Create Conda Environment

conda create –n yolov8 python=3.9

conda activate yolov8

pip install ultralytics

cls

yolov8>>python

>>import torch

>>torch.cuda.is_available()

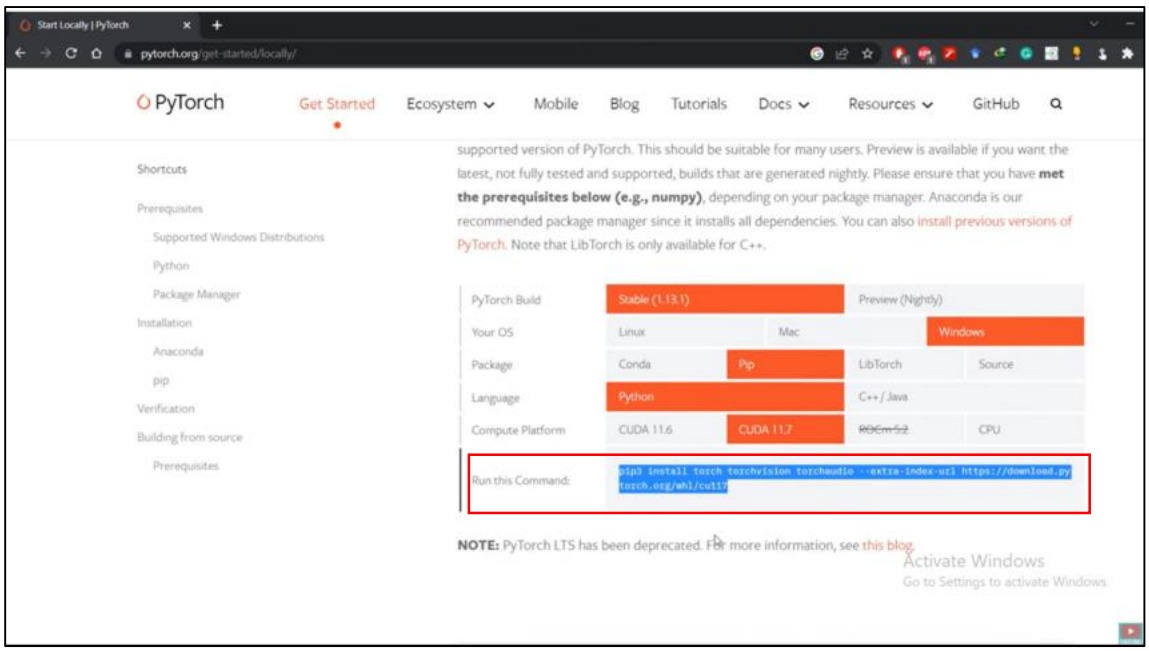
Fasle

>>torch.__version__

‘1.13.1+cpu’

```
(yolov8) C:\Users\zeeshan>python
Python 3.9.15 (main, Nov 24 2022, 14:39:17) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> torch.cuda.is_available()
False
>>> torch.__version__
'1.13.0+cpu'
>>>
```

Install CUDA to run yolov5 on GPU.



But I don't have GPU so I will use yolov8 on CPU. If you have GPU then install CUDA using the above mentioned line in the screenshot.

Object Detection:

Here I am using Nano model (YOLOv8n.pt)

Model	size (pixels)	mAP^{val} 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

```
>>yolo task=detect mode=predict model=yolov8n.pt source=img.jpg
```

Result: C:\Users\zeeshan\runs\detect\predict

Set the confidence level at 0.5:

```
>> yolo task=detect mode=predict model=yolov8n.pt source=1.jpg conf=0.5
```

Result: C:\Users\zeeshan\runs\detect\predict

Show output in real-time:

```
>> yolo task=detect mode=predict model=yolov8n.pt source=1.jpg conf=0.5 show=true
```

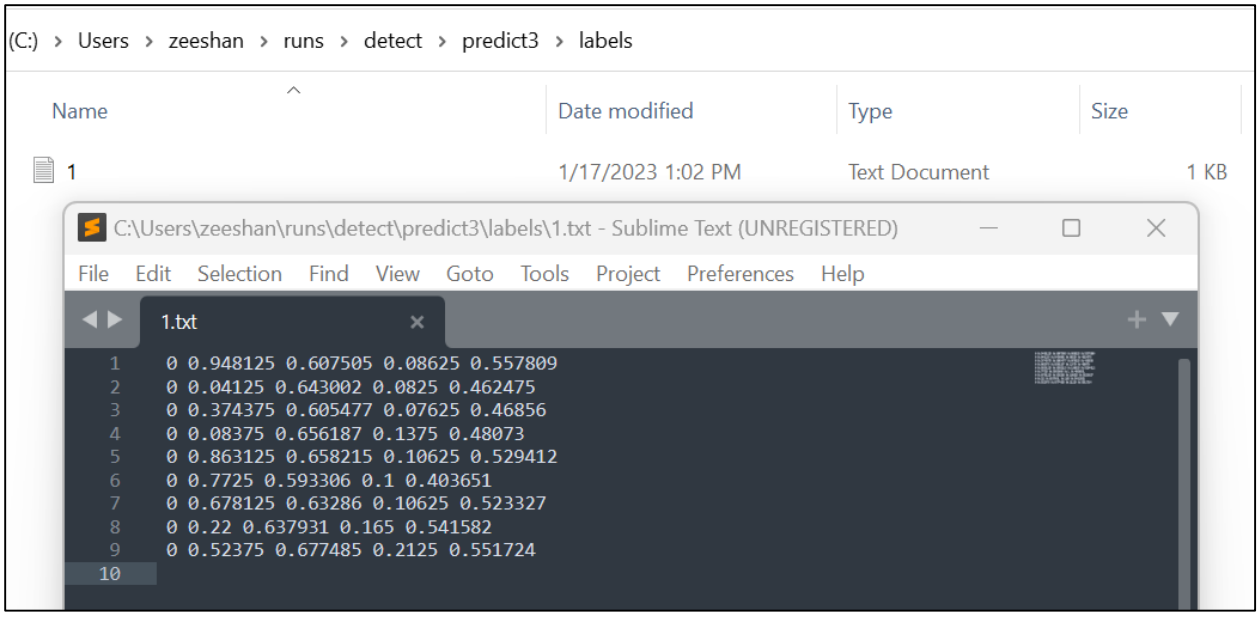
It will show output at real-time and it will close automatically in 1 millisecond.

Save the bounding boxes information:

```
>>yolo task=detect mode=predict model=yolov8n.pt source=1.jpg conf=0.5 save_txt=true
```

It will save the image and its bounding boxes information.

Result: C:\Users\zeeshan\runs\detect\predict3\labels



Save cropped objects:

```
>>yolo task=detect mode=predict model=yolov8n.pt source=1.jpg conf=0.5 save_crop=true
```

Results: C:\Users\zeeshan\runs\detect\predict4

Remove the label and confidence level:

We will use two flags. (hide_labels=true & hide_conf=true)

```
>yolo task=detect mode=predict model=yolov8n.pt source=1.jpg conf=0.5 save_crop=true
hide_labels=true hide_conf=true
```

Results:

Object Detection on webcam:

Now I am using small model (YOLOv8s.pt)

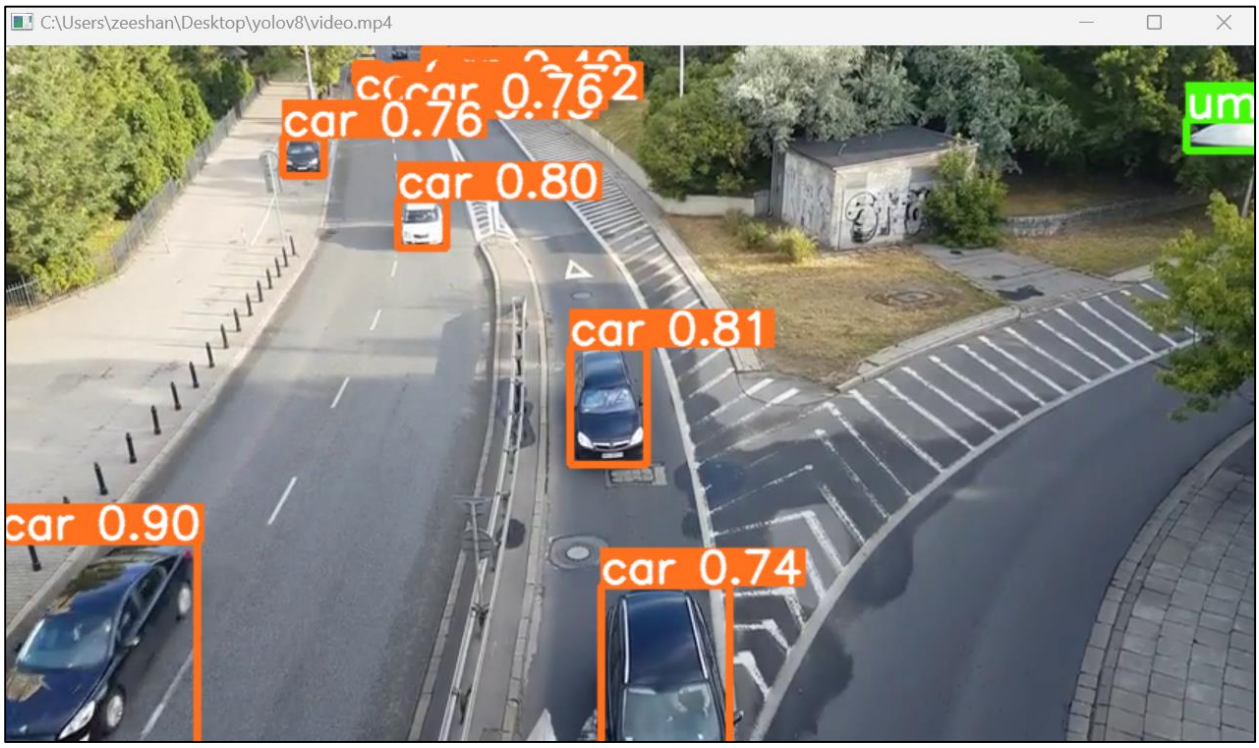
```
>>yolo task=detect mode=predict model=yolov8s.pt source=0
```

Model	size (pixels)	mAP^{val} 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

Object Detection on MP4 Video:

```
>>yolo task=detect mode=predict model=yolov8s.pt source=video.mp4 show=true
```

Result:



Object Detection on directory:

```
yolo task=detect mode=predict model=yolov8s.pt source='C:\Users\zeeshan\Desktop\yolov8'
```

It will save the results in one go.

Result:

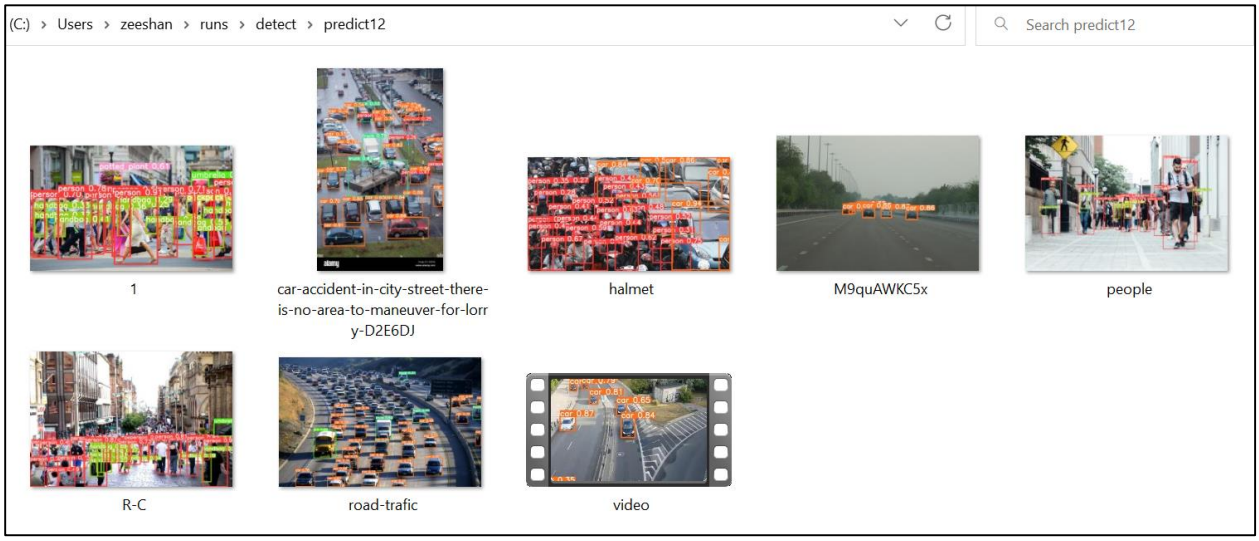


Image Segmentation using YOLO

v8

Image Classification

Export models into ONXX and TFlite