

گزارش پروژه پردازش تصویر

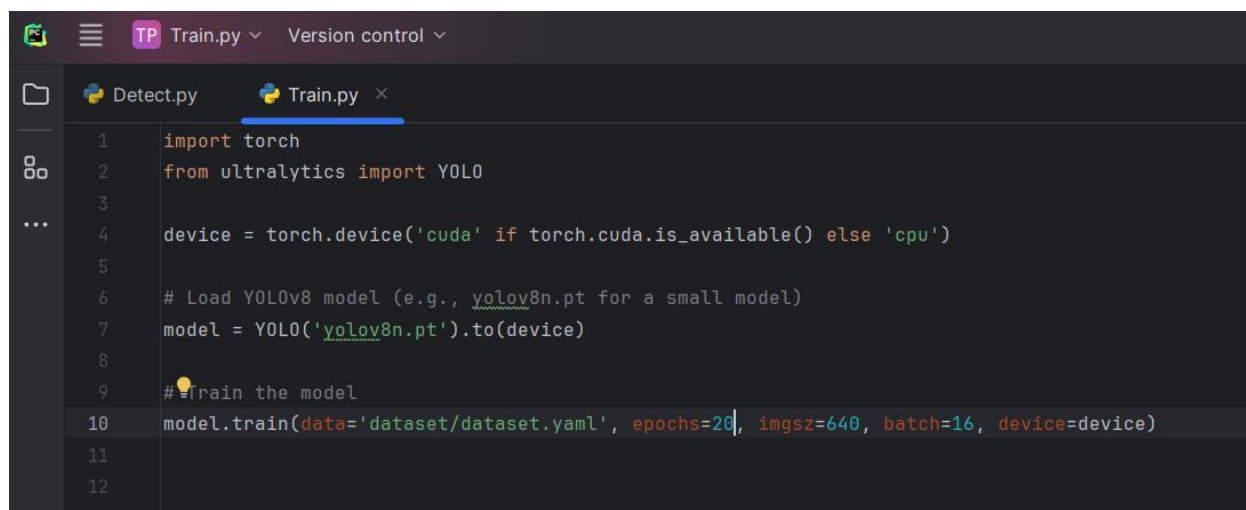
دکتر آکوشیده

فرید افراخته 990122680092

پیاده سازی با زبان پایتون در نرم افزار Pycharm انجام شده.

در مرحله اول با توجه به dataset آماده شده در Roboflow خروجی فایل yml را نیاز داریم. دیتاست طبق مدل yolov8 آماده شده، همچنین فایل آن الصاق شده.

در فایل train اقدام به آموزش مدل آماده خود کرده:



```
1 import torch
2 from ultralytics import YOLO
3
4 device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
5
6 # Load YOLOv8 model (e.g., yolov8n.pt for a small model)
7 model = YOLO('yolov8n.pt').to(device)
8
9 # Train the model
10 model.train(data='dataset/dataset.yml', epochs=20, imgsz=640, batch=16, device=device)
11
12
```

مدل آماده yolov8 دانلود شده را در فایل پروژه قرار داده و فراخوانی میکنیم.

و در خط 10 با تعیین تعداد 20 epochs به دلیل محدودیت cpu شروع به آموزش میکنیم:  
خروجی در ترمینال:

```
Transferred 319/355 items from pretrained weights
Freezing layer 'model.22.dfl.conv.weight'
train: Scanning C:\Users\Farid\datasets\dataset\train\labels.cache... 384 images, 0 backgrounds, 0 corrupt: 100%|██████████| 384/384 [00:00<?, ?it/s]
val: Scanning C:\Users\Farid\datasets\dataset\valid\labels.cache... 24 images, 0 backgrounds, 0 corrupt: 100%|██████████| 24/24 [00:00<?, ?it/s]
Plotting labels to runs\detect\train3\labels.jpg...
optimizer: 'optimizer=auto' found, ignoring 'lr0=0.01' and 'momentum=0.937' and determining best 'optimizer', 'lr0' and 'momentum' automatically...
optimizer: AdamW(lr=0.001429, momentum=0.9) with parameter groups 57 weight(decay=0.0), 64 weight(decay=0.0005), 63 bias(decay=0.0)
Image sizes 640 train, 640 val
Using 0 dataloader workers
Logging results to runs\detect\train3
Starting training for 20 epochs...

Epoch   GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
1/20      0G      1.389     4.092     1.054      245      640: 25%|██████| 6/24 [00:32<01:40, 5.60s/it]
```

بعد از پایان در آدرس runs\detect\train\weights در پوشه شبکه یکی از مدل های تولید شده را انتخاب کرده.

در فایل Detect.py نرم افزار فایل best.pt را به عنوان ورودی دریافت کرده و شروع به لیبل گذاری بر روی فایل CarVideo.mp4 که به آن می‌دهیم میکند.

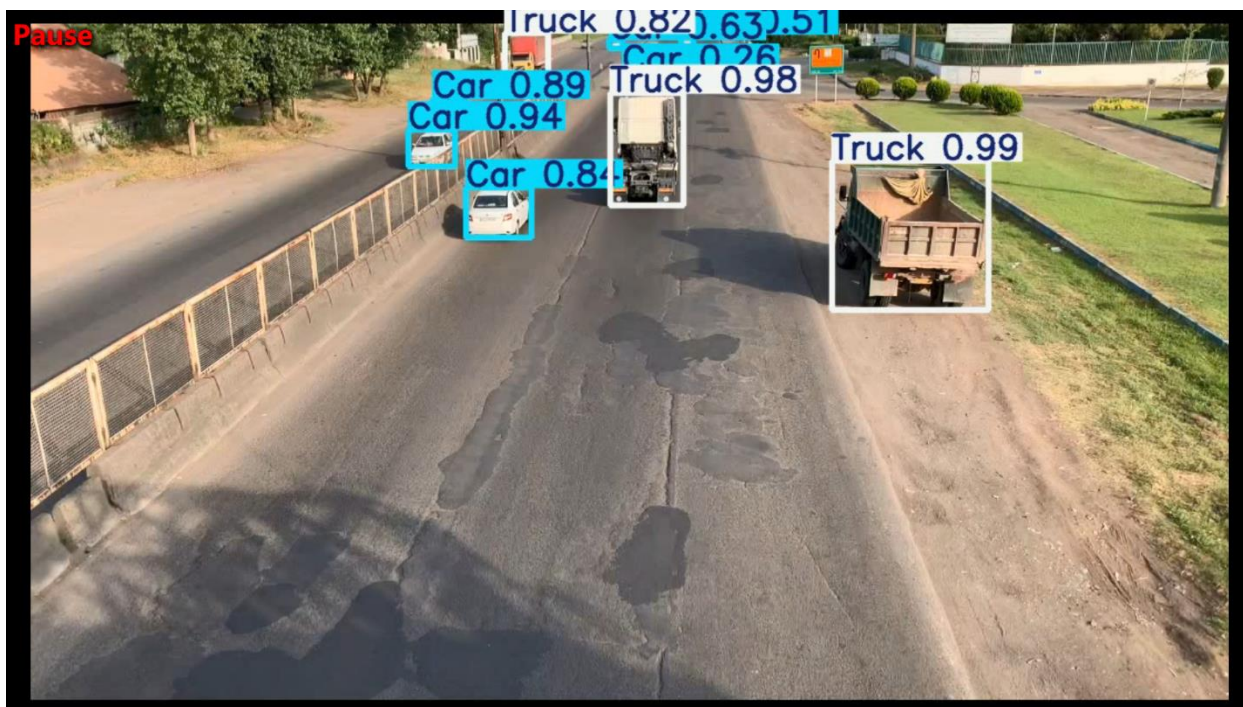
بس از پایان پردازش خروجی را تحت عنوان output.mp4 ذخیره میکند.

پردازش به این صورت است که نرم افزار فریم به فریم ویدیو ورودی را پردازش کرده و در پروسه پردازش بر روی ماشین ها طبق dataset که با استفاده از آن آموزش دیده لیبل گذاری کرده شمارش میکند:

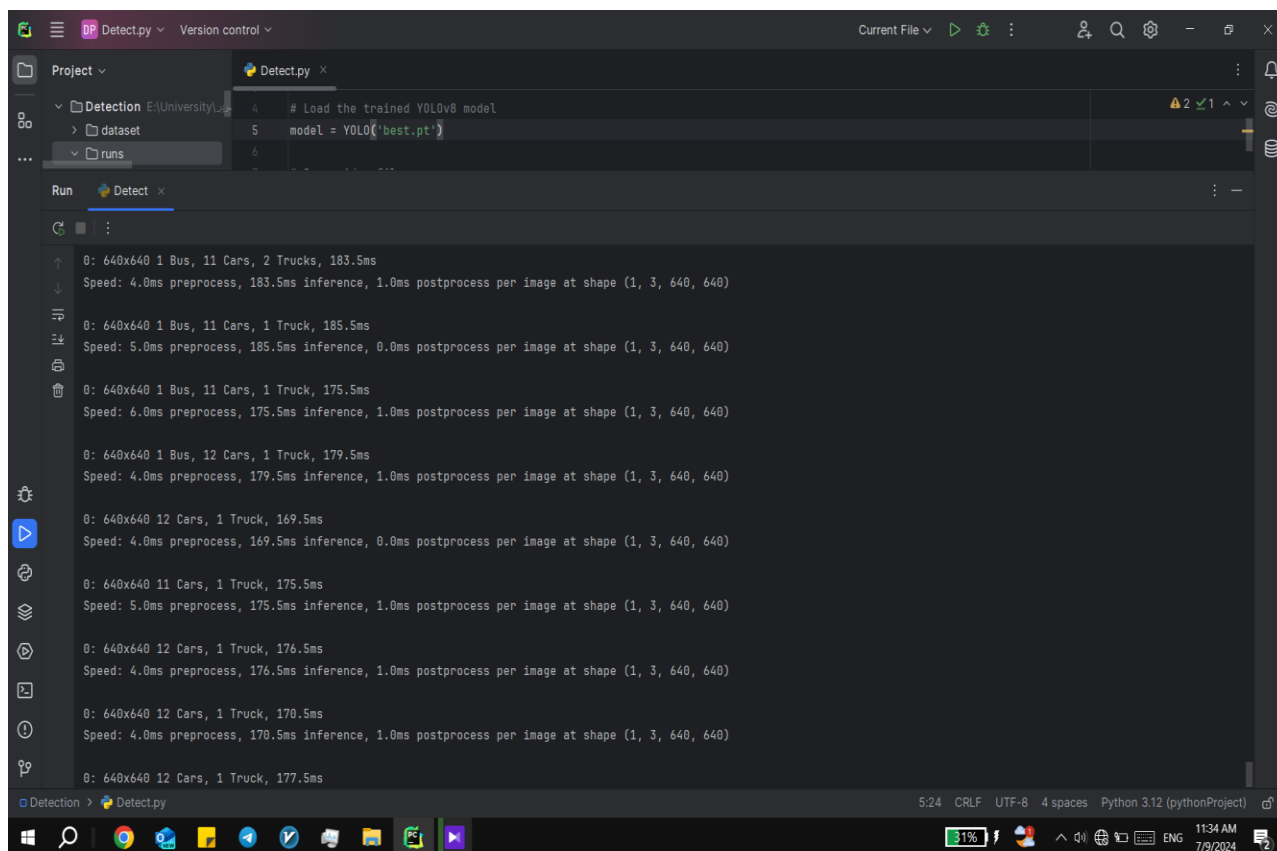
ورودی از یک فریم:



خروجی آن فریم:



همچنین در ترمینال تعداد لیبل‌های شمارش شده نمایش داده می‌شوند:



The screenshot shows a VS Code editor with a Python file named 'Detect.py' open. The file contains two lines of code: a comment '# Load the trained YOLOv8 model' and a line 'model = YOLO('best.pt')'. Below the editor, a terminal window titled 'Run Detect' displays the output of the script. The output consists of ten lines, each showing the dimensions of an image (0: 640x640), the detected objects (e.g., 1 Bus, 11 Cars, 2 Trucks), and the processing speed (e.g., 4.0ms preprocess, 183.5ms inference). The detected objects vary across the images, including combinations of Buses, Cars, and Trucks. The status bar at the bottom indicates the file is 'Detect.py' in the 'Detection' folder, using Python 3.12, with a 5:24 CRLF UTF-8 encoding and 4 spaces for indentation. The system tray at the bottom right shows the date and time as 11:34 AM on 7/9/2024.

```
4 # Load the trained YOLOv8 model
5 model = YOLO('best.pt')
```

```
0: 640x640 1 Bus, 11 Cars, 2 Trucks, 183.5ms
Speed: 4.0ms preprocess, 183.5ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)

0: 640x640 1 Bus, 11 Cars, 1 Truck, 185.5ms
Speed: 5.0ms preprocess, 185.5ms inference, 0.0ms postprocess per image at shape (1, 3, 640, 640)

0: 640x640 1 Bus, 11 Cars, 1 Truck, 175.5ms
Speed: 6.0ms preprocess, 175.5ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)

0: 640x640 1 Bus, 12 Cars, 1 Truck, 179.5ms
Speed: 4.0ms preprocess, 179.5ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)

0: 640x640 12 Cars, 1 Truck, 169.5ms
Speed: 4.0ms preprocess, 169.5ms inference, 0.0ms postprocess per image at shape (1, 3, 640, 640)

0: 640x640 11 Cars, 1 Truck, 175.5ms
Speed: 5.0ms preprocess, 175.5ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)

0: 640x640 12 Cars, 1 Truck, 176.5ms
Speed: 4.0ms preprocess, 176.5ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)

0: 640x640 12 Cars, 1 Truck, 170.5ms
Speed: 4.0ms preprocess, 170.5ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)

0: 640x640 12 Cars, 1 Truck, 177.5ms
```

Detection > Detect.py 5:24 CRLF UTF-8 4 spaces Python 3.12 (pythonProject) 11:34 AM 7/9/2024