

```
!pip install --upgrade setuptools pip --user
!pip install onnx
!pip install onnxruntime
#!pip install --ignore-installed PyYAML
#!pip install Pillow

!pip install protobuf<4.21.3
!pip install onnxruntime-gpu
!pip install onnx>=1.9.0
!pip install onnx-simplifier>=0.3.6 --user
```

```
Requirement already satisfied: setuptools in /root/.local/lib/python3.10/site-packages (68.2.2)
Requirement already satisfied: pip in /root/.local/lib/python3.10/site-packages (23.3)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is
Requirement already satisfied: onnx in /usr/local/lib/python3.10/dist-packages (1.14.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from onnx) (1.23.5)
Requirement already satisfied: protobuf>=3.20.2 in /usr/local/lib/python3.10/dist-packages (from onnx) (3.20.3)
Requirement already satisfied: typing-extensions>=3.6.2.1 in /usr/local/lib/python3.10/dist-packages (from onnx) (4.5.0)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is
Requirement already satisfied: onnxruntime in /usr/local/lib/python3.10/dist-packages (1.16.1)
Requirement already satisfied: coloredlogs in /usr/local/lib/python3.10/dist-packages (from onnxruntime) (15.0.1)
Requirement already satisfied: flatbuffers in /usr/local/lib/python3.10/dist-packages (from onnxruntime) (23.5.26)
Requirement already satisfied: numpy>=1.21.6 in /usr/local/lib/python3.10/dist-packages (from onnxruntime) (1.23.5)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from onnxruntime) (23.2)
Requirement already satisfied: protobuf in /usr/local/lib/python3.10/dist-packages (from onnxruntime) (3.20.3)
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages (from onnxruntime) (1.12)
Requirement already satisfied: humanfriendly>=9.1 in /usr/local/lib/python3.10/dist-packages (from coloredlogs->onnxruntime) (10.0)
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-packages (from sympy->onnxruntime) (1.3.0)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is
/bin/bash: line 1: 4.21.3: No such file or directory
Requirement already satisfied: onnxruntime-gpu in /usr/local/lib/python3.10/dist-packages (1.16.1)
Requirement already satisfied: coloredlogs in /usr/local/lib/python3.10/dist-packages (from onnxruntime-gpu) (15.0.1)
Requirement already satisfied: flatbuffers in /usr/local/lib/python3.10/dist-packages (from onnxruntime-gpu) (23.5.26)
Requirement already satisfied: numpy>=1.21.6 in /usr/local/lib/python3.10/dist-packages (from onnxruntime-gpu) (1.23.5)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from onnxruntime-gpu) (23.2)
Requirement already satisfied: protobuf in /usr/local/lib/python3.10/dist-packages (from onnxruntime-gpu) (3.20.3)
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages (from onnxruntime-gpu) (1.12)
Requirement already satisfied: humanfriendly>=9.1 in /usr/local/lib/python3.10/dist-packages (from coloredlogs->onnxruntime-gpu) (10.0)
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-packages (from sympy->onnxruntime-gpu) (1.3.0)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is
```

```
import sys
import torch
print(f"Python version: {sys.version}, {sys.version_info} ")
print(f"Pytorch version: {torch.__version__} ")
```

```
Python version: 3.10.12 (main, Jun 11 2023, 05:26:28) [GCC 11.4.0], sys.version_info(major=3, minor=10, micro=12, releaselevel='final',
Pytorch version: 2.0.1+cu118
```

```
!nvidia-smi
```

```
Thu Oct 19 07:59:07 2023
```

```
+-----+
| NVIDIA-SMI 525.105.17    Driver Version: 525.105.17    CUDA Version: 12.0     |
+-----+-----+
| GPU   Name               Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf    Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
|                                           MIG M.         |
+-----+-----+
|  0  Tesla T4             Off          | 00000000:00:04.0 Off |                    0 |
| N/A   72C    P0       31W /  70W    | 1351MiB / 15360MiB |      0%      Default |
+-----+-----+
```

```
+-----+
| Processes:                                                       GPU Memory |
|  GPU   GI    CI          PID    Type   Process name                  Usage    |
|-----+-----+
+-----+
```

```
!# Download YOLOv7 code
!git clone https://github.com/WongKinYiu/yolov7
%cd yolov7
!ls
```

```
Cloning into 'yolov7'...
remote: Enumerating objects: 1191, done.
remote: Total 1191 (delta 0), reused 0 (delta 0), pack-reused 1191
Receiving objects: 100% (1191/1191), 74.22 MiB | 32.08 MiB/s, done.
Resolving deltas: 100% (515/515), done.
/content/yolov7/yolov7
cfg      detect.py  hubconf.py  models      requirements.txt  tools      utils
data     export.py  inference  paper       scripts          train_aux.py
deploy   figure     LICENSE.md README.md   test.py      train.py
```

```
!# Download trained weights
!wget https://github.com/WongKinYiu/yolov7/releases/download/v0.1/yolov7-tiny.pt
```

```
--2023-10-19 07:59:10-- https://github.com/WongKinYiu/yolov7/releases/download/v0.1/yolov7-tiny.pt
Resolving github.com (github.com)... 192.30.255.113
Connecting to github.com (github.com)|192.30.255.113|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/511187726/ba7d01ee-125a-4134-8864-fa1abcbf94d5?X-
--2023-10-19 07:59:11-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/511187726/ba7d01ee-125a-4134-8864-
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108.133, 185.199.109.133, 185.199.110.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 12639769 (12M) [application/octet-stream]
Saving to: 'yolov7-tiny.pt'

yolov7-tiny.pt      100%[=====] 12.05M  --.-KB/s  in 0.07s

2023-10-19 07:59:11 (163 MB/s) - 'yolov7-tiny.pt' saved [12639769/12639769]
```

YOLOv7 Inference

```
!python detect.py --weights ./yolov7-tiny.pt --conf 0.25 --img-size 640 --source inference/images/horses.jpg
```

```
Namespace(weights=['./yolov7-tiny.pt'], source='inference/images/horses.jpg', img_size=640, conf_thres=0.25, iou_thres=0.45, device='',
YOLOR 🚀 v0.1-126-g84932d7 torch 2.0.1+cu118 CUDA:0 (Tesla T4, 15101.8125MB)

Fusing layers...
Model Summary: 200 layers, 6219709 parameters, 229245 gradients
Convert model to Traced-model...
traced_script_module saved!
model is traced!

/usr/local/lib/python3.10/dist-packages/torch/functional.py:504: UserWarning: torch.meshgrid: in an upcoming release, it will be require
return _VF.meshgrid(tensors, **kwargs) # type: ignore[attr-defined]
5 horses, Done. (5.1ms) Inference, (1.7ms) NMS
The image with the result is saved in: runs/detect/exp/horses.jpg
Done. (0.171s)
```

```
from PIL import Image
Image.open('/content/yolov7/runs/detect/exp/horses.jpg')
```



ONNX INFERENCE



```
# export ONNX for ONNX inference
%cd /content/yolov7/
!python export.py --weights ./yolov7-tiny.pt \
    --grid --end2end --simplify \
    --topk-all 100 --iou-thres 0.65 --conf-thres 0.35 \
    --img-size 640 640 --max-wh 640 # For onnxruntime, you need to specify this value as an integer, when it is 0 it means agnostic NMS,
    # otherwise it is non-agnostic NMS

/content/yolov7
Import onnx_graphsurgeon failure: No module named 'onnx_graphsurgeon'
Namespace(weights='./yolov7-tiny.pt', img_size=[640, 640], batch_size=1, dynamic=False, dynamic_batch=False, grid=True, end2end=True, ma
YOLOR 🚀 v0.1-126-g84932d7 torch 2.0.1+cu118 CPU

Fusing layers...
Model Summary: 200 layers, 6219709 parameters, 6219709 gradients
/usr/local/lib/python3.10/dist-packages/torch/functional.py:504: UserWarning: torch.meshgrid: in an upcoming release, it will be require
    return _VF.meshgrid(tensors, **kwargs) # type: ignore[attr-defined]

Starting TorchScript export with torch 2.0.1+cu118...
/content/yolov7/models/yolo.py:52: TracerWarning: Converting a tensor to a Python boolean might cause the trace to be incorrect. We can'
    if self.grid[i].shape[2:4] != x[i].shape[2:4]:
TorchScript export success, saved as ./yolov7-tiny.torchscript.pt
CoreML export failure: No module named 'coremltools'

Starting TorchScript-Lite export with torch 2.0.1+cu118...
TorchScript-Lite export success, saved as ./yolov7-tiny.torchscript.ptl

Starting ONNX export with onnx 1.14.1...
onnxruntime
/usr/local/lib/python3.10/dist-packages/torch/nn/modules/module.py:831: UserWarning: The .grad attribute of a Tensor that is not a leaf
    if param.grad is not None:
/usr/local/lib/python3.10/dist-packages/torch/onnx/symbolic_opset9.py:5589: UserWarning: Exporting aten::index operator of advanced inde
    warnings.warn(
===== Diagnostic Run torch.onnx.export version 2.0.1+cu118 =====
verbose: False, log level: Level.ERROR
===== 0 NONE 0 NOTE 0 WARNING 0 ERROR =====

Starting to simplify ONNX...
ONNX export success, saved as ./yolov7-tiny.onnx

Export complete (9.91s). Visualize with https://github.com/lutzroeder/netron.
```

```
# show ONNX model
!ls

'='0.3.6'    export.py    paper        tools        yolov7-tiny.onnx
'='1.9.0'    figure      README.md    traced_model.pt yolov7-tiny.pt
cfg          hubconf.py  requirements.txt train_aux.py  yolov7-tiny.torchscript.pt
data         inference   runs         train.py      yolov7-tiny.torchscript.ptl
deploy      LICENSE.md  scripts      utils
detect.py   models     test.py      yolov7

import cv2
cuda = True
w = "/content/yolov7/yolov7-tiny.onnx"
img = cv2.imread('/content/yolov7/inference/images/horses.jpg')
```

```

import cv2
import time
import requests
import random
import numpy as np
import onnxruntime as ort
from PIL import Image
from pathlib import Path
from collections import OrderedDict, namedtuple

providers = ['CUDAExecutionProvider', 'CPUExecutionProvider'] if cuda else ['CPUExecutionProvider']
session = ort.InferenceSession(w, providers=providers)

def letterbox(im, new_shape=(640, 640), color=(114, 114, 114), auto=True, scaleup=True, stride=32):
    # Resize and pad image while meeting stride-multiple constraints
    shape = im.shape[:2] # current shape [height, width]
    if isinstance(new_shape, int):
        new_shape = (new_shape, new_shape)

    # Scale ratio (new / old)
    r = min(new_shape[0] / shape[0], new_shape[1] / shape[1])
    if not scaleup: # only scale down, do not scale up (for better val mAP)
        r = min(r, 1.0)

    # Compute padding
    new_unpad = int(round(shape[1] * r)), int(round(shape[0] * r))
    dw, dh = new_shape[1] - new_unpad[0], new_shape[0] - new_unpad[1] # wh padding

    if auto: # minimum rectangle
        dw, dh = np.mod(dw, stride), np.mod(dh, stride) # wh padding

    dw /= 2 # divide padding into 2 sides
    dh /= 2

    if shape[::-1] != new_unpad: # resize
        im = cv2.resize(im, new_unpad, interpolation=cv2.INTER_LINEAR)
    top, bottom = int(round(dh - 0.1)), int(round(dh + 0.1))
    left, right = int(round(dw - 0.1)), int(round(dw + 0.1))
    im = cv2.copyMakeBorder(im, top, bottom, left, right, cv2.BORDER_CONSTANT, value=color) # add border
    return im, r, (dw, dh)

names = ['person', 'bicycle', 'car', 'motorcycle', 'airplane', 'bus', 'train', 'truck', 'boat', 'traffic light',
        'fire hydrant', 'stop sign', 'parking meter', 'bench', 'bird', 'cat', 'dog', 'horse', 'sheep', 'cow',
        'elephant', 'bear', 'zebra', 'giraffe', 'backpack', 'umbrella', 'handbag', 'tie', 'suitcase', 'frisbee',
        'skis', 'snowboard', 'sports ball', 'kite', 'baseball bat', 'baseball glove', 'skateboard', 'surfboard',
        'tennis racket', 'bottle', 'wine glass', 'cup', 'fork', 'knife', 'spoon', 'bowl', 'banana', 'apple',
        'sandwich', 'orange', 'broccoli', 'carrot', 'hot dog', 'pizza', 'donut', 'cake', 'chair', 'couch',
        'potted plant', 'bed', 'dining table', 'toilet', 'tv', 'laptop', 'mouse', 'remote', 'keyboard', 'cell phone',
        'microwave', 'oven', 'toaster', 'sink', 'refrigerator', 'book', 'clock', 'vase', 'scissors', 'teddy bear',
        'hair drier', 'toothbrush']
colors = {name:[random.randint(0, 255) for _ in range(3)] for i,name in enumerate(names)}

img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

image = img.copy()
image, ratio, dw, dh = letterbox(image, auto=False)
image = image.transpose((2, 0, 1))
image = np.expand_dims(image, 0)
image = np.ascontiguousarray(image)

im = image.astype(np.float32)
im /= 255
im.shape

outname = [i.name for i in session.get_outputs()]
outname

inname = [i.name for i in session.get_inputs()]
inname

inp = {inname[0]:im}

# ONNX inference
outputs = session.run(outname, inp)[0]
outputs

```

```
array([[ 0.0000000e+00,  3.6190897e+02,  2.8389810e+02,  4.9353049e+02,
        3.9562729e+02,  1.7000000e+01,  9.2383695e-01],
       [ 0.0000000e+00, -1.0339508e+00,  2.6461755e+02,  2.6221350e+02,
        4.4826135e+02,  1.7000000e+01,  9.2106491e-01],
       [ 0.0000000e+00,  2.1546234e+02,  2.7049042e+02,  3.5089423e+02,
        4.1111603e+02,  1.7000000e+01,  7.6384616e-01],
       [ 0.0000000e+00, -9.6609497e-01,  2.6136026e+02,  1.2928018e+02,
        3.3445981e+02,  1.7000000e+01,  6.9170254e-01],
       [ 0.0000000e+00,  3.0596024e+02,  2.8081891e+02,  3.7848898e+02,
        3.7234491e+02,  1.7000000e+01,  4.6638149e-01]], dtype=float32)
```

```
ori_images = [img.copy()]
```

```
for i,(batch_id,x0,y0,x1,y1,cls_id,score) in enumerate(outputs):
    image = ori_images[int(batch_id)]
    box = np.array([x0,y0,x1,y1])
    box -= np.array(dwdh*2)
    box /= ratio
    box = box.round().astype(np.int32).tolist()
    cls_id = int(cls_id)
    score = round(float(score),3)
    name = names[cls_id]
    color = colors[name]
    name += ' '+str(score)
    cv2.rectangle(image,box[:2],box[2:],color,2)
    cv2.putText(image,name,(box[0], box[1] - 2),cv2.FONT_HERSHEY_SIMPLEX,0.75,[225, 255, 255],thickness=2)
```

```
Image.fromarray(ori_images[0])
```

