```
# Importing required packages
# tesnsorflow, numpy, pandas
# cv2 for drawing rectangle on images
# tdqm for showing progress bar
# shutil for copying images


import os, time, random
import numpy as np
import pandas as pd
import cv2, torch
from tqdm.auto import tqdm
import shutil as sh

from IPython.display import Image, clear_output
import matplotlib.pyplot as plt
%matplotlib inline


from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
%%time

!git clone https://github.com/ultralytics/yolov5  # clone repo
!pip install -U pycocotools
!pip install -qr yolov5/requirements.txt  # install dependencies
!cp yolov5/requirements.txt ./
```

```
Cloning into 'yolov5'...
remote: Enumerating objects: 15529, done.
remote: Counting objects: 100% (136/136), done.
remote: Compressing objects: 100% (93/93), done.
remote: Total 15529 (delta 49), reused 118 (delta 43), pack-reused 15393
Receiving objects: 100% (15529/15529), 14.59 MiB | 31.05 MiB/s, done.
Resolving deltas: 100% (10573/10573), done.
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: pycocotools in /usr/local/lib/python3.9/dist-packages (2.0.6)
Requirement already satisfied: matplotlib>=2.1.0 in /usr/local/lib/python3.9/dist-packages (from pycocotools) (3.7.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.9/dist-packages (from pycocotools) (1.22.4)
Requirement already satisfied: importlib-resources>=3.2.0 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycoco
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycocotools) (1.
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycocotools) (1
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycocotools) (3.
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycocotools) (0.11.0
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycocotools) (4
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycocotools)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycocotools) (8.4.0
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.9/dist-packages (from matplotlib>=2.1.0->pycocotools) (23.
Requirement already satisfied: zipp>=3.1.0 in /usr/local/lib/python3.9/dist-packages (from importlib-resources>=3.2.0->matplotlib>=
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.9/dist-packages (from python-dateutil>=2.7->matplotlib>=2.1.0->py
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 184.3/184.3 kB 7.5 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 62.7/62.7 kB 6.7 MB/s eta 0:00:00
CPU times: user 107 ms, sys: 53.8 ms, total: 161 ms
Wall time: 12.2 s
```

```
# Loading Data / Preprocessing

img_h, img_w, num_channels = (380, 676, 3)
df = pd.read_csv('/content/drive/MyDrive/cvd_practical/CVD/data/train_solution_bounding_boxes (1).csv')
df.rename(columns={'image':'image_id'}, inplace=True)
df['image_id'] = df['image_id'].apply(lambda x: x.split('.')[0])
df['x_center'] = (df['xmin'] + df['xmax'])/2
df['y_center'] = (df['ymin'] + df['ymax'])/2
df['w'] = df['xmax'] - df['xmin']
df['h'] = df['ymax'] - df['ymin']
df['classes'] = 0
df['x_center'] = df['x_center']/img_w
df['w'] = df['w']/img_w
df['y_center'] = df['y_center']/img_h
df['h'] = df['h']/img_h
df.head()
```

	image_id	xmin	ymin	xmax	ymax	x_center	y_center	w	
0	vid_4_1000	281.259045	187.035071	327.727931	223.225547	0.450434	0.539817	0.068741	0.

```
index = list(set(df.image_id))
image = random.choice(index)
print("Image ID: %s"%(image))
img = cv2.imread(f'/content/drive/MyDrive/cvd_practical/CVD/data/training_images/{image}.jpg')
img.shape
```

```
Image ID: vid_4_880
(380, 676, 3)
```

```
image = random.choice(index)
Image(filename=f'/content/drive/MyDrive/cvd_practical/CVD/data/training_images/{image}.jpg',width=600)
```

```
# Training
source = 'training_images'
if True:
    for fold in [0]:
        val_index = index[len(index)*fold//5:len(index)*(fold+1)//5]
        for name,mini in tqdm(df.groupby('image_id')):
            if name in val_index:
                path2save = 'val2017/'
            else:
                path2save = 'train2017/'
            if not os.path.exists('/tmp/convertor/fold{}/labels/'.format(fold)+path2save):
                os.makedirs('/tmp/convertor/fold{}/labels/'.format(fold)+path2save)
            with open('/tmp/convertor/fold{}/labels/'.format(fold)+path2save+name+".txt", 'w+') as f:
                row = mini[['classes','x_center','y_center','w','h']].astype(float).values
                row = row.astype(str)
                for j in range(len(row)):
                    text = ' '.join(row[j])
                    f.write(text)
                    f.write("\n")
            if not os.path.exists('/tmp/convertor/fold{}/images/{}'.format(fold,path2save)):
                os.makedirs('/tmp/convertor/fold{}/images/{}'.format(fold,path2save))
            sh.copy("/content/drive/MyDrive/cvd_practical/CVD/data/{}/{}.jpg".format(source,name),'/tmp/convertor/fold{}/images/{}/{}.jpg
```

```
100%                                                  355/355 [01:42<00:00, 3.57it/s]
```

```
!python yolov5/detect.py --weights yolov5/yolov5s.pt --img 676 --conf 0.4 --source /content/drive/MyDrive/cvd_practical/CVD/data/testing_
```

```
image 141/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_30840.jpg: 416x704 2 cars, 5 traffic lights, 8.
image 142/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_30860.jpg: 416x704 3 cars, 3 traffic lights, 8.
image 143/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_30920.jpg: 416x704 2 cars, 2 traffic lights, 8.
image 144/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_30940.jpg: 416x704 2 cars, 2 traffic lights, 8.
image 145/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31020.jpg: 416x704 2 cars, 8.8ms
image 146/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31040.jpg: 416x704 3 cars, 8.7ms
image 147/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31060.jpg: 416x704 1 car, 8.7ms
image 148/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31080.jpg: 416x704 2 cars, 8.8ms
image 149/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31100.jpg: 416x704 1 car, 8.7ms
image 150/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31120.jpg: 416x704 2 cars, 8.7ms
image 151/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31140.jpg: 416x704 1 car, 8.8ms
image 152/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31160.jpg: 416x704 1 car, 8.7ms
image 153/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31180.jpg: 416x704 (no detections), 8.8ms
image 154/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31200.jpg: 416x704 (no detections), 8.7ms
image 155/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31260.jpg: 416x704 (no detections), 8.8ms
image 156/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31280.jpg: 416x704 (no detections), 8.7ms
image 157/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31300.jpg: 416x704 (no detections), 8.8ms
image 158/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31360.jpg: 416x704 (no detections), 8.7ms
image 159/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31380.jpg: 416x704 (no detections), 8.7ms
image 160/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31400.jpg: 416x704 (no detections), 8.7ms
image 161/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31420.jpg: 416x704 (no detections), 8.7ms
image 162/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31480.jpg: 416x704 (no detections), 8.8ms
image 163/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31500.jpg: 416x704 (no detections), 8.8ms
image 164/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31520.jpg: 416x704 (no detections), 8.7ms
image 165/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31560.jpg: 416x704 1 car, 8.8ms
image 166/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31600.jpg: 416x704 2 cars, 8.7ms
image 167/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31620.jpg: 416x704 1 car, 8.8ms
image 168/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31640.jpg: 416x704 (no detections), 8.7ms
image 169/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31660.jpg: 416x704 (no detections), 8.8ms
image 170/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31680.jpg: 416x704 (no detections), 8.8ms
image 171/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31700.jpg: 416x704 1 car, 8.7ms
image 172/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_31720.jpg: 416x704 1 car, 8.8ms
image 173/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_400.jpg: 416x704 1 car, 8.8ms
image 174/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_420.jpg: 416x704 1 car, 9.6ms
image 175/175 /content/drive/MyDrive/cvd_practical/CVD/data/testing_images/vid_5_440.jpg: 416x704 1 car, 8.8ms
Speed: 0.4ms pre-process, 9.3ms inference, 0.9ms NMS per image at shape (1, 3, 704, 704)
```

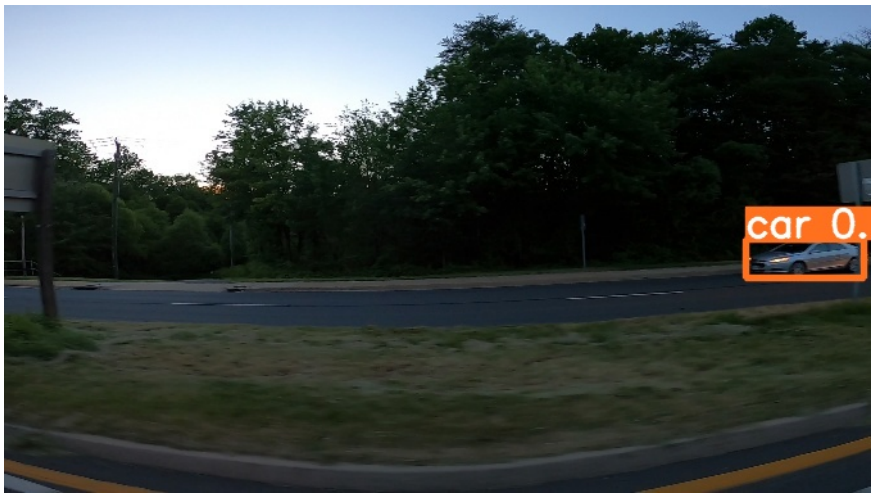
```
# Prediction
predicted_files = []
for (dirpath, dirnames, filenames) in os.walk("yolov5/runs/detect/exp"):
    predicted_files.extend(filenames)
```

```
len(predicted_files)
```

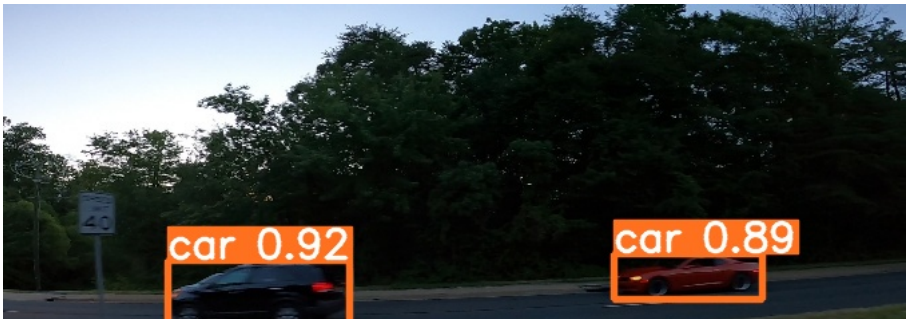
```
175
```

ChatGPT

```
Image(filename=f'yolov5/runs/detect/exp/{random.choice(predicted_files)}')
```




```
Image(filename=f'yolov5/runs/detect/exp/{random.choice(predicted_files)}')
```




```
Image(filename=f'yolov5/runs/detect/exp/{random.choice(predicted_files)}')
```

```
Image(filename=f'yolov5/runs/detect/exp/{random.choice(predicted_files)}')
```


```
# Image(filename=f'yolov5/runs/detect/exp/{random.choice(predicted_files)}')
```