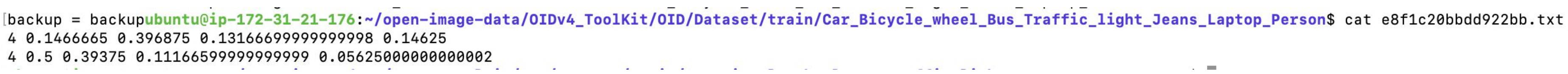
- annotation for each image
- train.txt
- test.txt
- data.data
- classes.names

Data preprocessing

Annotation.txt (each image has its own .txt):
class number
center x
center y
width
height



```
48423f305e4d4134.jpg  a36b95831e634ebb.txt  fcae19c3aa2ab89d.jpg
48423f305e4d4134.txt  a36cf1b139335b59.jpg  fcae19c3aa2ab89d.txt
48759ea628dd9303.jpg  a36cf1b139335b59.txt  fcb18250b35918dd.jpg
48759ea628dd9303.txt  a3747c94eef6e218.jpg  fcb18250b35918dd.txt
48792862b5223cce.jpg  a3747c94eef6e218.txt  fcc8ad85190adf4d.jpg
48792862b5223cce.txt  a3768f86c37dbe38.jpg  fcc8ad85190adf4d.txt
488b42427e210e88.jpg  a3768f86c37dbe38.txt  fcce9393e3e20a70.jpg
488b42427e210e88.txt  a377e64714b9e20b.jpg  fcce9393e3e20a70.txt
48983dc953c2aa4b.jpg  a377e64714b9e20b.txt  fcd060aac3a96e52.jpg
48983dc953c2aa4b.txt  a38c15980e71329a.jpg  fcd060aac3a96e52.txt
48997532d8dc37df.jpg  a38c15980e71329a.txt  fcdd46bb475bb0c0.jpg
48997532d8dc37df.txt  a38c1d6b79b45946.jpg  fcdd46bb475bb0c0.txt
489cfb44e6720e23.jpg  a38c1d6b79b45946.txt  fce3967c5544491f.jpg
489cfb44e6720e23.txt  a38f2224a7f54d53.jpg  fce3967c5544491f.txt
489d7f00e48a997d.jpg  a38f2224a7f54d53.txt  fceb2fb46488bffa.jpg
489d7f00e48a997d.txt  a39427845ff87f32.jpg  fceb2fb46488bffa.txt
48a0205b7e539524.jpg  a39427845ff87f32.txt  fcf6635136c1aeab.jpg
48a0205b7e539524.txt  a395f726e66a6887.jpg  fcf6635136c1aeab.txt
48ae7f62f4c3c247.jpg  a395f726e66a6887.txt  fcf696e1fc9f8858.jpg
48ae7f62f4c3c247.txt  a39a8388f332ff9c.jpg  fcf696e1fc9f8858.txt
48b7b2b83dd4fdff.jpg  a39a8388f332ff9c.txt  fcfbe31d1964079c.jpg
48b7b2b83dd4fdff.txt  a3a7ac123405d188.jpg  fcfbe31d1964079c.txt
48ba2a19c833933f.jpg  a3a7ac123405d188.txt  fd08a474931a556c.jpg
48ba2a19c833933f.txt  a3ad4f708c2363a2.jpg  fd08a474931a556c.txt
48bb7c570b705903.jpg  a3ad4f708c2363a2.txt  fd1181433c87755b.jpg
48bb7c570b705903.txt  a3b116504a2a655e.jpg  fd1181433c87755b.txt
48cc5eb728012dd0.jpg  a3b116504a2a655e.txt  fd118c1483dacca7.jpg
48cc5eb728012dd0.txt  a3b24b970627fa2c.jpg  fd118c1483dacca7.txt
48d1426a518abf0d.jpg  a3b24b970627fa2c.txt  fd13e3245372c7a1.jpg
48d1426a518abf0d.txt  a3c63fb14286df94.jpg  fd13e3245372c7a1.txt
48d989cbca80c862.jpg  a3c63fb14286df94.txt  fd2bbf90d34e0ec4.jpg
48d989cbca80c862.txt  a3c6f3a331db1854.jpg  fd2bbf90d34e0ec4.txt
48e2e72e8ea0c53c.jpg  a3c6f3a331db1854.txt  fd2bdee8384e1dc1.jpg
48e2e72e8ea0c53c.txt  a3c8b5853e04b672.jpg  fd2bdee8384e1dc1.txt
48e5c5aeea1679ae.jpg  a3c8b5853e04b672.txt  fd2c7bab99dc25fb.jpg
48e5c5aeea1679ae.txt  a3d5e03b45b470eb.jpg  fd2c7bab99dc25fb.txt
48f3323dbd4d8529.jpg  a3d5e03b45b470eb.txt  fd2cc3a469425ed9.jpg
48f3323dbd4d8529.txt  a3dafea103a28fe5.jpg  fd2cc3a469425ed9.txt
```



```
[backup = backupubuntu@ip-172-31-21-176:~/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person$ cat e8f1c20bbdd922bb.txt
 4 0.1466665 0.396875 0.13166699999999998 0.14625
 4 0.5 0.39375 0.11166599999999999 0.05625000000000002
```

Data preprocessing

train.txt:

/full path/image1.jpg

/full path/image2.jpg

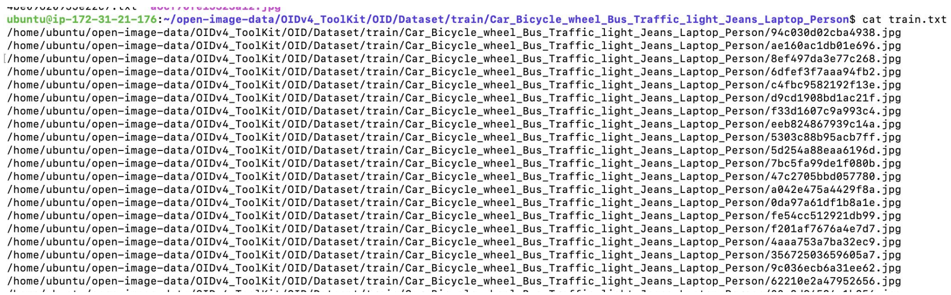
…..

test.txt:

/full path/image1.jpg

/full path/image2.jpg

…..



```
ubuntu@ip-172-31-21-176:~/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person$ cat train.txt
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/94c030d02cba4938.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/ae160ac1db01e696.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/8ef497da3e77c268.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/6dfef3f7aaa94fb2.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/c4fbc9582192f13e.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/d9cd1908bd1ac21f.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/f33d1607c9a993c4.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/eeb824867939c14a.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/5303c88b95acb7ff.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/5d254a88eaa6196d.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/7bc5fa99de1f080b.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/47c2705bbd057780.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/a042e475a4429f8a.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/0da97a61df1b8a1e.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/fe54cc512921db99.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/f201af7676a4e7d7.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/4aaa753a7ba32ec9.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/35672503659605a7.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/9c036ecb6a31ee62.jpg
/home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/62210e2a47952656.jpg
```

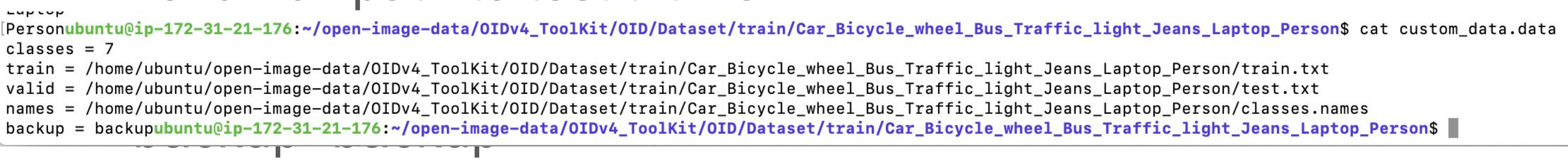
Data preprocessing

data.data:

classes=the number of classes in the dataset

train=full path to train.txt file

valid=full path to test.txt file



```
Personubuntu@ip-172-31-21-176:~/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person$ cat custom_data.data
classes = 7
train = /home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/train.txt
valid = /home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/test.txt
names = /home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/classes.names
backup = backupubuntu@ip-172-31-21-176:~/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person$
```

backup=backup

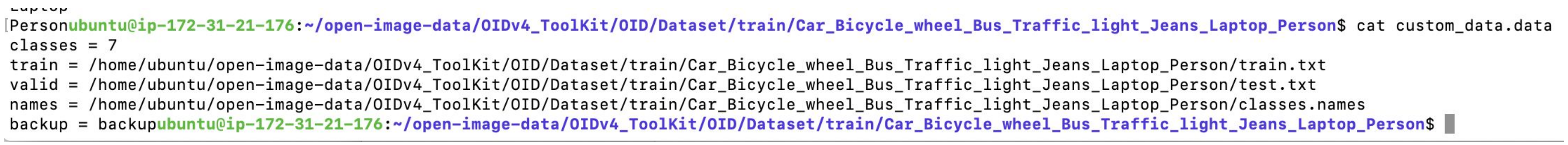
classes.names:

Car

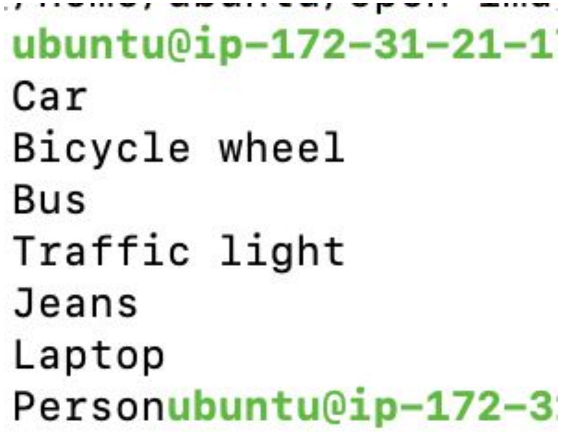
Bicycle wheel

.....

Data preprocessing



```
Laptop
Personubuntu@ip-172-31-21-176:~/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person$ cat custom_data.data
classes = 7
train = /home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/train.txt
valid = /home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/test.txt
names = /home/ubuntu/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person/classes.names
backup = backupubuntu@ip-172-31-21-176:~/open-image-data/OIDv4_ToolKit/OID/Dataset/train/Car_Bicycle_wheel_Bus_Traffic_light_Jeans_Laptop_Person$
```



```
/home/ubuntu/open-ima
ubuntu@ip-172-31-21-1
Car
Bicycle wheel
Bus
Traffic light
Jeans
Laptop
Personubuntu@ip-172-3
```

THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

Training YOLO in Darknet Framework

Setting up Configuration

Classes = 7

filters = (7 + 5) x 3 = 36

max_batches = 7 x 2000 = 14000

steps = 0.8 x max_batches / 0.9 x max_batches

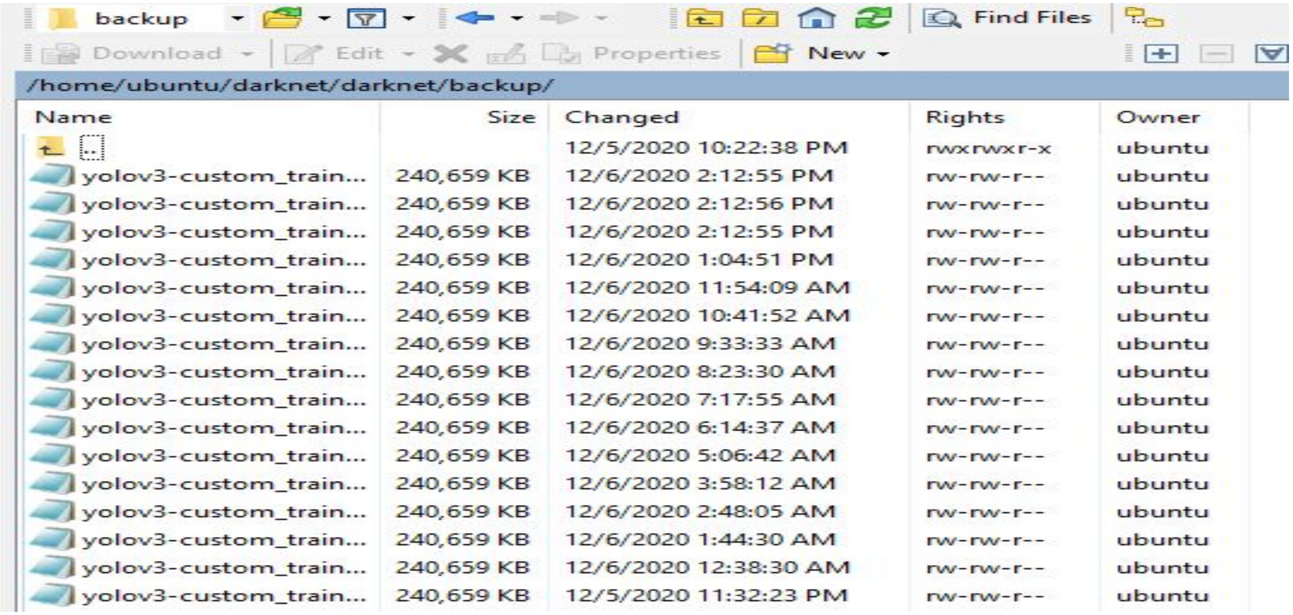
= 11200 / 12600

batch = 32

subdivisions = 8

minibatch = batch/subdivisions = 4

Result of Training Process

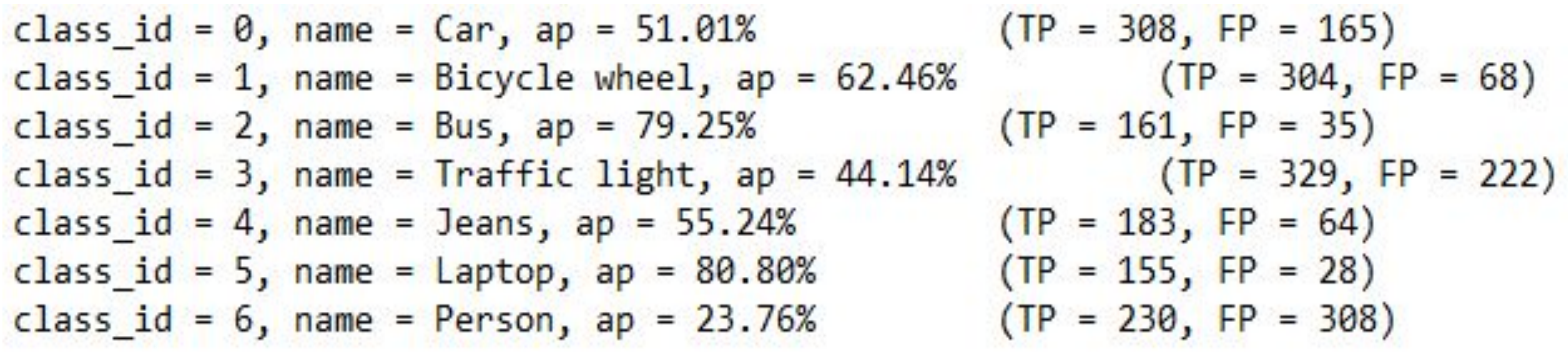


/home/ubuntu/darknet/darknet/backup/

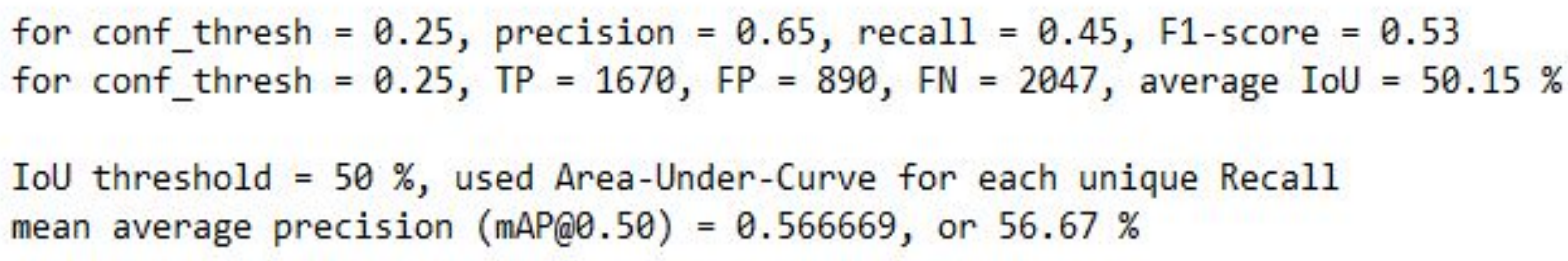
Name	Size	Changed	Rights	Owner
..		12/5/2020 10:22:38 PM	rwxrwxr-x	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 2:12:55 PM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 2:12:56 PM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 2:12:55 PM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 1:04:51 PM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 11:54:09 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 10:41:52 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 9:33:33 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 8:23:30 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 7:17:55 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 6:14:37 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 5:06:42 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 3:58:12 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 2:48:05 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 1:44:30 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/6/2020 12:38:30 AM	rw-rw-r--	ubuntu
yolov3-custom_train...	240,659 KB	12/5/2020 11:32:23 PM	rw-rw-r--	ubuntu

THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

Mean Average Precision Example:



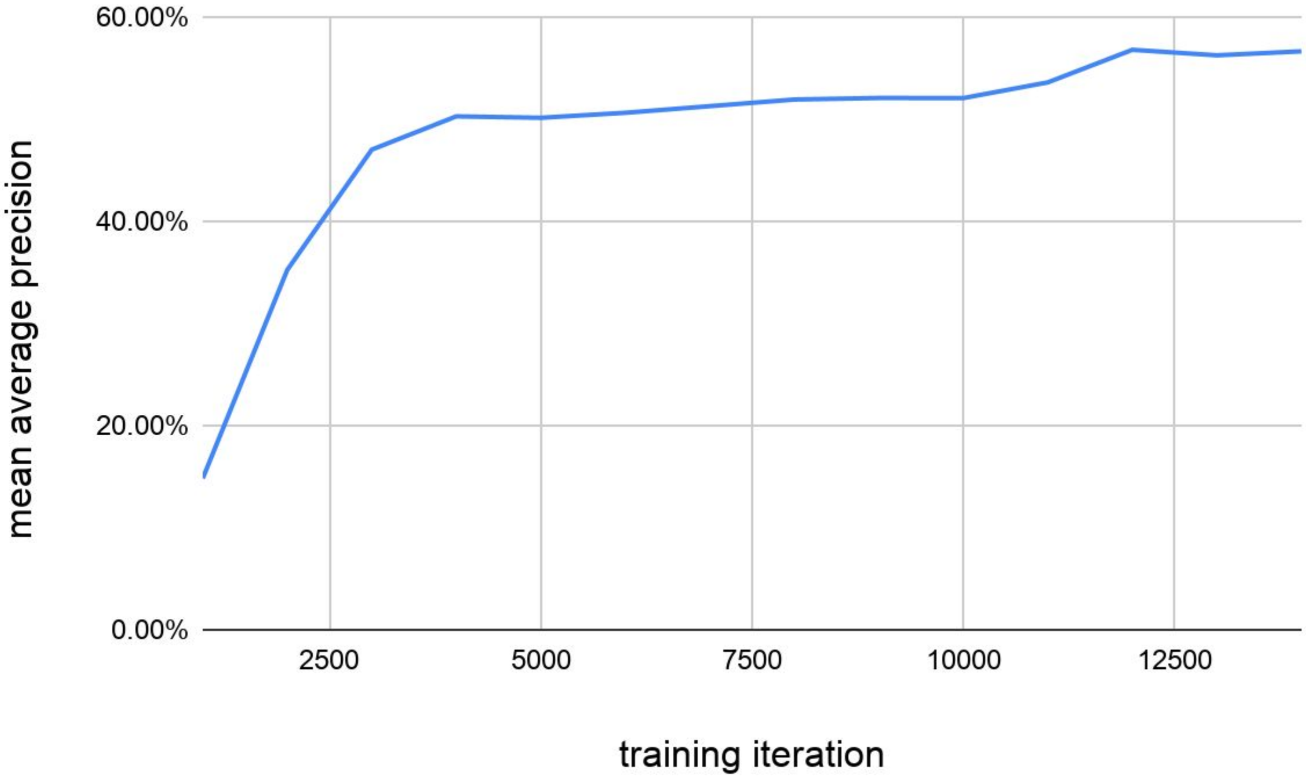
```
class_id = 0, name = Car, ap = 51.01%           (TP = 308, FP = 165)
class_id = 1, name = Bicycle wheel, ap = 62.46%         (TP = 304, FP = 68)
class_id = 2, name = Bus, ap = 79.25%           (TP = 161, FP = 35)
class_id = 3, name = Traffic light, ap = 44.14%         (TP = 329, FP = 222)
class_id = 4, name = Jeans, ap = 55.24%         (TP = 183, FP = 64)
class_id = 5, name = Laptop, ap = 80.80%        (TP = 155, FP = 28)
class_id = 6, name = Person, ap = 23.76%        (TP = 230, FP = 308)
```



```
 for conf_thresh = 0.25, precision = 0.65, recall = 0.45, F1-score = 0.53
 for conf_thresh = 0.25, TP = 1670, FP = 890, FN = 2047, average IoU = 50.15 %

 IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
 mean average precision (mAP@0.50) = 0.566669, or 56.67 %
```

Performance of Different Training Weights



The best mAP is 56.82%, appears at iteration 12000.

THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

Apply The Best Model on Testing image and video

Loading Best Model

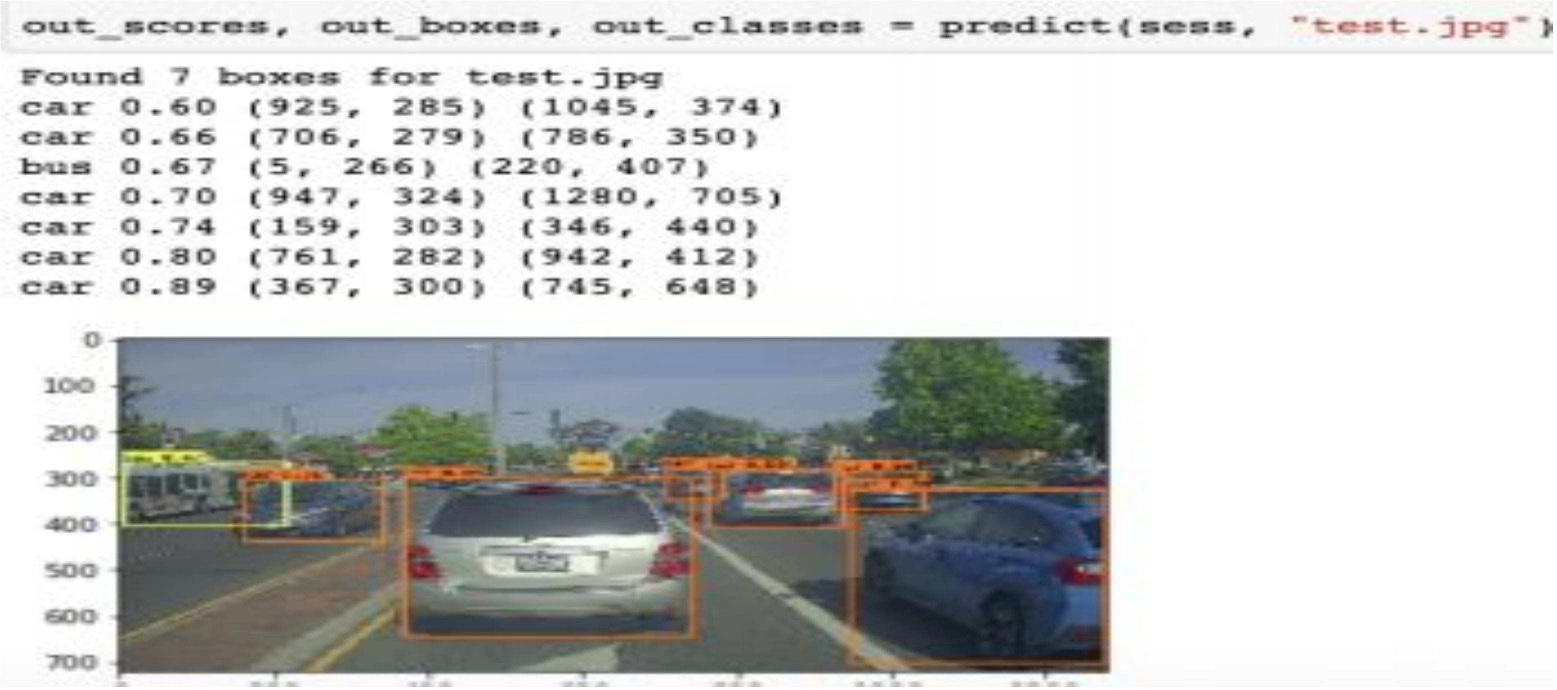
There are two different ways to do object detection:

- Using darknet framework’s command line
    - ‘./darknet detector test cfg/custom_data.data cfg/yolov3-custom_train.cfg backup/yolov3-custom_train_last.weights data/laptop-jean-test.jpg -out_filename data/result-laptop-jean-test.jpg -dont_show’
- Run process_image_video.py file
    - python3 process_image_video.py

Algorithm of process_image_video.py

1. Read the input image and convert it into a blob object which is accepted by the YOLO framework.
2. Load the best trained weights and the network structure file into the YOLO framework.
3. Process blob image file into the model and generate output bounding boxes for each object inside the image.
4. Using Non-Maximun_Suppression technique on the output bounding boxes to drop duplicated boxes on the same object with lower confidence.

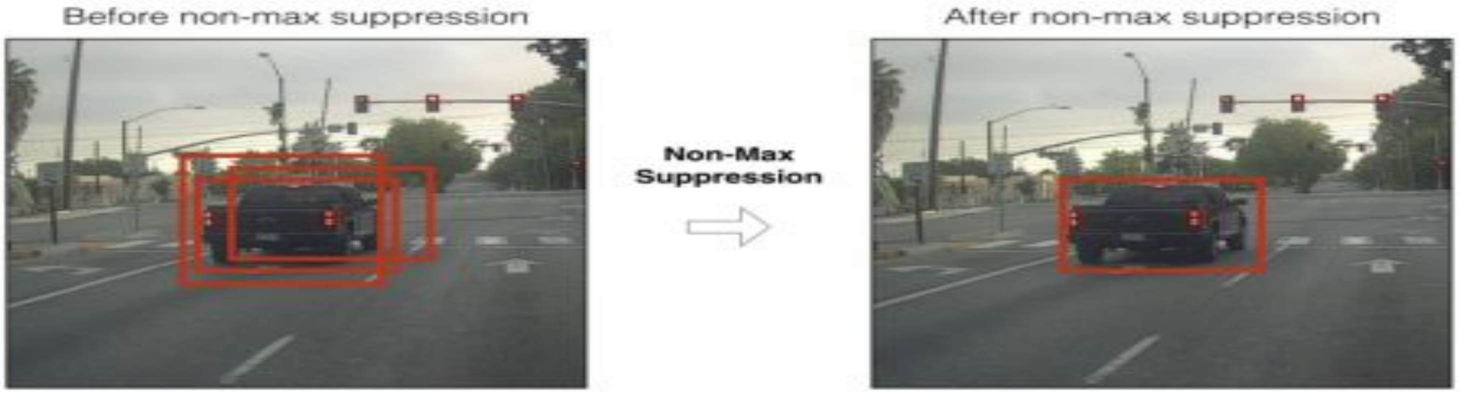
What the Results look like

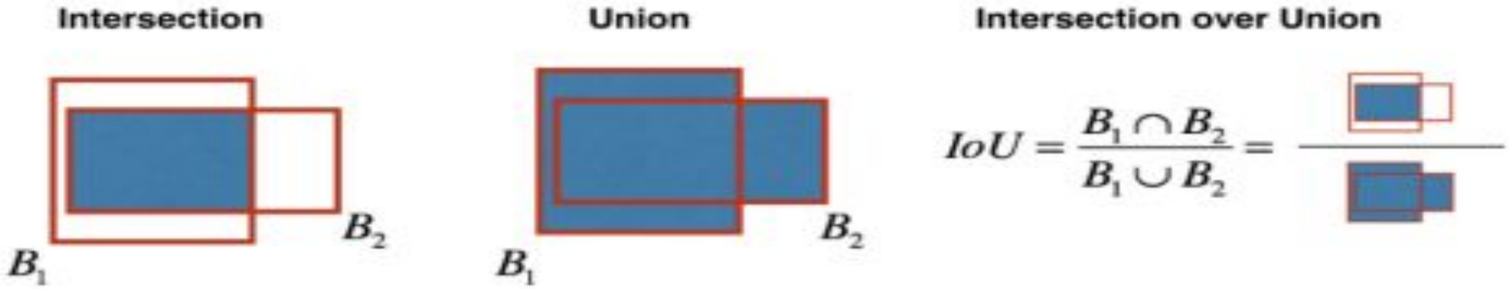


```
out_scores, out_boxes, out_classes = predict(sess, "test.jpg")
```

```
Found 7 boxes for test.jpg
car 0.60 (925, 285) (1045, 374)
car 0.66 (706, 279) (786, 350)
bus 0.67 (5, 266) (220, 407)
car 0.70 (947, 324) (1280, 705)
car 0.74 (159, 303) (346, 440)
car 0.80 (761, 282) (942, 412)
car 0.89 (367, 300) (745, 648)
```

Non-Maximum-Suppression

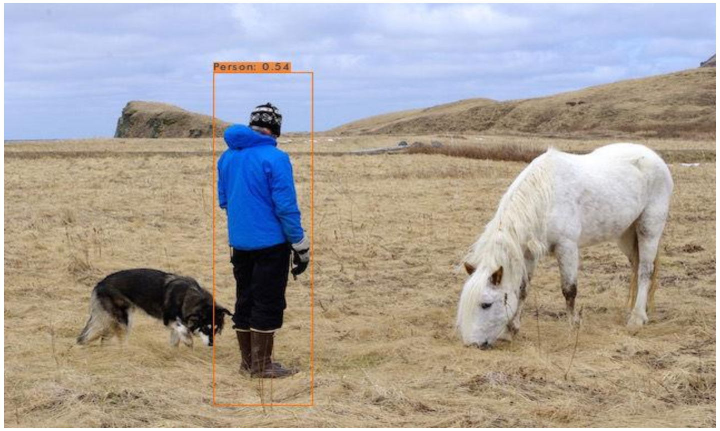




$$IoU = \frac{B_1 \cap B_2}{B_1 \cup B_2} =$$

Results of Object Detected Images

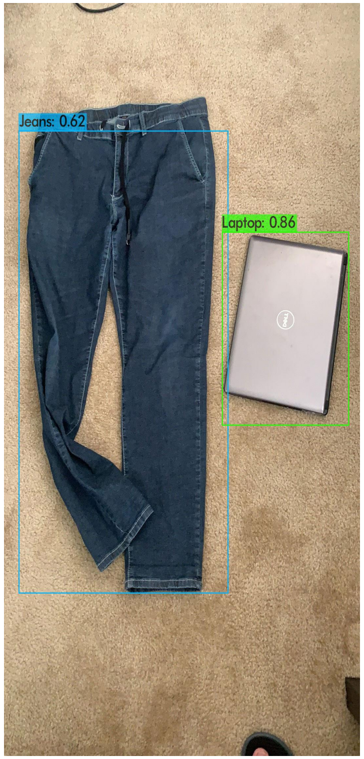
Original Image



Model Detection

Results of Object Detected Images

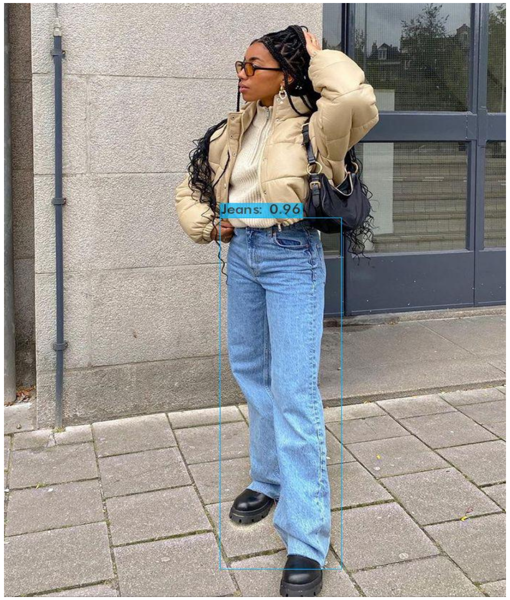
Original Image



Model Detection

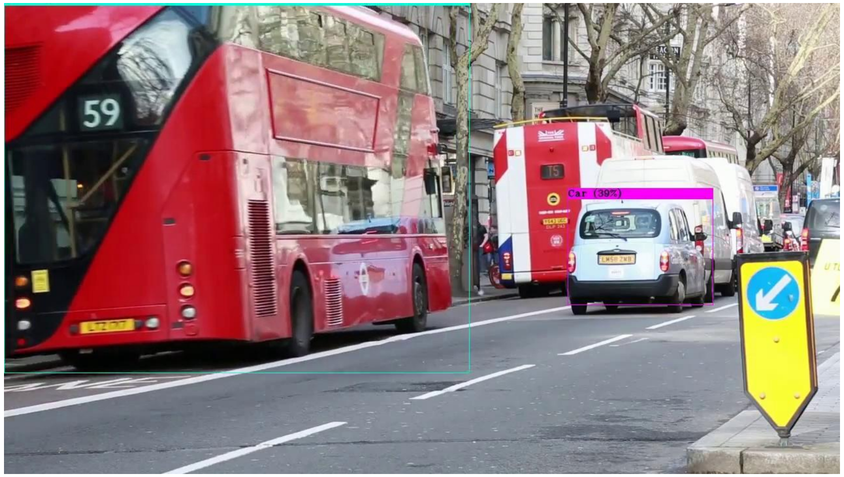
Results of Object Detected Images

Original Image

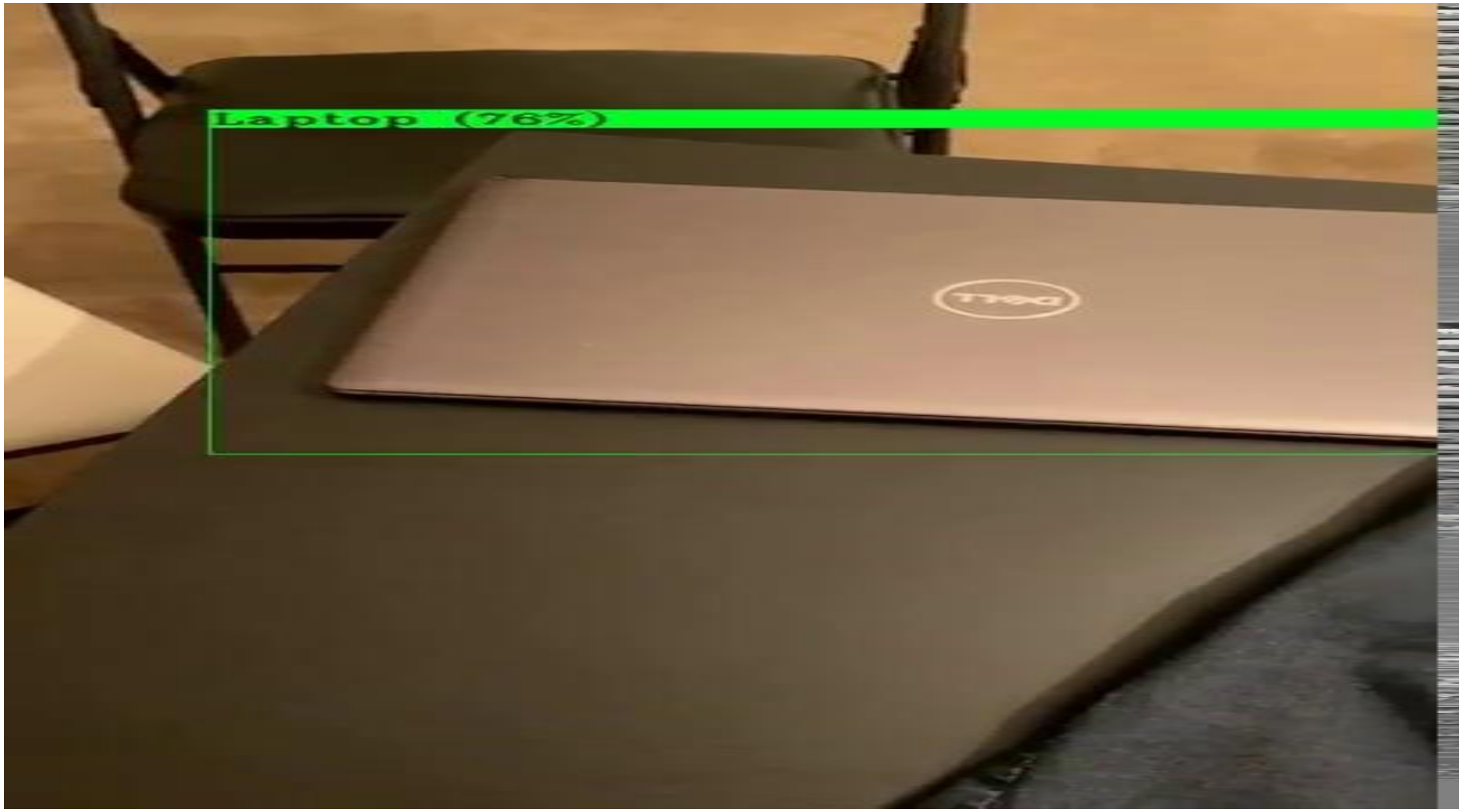


Model Detection

Results of Object Detected Videos



https://drive.google.com/file/d/1luAp5ggEG8hvPBlzG8yjvvBXh1PIXClW/view?usp=sharing



https://drive.google.com/file/d/1xc2RdESogHPag3VwwnbuNQs-RX--mZzF/view?usp=sharing

THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

Limitation and Conclusion

Limitations to YOLO

In spite of the high accuracy numbers and almost perfect bounding boxes, there are a few limitations to this object detection methods:

- Sometimes fails to detect overlapping objects, and objects that are partially visible in the frame.
- Detects the object class falsely if the features are a little blurred.
- It is very sensitive to model overfitting.

Conclusion and discussion

- The number of iteration of training process is not long enough to predict all categories we choose. (2000 iteration for each class)
- Our model is under-trained, compared to coco weight. Our dataset is not big enough to train a very accurate model
- Our model performs not well at detecting bus and person
- Our model has some difficult to detect object when there are multiple classes are overlapped. It will only detect one class.

References

[1]Joseph Redmon, Santosh Divvala, Ross Girshick, ALi Farhadi “You Only Look Once: Unified, Real-Time Object Detection”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[2]Open Images Dataset V6 + Extensions. (n.d.). Retrieved December 08, 2020, from https://storage.googleapis.com/openimages/web/index.html

[3]Joseph R. . Darknet: Open Source Neural Networks in C. Retrieved (2013-2016) from http://pjreddie.com/darknet/

[4]Valentyn S. Training YOLO v3 for Objects Detection with Custom Data.
Retrieved from https://storage.googleapis.com/openimages/web/index.html

Thank you!