

```
In [1]: import os
import glob as glob
import matplotlib.pyplot as plt
import cv2
import requests
import random
import numpy as np

np.random.seed(42)
```

```
In [2]: TRAIN = True
EPOCHS = 25
```

```
In [5]: class_names = ['Ambulance', 'Bus', 'Car', 'Motorcycle', 'Truck']
colors = np.random.uniform(0, 255, size=(len(class_names), 3))
```

```
In [6]: def yolo2bbox(bboxes):
    xmin, ymin = bboxes[0]-bboxes[2]/2, bboxes[1]-bboxes[3]/2
    xmax, ymax = bboxes[0]+bboxes[2]/2, bboxes[1]+bboxes[3]/2
    return xmin, ymin, xmax, ymax
```

```
In [7]: def plot_box(image, bboxes, labels):
    h, w, _ = image.shape
    for box_num, box in enumerate(bboxes):
        x1, y1, x2, y2 = yolo2bbox(box)
        xmin = int(x1*w)
        ymin = int(y1*h)
        xmax = int(x2*w)
        ymax = int(y2*h)
        width = xmax - xmin
        height = ymax - ymin

        class_name = class_names[int(labels[box_num])]

        cv2.rectangle(
            image,
            (xmin, ymin), (xmax, ymax),
            color=colors[class_names.index(class_name)],
            thickness=2
        )

        font_scale = min(1,max(3,int(w/500)))
        font_thickness = min(2, max(10,int(w/50)))

        p1, p2 = (int(xmin), int(ymin)), (int(xmax), int(ymax))
        tw, th = cv2.getTextSize(
            class_name,
            0, fontScale=font_scale, thickness=font_thickness
        )[0]
        p2 = p1[0] + tw, p1[1] + -th - 10
        cv2.rectangle(
            image,
            p1, p2,
            color=colors[class_names.index(class_name)],
            thickness=-1,
        )
        cv2.putText(
            image,
            class_name,
            (xmin+1, ymin-10),
            cv2.FONT_HERSHEY_SIMPLEX,
            font_scale,
```

```
        (255, 255, 255),
        font_thickness
    )
return image
```

```
In [8]: def plot(image_paths, label_paths, num_samples):
    all_training_images = glob.glob(image_paths)
    all_training_labels = glob.glob(label_paths)
    all_training_images.sort()
    all_training_labels.sort()

    num_images = len(all_training_images)

    plt.figure(figsize=(15, 12))
    for i in range(num_samples):
        j = random.randint(0,num_images-1)
        image = cv2.imread(all_training_images[j])
        with open(all_training_labels[j], 'r') as f:
            bboxes = []
            labels = []
            label_lines = f.readlines()
            for label_line in label_lines:
                label = label_line[0]
                bbox_string = label_line[2:]
                x_c, y_c, w, h = bbox_string.split(' ')
                x_c = float(x_c)
                y_c = float(y_c)
                w = float(w)
                h = float(h)
                bboxes.append([x_c, y_c, w, h])
                labels.append(label)
            result_image = plot_box(image, bboxes, labels)
            plt.subplot(2, 2, i+1)
            plt.imshow(result_image[:, :, ::-1])
            plt.axis('off')
    plt.subplots_adjust(wspace=0)
    plt.tight_layout()
    plt.show()
```

```
In [9]: plot(  
    image_paths='train/images/*',  
    label_paths='train/labels/*',  
    num_samples=4,  
)
```





Here, we write the helper functions that we need for logging of the results in the notebook while training the models.

Let's create our custom result directories so that we can easily keep track of them and carry out inference using the proper model.

```
In [10]: def set_res_dir():
    res_dir_count = len(glob.glob('runs/train/*'))
    print(f"Current number of result directories: {res_dir_count}")
    if TRAIN:
        RES_DIR = f"results_{res_dir_count+1}"
        print(RES_DIR)
    else:
        RES_DIR = f"results_{res_dir_count}"
    return RES_DIR
```

Function to Monitor TensorBoard logs.

```
In [11]: def monitor_tensorboard():
    %load_ext tensorboard
    %tensorboard --logdir runs/train
```

Clone YOLOV5 Repository

```
In [12]: if not os.path.exists('yolov5'):
    !git clone https://github.com/ultralytics/yolov5.git
```

```
In [13]: %cd yolov5/
!pwd

/home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5/UPDATED_code/yolov5
/home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5/UPDATED_code/yolov5
```

In [14]: !pip install -r requirements.txt

```
Requirement already satisfied: matplotlib>=3.2.2 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 4)) (3.5.1)
Requirement already satisfied: numpy>=1.18.5 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 5)) (1.22.3)
Requirement already satisfied: opencv-python>=4.1.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 6)) (4.5.5.64)
Requirement already satisfied: Pillow>=7.1.2 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 7)) (9.1.0)
Requirement already satisfied: PyYAML>=5.3.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 8)) (6.0)
Requirement already satisfied: requests>=2.23.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 9)) (2.27.1)
Requirement already satisfied: scipy>=1.4.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 10)) (1.8.0)
Requirement already satisfied: torch>=1.7.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 11)) (1.11.0+cu113)
Requirement already satisfied: torchvision>=0.8.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 12)) (0.12.0+cu113)
Requirement already satisfied: tqdm>=4.41.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 13)) (4.64.0)
Requirement already satisfied: tensorboard>=2.4.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 16)) (2.8.0)
Requirement already satisfied: pandas>=1.1.4 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 20)) (1.4.2)
Requirement already satisfied: seaborn>=0.11.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 21)) (0.11.2)
Requirement already satisfied: thop in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from -r requirements.txt (line 37)) (0.0.31.post2005241907)
Requirement already satisfied: cycler>=0.10 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from matplotlib>=3.2.2->-r requirements.txt (line 4)) (0.11.0)
Requirement already satisfied: packaging>=20.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from matplotlib>=3.2.2->-r requirements.txt (line 4)) (21.3)
Requirement already satisfied: python-dateutil>=2.7 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from matplotlib>=3.2.2->-r requirements.txt (line 4)) (2.8.2)
Requirement already satisfied: pyparsing>=2.2.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from matplotlib>=3.2.2->-r requirements.txt (line 4)) (3.0.8)
Requirement already satisfied: kiwisolver>=1.0.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from matplotlib>=3.2.2->-r requirements.txt (line 4)) (1.4.2)
Requirement already satisfied: fonttools>=4.22.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-
```

```
e-packages (from matplotlib>=3.2.2->-r requirements.txt (line 4)) (4.32.0)
Requirement already satisfied: charset-normalizer~=2.0.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from requests>=2.23.0->-r requirements.txt (line 9)) (2.0.12)
Requirement already satisfied: certifi>=2017.4.17 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from requests>=2.23.0->-r requirements.txt (line 9)) (2020.6.20)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from requests>=2.23.0->-r requirements.txt (line 9)) (1.26.9)
Requirement already satisfied: idna<4,>=2.5 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from requests>=2.23.0->-r requirements.txt (line 9)) (3.3)
Requirement already satisfied: typing-extensions in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from torch>=1.7.0->-r requirements.txt (line 11)) (4.1.1)
Requirement already satisfied: wheel>=0.26 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (0.37.1)
Requirement already satisfied: setuptools>=41.0.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (61.2.0)
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (0.6.1)
Requirement already satisfied: markdown>=2.6.8 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (3.3.6)
Requirement already satisfied: werkzeug>=0.11.15 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (2.1.1)
Requirement already satisfied: protobuf>=3.6.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (3.20.0)
Requirement already satisfied: absl-py>=0.4 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (1.0.0)
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (0.4.6)
Requirement already satisfied: grpcio>=1.24.3 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (1.44.0)
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (1.8.1)
Requirement already satisfied: google-auth<3,>=1.6.3 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from tensorboard>=2.4.1->-r requirements.txt (line 16)) (2.6.5)
Requirement already satisfied: pytz>=2020.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from pandas>=1.1.4->-r requirements.txt (line 20)) (2022.1)
Requirement already satisfied: six in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from absl-py>=0.4->tensorboard>=2.4.1->-r requirements.txt (line 16)) (1.16.0)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from google-auth<3,>=1.6.3->tensorboard>=2.4.1->-r requirements.txt (line 16)) (0.2.8)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-packages (from google-auth<3,>=1.6.3->tensorboard>=2.4.1->-r requirements.txt (line 16)) (5.0.0)
Requirement already satisfied: rsa<5,>=3.1.4 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-pa
```

```
ckages (from google-auth<3,>=1.6.3->tensorboard>=2.4.1->-r requirements.txt (line 16)) (4.8)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python
3.10/site-packages (from google-auth-oauthlib<0.5,>=0.4.1->tensorboard>=2.4.1->-r requirements.txt (line 1
6)) (1.3.1)
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/
site-packages (from pyasn1-modules>=0.2.1->google-auth<3,>=1.6.3->tensorboard>=2.4.1->-r requirements.txt (l
ine 16)) (0.4.8)
Requirement already satisfied: oauthlib>=3.0.0 in /home/sovitz/miniconda3/envs/torch1110/lib/python3.10/site-
packages (from requests-oauthlib>=0.7.0->google-auth-oauthlib<0.5,>=0.4.1->tensorboard>=2.4.1->-r requiremen
```

In [15]: monitor_tensorboard()

In [16]:
RES_DIR = set_res_dir()
if TRAIN:
 !python train.py --data ./data.yaml --weights yolov5s.pt \
 --img 640 --epochs {EPOCHS} --batch-size 16 --name {RES_DIR}

```
Current number of result directories: 0
results_1
train: weights=yolov5s.pt, cfg=, data=../data.yaml, hyp=data/hyps/hyp.scratch-low.yaml, epochs=25, batch_size=16, imgsz=640, rect=False, resume=False, nosave=False, noval=False, noautoanchor=False, noplots=False, evolive=None, bucket=, cache=None, image_weights=False, device=, multi_scale=False, single_cls=False, optimizer=SGD, sync_bn=False, workers=8, project=runs/train, name=results_1, exist_ok=False, quad=False, cos_lr=False, label_smoothing=0.0, patience=100, freeze=[0], save_period=-1, local_rank=-1, entity=None, upload_dataset=False, bbox_interval=-1, artifact_alias=latest
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 19 (delta 6), reused 14 (delta 4), pack-reused 0
Unpacking objects: 100% (19/19), 35.26 KiB | 120.00 KiB/s, done.
From https://github.com/ultralytics/yolov5 (https://github.com/ultralytics/yolov5)
  7d46c69..f43cd53  master           -> origin/master
  07ece63..f6585dd  apple/mps        -> origin/apple/mps
  34b2c0c..ba41a06  update/ConvTranspose2d -> origin/update/ConvTranspose2d
github: ⚠️ YOLOv5 is out of date by 27 commits. Use `git pull` or `git clone https://github.com/ultralytics/yolov5` (https://github.com/ultralytics/yolov5`) to update.
```

```
In [17]: def show_valid_results(RES_DIR):
    !ls runs/train/{RES_DIR}
    EXP_PATH = f"runs/train/{RES_DIR}"
    validation_pred_images = glob.glob(f"{EXP_PATH}/*_pred.jpg")
    print(validation_pred_images)
    for pred_image in validation_pred_images:
        image = cv2.imread(pred_image)
        plt.figure(figsize=(19, 16))
        plt.imshow(image[:, :, ::-1])
        plt.axis('off')
        plt.show()
```

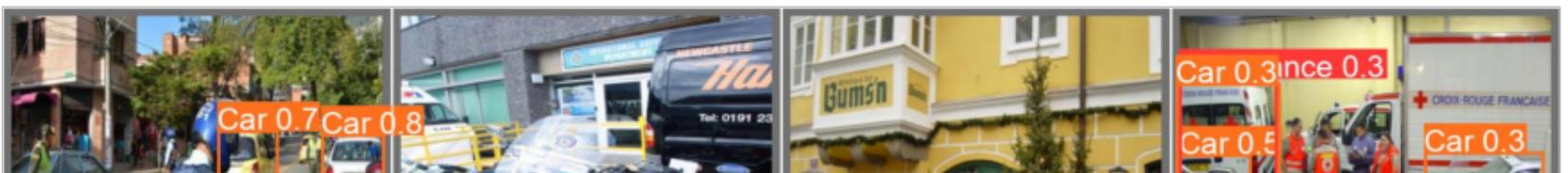
The following functions are for carrying out inference on images and videos.

```
In [18]: def inference(RES_DIR, data_path):
    infer_dir_count = len(glob.glob('runs/detect/*'))
    print(f"Current number of inference detection directories: {infer_dir_count}")
    INFER_DIR = f"inference_{infer_dir_count+1}"
    print(INFER_DIR)
    !python detect.py --weights runs/train/{RES_DIR}/weights/best.pt \
    --source {data_path} --name {INFER_DIR}
    return INFER_DIR
```

```
In [19]: def visualize(INFER_DIR):
    INFER_PATH = f"runs/detect/{INFER_DIR}"
    infer_images = glob.glob(f"{INFER_PATH}/*.jpg")
    print(infer_images)
    for pred_image in infer_images:
        image = cv2.imread(pred_image)
        plt.figure(figsize=(19, 16))
        plt.imshow(image[:, :, ::-1])
        plt.axis('off')
        plt.show()
```

```
In [20]: show_valid_results(RES_DIR)
```

```
confusion_matrix.png          results.png
events.out.tfevents.1653007447.sovit-GV62-8RE.13809.0 train_batch0.jpg
F1_curve.png           train_batch1.jpg
hyp.yaml             train_batch2.jpg
labels_correlogram.jpg val_batch0_labels.jpg
labels.jpg            val_batch0_pred.jpg
opt.yaml             val_batch1_labels.jpg
P_curve.png          val_batch1_pred.jpg
PR_curve.png         val_batch2_labels.jpg
R_curve.png          val_batch2_pred.jpg
results.csv          weights
['runs/train/results_1/val_batch0_pred.jpg', 'runs/train/results_1/val_batch1_pred.jpg', 'runs/train/results_1/val_batch2_pred.jpg']
```



Inference on Images

To carry out inference on images, we just need to provide the directory path where all the images are stored, and inference will happen on all images automatically.

In [22]: # Inference on images.

```
IMAGE_INFER_DIR = inference(RES_DIR, 'inference_images')
```

Current number of inference detection directories: 0

inference_1

```
detect: weights=['runs/train/results_1/weights/best.pt'], source=inference_images, data=data/coco128.yaml, imgsz=[640, 640], conf_thres=0.25, iou_thres=0.45, max_det=1000, device=, view_img=False, save_txt=False, save_conf=False, save_crop=False, nosave=False, classes=None, agnostic_nms=False, augment=False, visualize=False, update=False, project=runs/detect, name=inference_1, exist_ok=False, line_thickness=3, hide_labels=False, hide_conf=False, half=False, dnn=False
```

YOLOv5 🚀 v6.1-177-gd059d1d torch 1.11.0+cu113 CUDA:0 (NVIDIA GeForce GTX 1060, 6078MiB)

Fusing layers...

Model summary: 213 layers, 7023610 parameters, 0 gradients

image 1/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_1.jpg: 448x640 1 Car, Done. (0.009s)

image 2/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_2.jpg: 416x640 2 Cars, Done. (0.009s)

image 3/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_3.jpg: 448x640 3 Ambulances, 1 Bus, Done. (0.009s)

image 4/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_4.jpg: 480x640 4 Cars, Done. (0.010s)

Speed: 0.3ms pre-process, 9.4ms inference, 0.8ms NMS per image at shape (1, 3, 640, 640)

Results saved to runs/detect/inference_1

In [23]: `visualize(IMAGE_INFERENCE_DIR)`

```
['runs/detect/inference_1/image_1.jpg', 'runs/detect/inference_1/image_4.jpg', 'runs/detect/inference_1/image_2.jpg', 'runs/detect/inference_1/image_3.jpg']
```



Inference on Videos

```
In [24]: inference(RES_DIR, 'inference_videos')
```

```
Current number of inference detection directories: 1
inference_2
detect: weights=['runs/train/results_1/weights/best.pt'], source=inference_videos, data=data/coco128.yaml, i
mgsz=[640, 640], conf_thres=0.25, iou_thres=0.45, max_det=1000, device=, view_img=False, save_txt=False, sav
e_conf=False, save_crop=False, nosave=False, classes=None, agnostic_nms=False, augment=False, visualize=False,
update=False, project=runs/detect, name=inference_2, exist_ok=False, line_thickness=3, hide_labels=False,
hide_conf=False, half=False, dnn=False
YOLOv5 🚀 v6.1-177-gd059d1d torch 1.11.0+cu113 CUDA:0 (NVIDIA GeForce GTX 1060, 6078MiB)

Fusing layers...
Model summary: 213 layers, 7023610 parameters, 0 gradients
video 1/4 (1/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 Done. (0.015s)
video 1/4 (2/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 Done. (0.010s)
video 1/4 (3/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 2 Cars, Done. (0.007s)
video 1/4 (4/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 3 Cars, Done. (0.007s)
```

Training and Inference using Medium Model

```
In [25]: monitor_tensorboard()
```

```
The tensorboard extension is already loaded. To reload it, use:
%reload_ext tensorboard
```

```
Reusing TensorBoard on port 6006 (pid 13752), started 0:14:09 ago. (Use '!kill 13752' to kill it.)
```

```
In [26]: RES_DIR = set_res_dir()
if TRAIN:
    !python train.py --data ./data.yaml --weights yolov5m.pt \
    --img 640 --epochs {EPOCHS} --batch-size 16 --name {RES_DIR}

Current number of result directories: 1
results_2
train: weights=yolov5m.pt, cfg=, data=../data.yaml, hyp=data/hyps/hyp.scratch-low.yaml, epochs=25, batch_size=16, imgsz=640, rect=False, resume=False, nosave=False, noval=False, noautoanchor=False, noplots=False, evo_lve=None, bucket=, cache=None, image_weights=False, device=, multi_scale=False, single_cls=False, optimizer=SGD, sync_bn=False, workers=8, project=runs/train, name=results_2, exist_ok=False, quad=False, cos_lr=False, label_smoothing=0.0, patience=100, freeze=[0], save_period=-1, local_rank=-1, entity=None, upload_dataset=False, bbox_interval=-1, artifact_alias=latest
github: ⚠️ YOLOv5 is out of date by 27 commits. Use `git pull` or `git clone https://github.com/ultralytics/yolov5` (https://github.com/ultralytics/yolov5) to update.
YOLOv5 🚀 v6.1-177-gd059d1d torch 1.11.0+cu113 CUDA:0 (NVIDIA GeForce GTX 1060, 6078MiB)

hyperparameters: lr0=0.01, lrf=0.01, momentum=0.937, weight_decay=0.0005, warmup_epochs=3.0, warmup_momentum=0.8, warmup_bias_lr=0.1, box=0.05, cls=0.5, cls_pw=1.0, obj=1.0, obj_pw=1.0, iou_t=0.2, anchor_t=4.0, fl_gamma=0.0, hsv_h=0.015, hsv_s=0.7, hsv_v=0.4, degrees=0.0, translate=0.1, scale=0.5, shear=0.0, perspective=0.0, flipud=0.0, fliplr=0.5, mosaic=1.0, mixup=0.0, copy_paste=0.0
Weights & Biases: run 'pip install wandb' to automatically track and visualize YOLOv5 🚀 runs (RECOMMENDED)
TensorBoard: Start with 'tensorboard --logdir runs/train', view at http://localhost:6006/ (http://localhost:6006/)
```

In [27]: # Inference on images.

```
IMAGE_INFER_DIR = inference(RES_DIR, 'inference_images')
```

Current number of inference detection directories: 2

inference_3

```
detect: weights=['runs/train/results_2/weights/best.pt'], source=inference_images, data=data/coco128.yaml, imgsz=[640, 640], conf_thres=0.25, iou_thres=0.45, max_det=1000, device=, view_img=False, save_txt=False, save_conf=False, save_crop=False, nosave=False, classes=None, agnostic_nms=False, augment=False, visualize=False, update=False, project=runs/detect, name=inference_3, exist_ok=False, line_thickness=3, hide_labels=False, hide_conf=False, half=False, dnn=False
```

YOLOv5 🚀 v6.1-177-gd059d1d torch 1.11.0+cu113 CUDA:0 (NVIDIA GeForce GTX 1060, 6078MiB)

Fusing layers...

Model summary: 290 layers, 20869098 parameters, 0 gradients

image 1/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_1.jpg: 448x640 2 Ambulances, 1 Car, 4 Trucks, Done. (0.021s)

image 2/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_2.jpg: 416x640 3 Cars, Done. (0.021s)

image 3/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_3.jpg: 448x640 2 Ambulances, 1 Bus, Done. (0.021s)

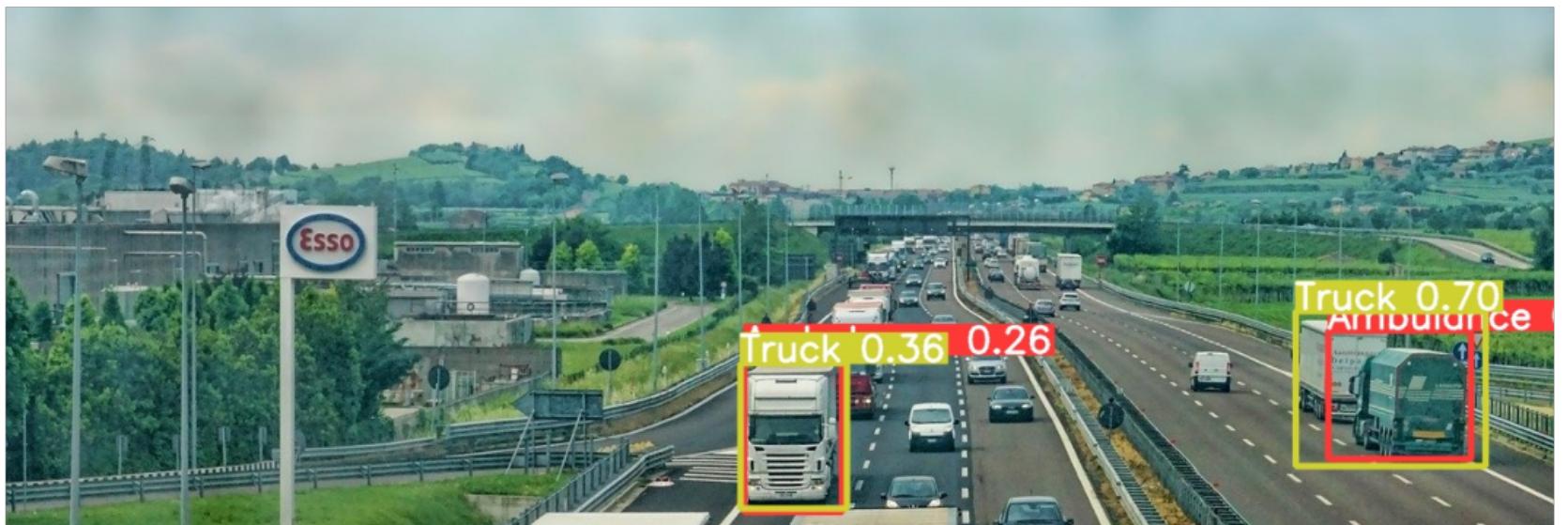
image 4/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_4.jpg: 480x640 5 Cars, Done. (0.022s)

Speed: 0.3ms pre-process, 21.4ms inference, 0.8ms NMS per image at shape (1, 3, 640, 640)

Results saved to runs/detect/inference_3

```
In [28]: visualize(IMAGE_INFERENCE_DIR)
```

```
['runs/detect/inference_3/image_1.jpg', 'runs/detect/inference_3/image_4.jpg', 'runs/detect/inference_3/image_2.jpg', 'runs/detect/inference_3/image_3.jpg']
```



```
In [29]: inference(RES_DIR, 'inference_videos')
```

```
Current number of inference detection directories: 3
inference_4
detect: weights=['runs/train/results_2/weights/best.pt'], source=inference_videos, data=data/coco128.yaml, imgsz=[640, 640], conf_thres=0.25, iou_thres=0.45, max_det=1000, device=, view_img=False, save_txt=False, save_conf=False, save_crop=False, nosave=False, classes=None, agnostic_nms=False, augment=False, visualize=False, update=False, project=runs/detect, name=inference_4, exist_ok=False, line_thickness=3, hide_labels=False, hide_conf=False, half=False, dnn=False
YOLOv5 🚀 v6.1-177-gd059d1d torch 1.11.0+cu113 CUDA:0 (NVIDIA GeForce GTX 1060, 6078MiB)

Fusing layers...
Model summary: 290 layers, 20869098 parameters, 0 gradients
video 1/4 (1/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 Done. (0.030s)
video 1/4 (2/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 5 Cars, Done. (0.019s)
video 1/4 (3/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 4 Cars, Done. (0.019s)
video 1/4 (4/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 4 Cars, Done. (0.019s)
video 1/4 (5/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 4 Cars, Done. (0.019s)
```

Freezing Layers and Training the Medium Model

The Medium model (yolov5m) contains 25 blocks layers in total more than 20 million parameters. We need not train all the layers. Let's freeze a few layers and train again. This will result in faster iteration per epoch. Here, we freeze the first 15 blocks.

```
In [30]: monitor_tensorboard()
```

```
The tensorboard extension is already loaded. To reload it, use:
%reload_ext tensorboard
```

```
Reusing TensorBoard on port 6006 (pid 13752), started 0:47:16 ago. (Use '!kill 13752' to kill it.)
```

```
In [31]: RES_DIR = set_res_dir()
if TRAIN:
    !python train.py --data ./data.yaml --weights yolov5m.pt \
    --img 640 --epochs {EPOCHS} --batch-size 16 --name {RES_DIR} \
    --freeze 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

Current number of result directories: 2

results_3

train: weights=yolov5m.pt, cfg=, data=../data.yaml, hyp=data/hyps/hyp.scratch-low.yaml, epochs=25, batch_size=16, imgsz=640, rect=False, resume=False, nosave=False, noval=False, noautoanchor=False, noplots=False, evo_lve=None, bucket=, cache=None, image_weights=False, device=, multi_scale=False, single_cls=False, optimizer=SGD, sync_bn=False, workers=8, project=runs/train, name=results_3, exist_ok=False, quad=False, cos_lr=False, label_smoothing=0.0, patience=100, freeze=[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14], save_period=-1, local_rank=-1, entity=None, upload_dataset=False, bbox_interval=-1, artifact_alias=latest

github: ⚠️ YOLOv5 is out of date by 27 commits. Use `git pull` or `git clone <https://github.com/ultralytics/yolov5>` to update.

YOLOv5 🚀 v6.1-177-gd059d1d torch 1.11.0+cu113 CUDA:0 (NVIDIA GeForce GTX 1060, 6078MiB)

hyperparameters: lr0=0.01, lrf=0.01, momentum=0.937, weight_decay=0.0005, warmup_epochs=3.0, warmup_momentum=0.8, warmup_bias_lr=0.1, box=0.05, cls=0.5, cls_pw=1.0, obj=1.0, obj_pw=1.0, iou_t=0.2, anchor_t=4.0, fl_gamma=0.0, hsv_h=0.015, hsv_s=0.7, hsv_v=0.4, degrees=0.0, translate=0.1, scale=0.5, shear=0.0, perspective=0.0, flipud=0.0, fliplr=0.5, mosaic=1.0, mixup=0.0, copy_paste=0.0

Weights & Biases: run 'pip install wandb' to automatically track and visualize YOLOv5 🚀 runs (RECOMMENDED)

TensorBoard: Start with 'tensorboard --logdir runs/train', view at <http://localhost:6006/> (<http://localhost:6006/>)

In [32]: # Inference on images.

```
IMAGE_INFER_DIR = inference(RES_DIR, 'inference_images')
```

Current number of inference detection directories: 4

inference_5

```
detect: weights=['runs/train/results_3/weights/best.pt'], source=inference_images, data=data/coco128.yaml, imgsz=[640, 640], conf_thres=0.25, iou_thres=0.45, max_det=1000, device=, view_img=False, save_txt=False, save_conf=False, save_crop=False, nosave=False, classes=None, agnostic_nms=False, augment=False, visualize=False, update=False, project=runs/detect, name=inference_5, exist_ok=False, line_thickness=3, hide_labels=False, hide_conf=False, half=False, dnn=False
```

YOLOv5 🚀 v6.1-177-gd059d1d torch 1.11.0+cu113 CUDA:0 (NVIDIA GeForce GTX 1060, 6078MiB)

Fusing layers...

Model summary: 290 layers, 20869098 parameters, 0 gradients

image 1/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_1.jpg: 448x640 3 Buss, 5 Cars, Done. (0.021s)

image 2/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_2.jpg: 416x640 1 Bus, 6 Cars, Done. (0.020s)

image 3/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_3.jpg: 448x640 2 Buss, 1 Car, Done. (0.021s)

image 4/4 /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_using_yolov5 /UPDATED_code/yolov5/inference_images/image_4.jpg: 480x640 3 Cars, Done. (0.022s)

Speed: 0.3ms pre-process, 21.3ms inference, 0.9ms NMS per image at shape (1, 3, 640, 640)

Results saved to runs/detect/inference_5

```
In [33]: visualize(IMAGE_INFERENCE_DIR)
```

```
['runs/detect/inference_5/image_1.jpg', 'runs/detect/inference_5/image_4.jpg', 'runs/detect/inference_5/image_2.jpg', 'runs/detect/inference_5/image_3.jpg']
```



```
In [34]: inference(RES_DIR, 'inference_videos')
```

```
Current number of inference detection directories: 5
inference_6
detect: weights=['runs/train/results_3/weights/best.pt'], source=inference_videos, data=data/coco128.yaml, i
mgsz=[640, 640], conf_thres=0.25, iou_thres=0.45, max_det=1000, device=, view_img=False, save_txt=False, sav
e_conf=False, save_crop=False, nosave=False, classes=None, agnostic_nms=False, augment=False, visualize=False,
update=False, project=runs/detect, name=inference_6, exist_ok=False, line_thickness=3, hide_labels=False,
hide_conf=False, half=False, dnn=False
YOLOv5 🚀 v6.1-177-gd059d1d torch 1.11.0+cu113 CUDA:0 (NVIDIA GeForce GTX 1060, 6078MiB)

Fusing layers...
Model summary: 290 layers, 20869098 parameters, 0 gradients
video 1/4 (1/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 Done. (0.019s)
video 1/4 (2/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 4 Cars, Done. (0.019s)
video 1/4 (3/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 1 Bus, 4 Cars, Done. (0.019s)
video 1/4 (4/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 2 Buss, 4 Cars, Done. (0.019s)
video 1/4 (5/513) /home/sovitz/my_data/Data_Science/Big_Vision/my_blogs/custom_object_detection_training_usin
g_yolov5/UPDATED_code/yolov5/inference_videos/video_1.mp4: 384x640 1 Bus, 4 Cars, Done. (0.019s)
```