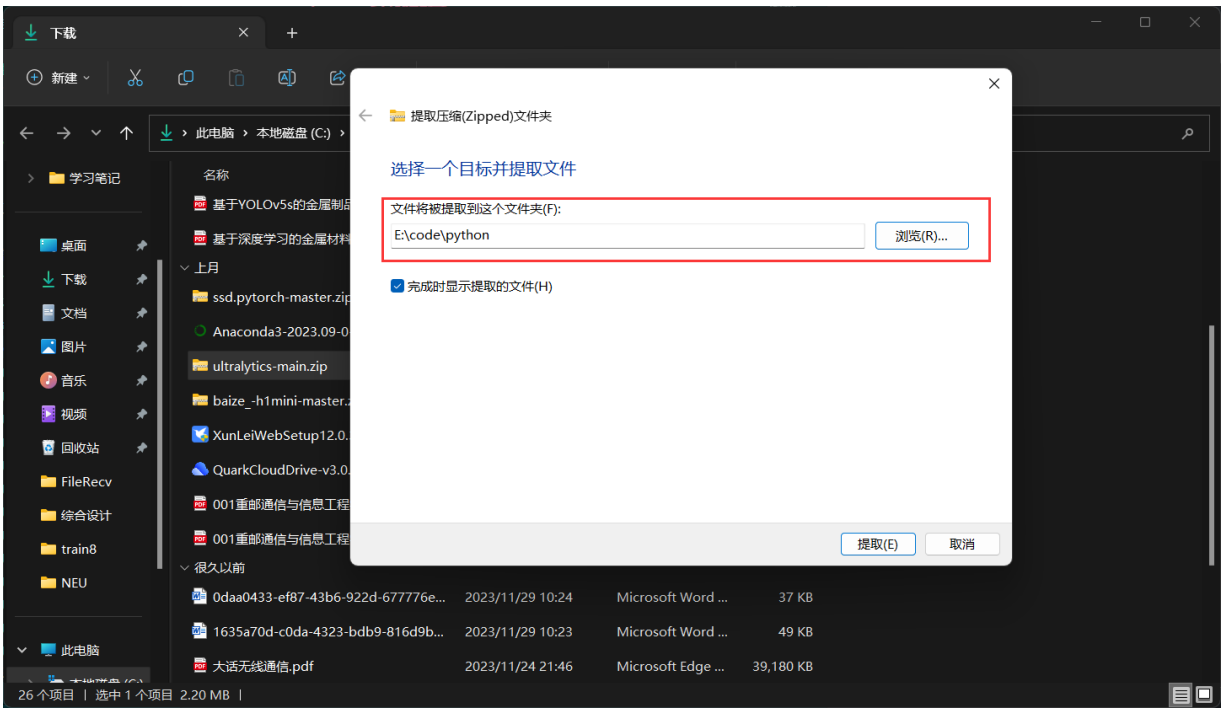
yolov8环境配置

1.安装源码

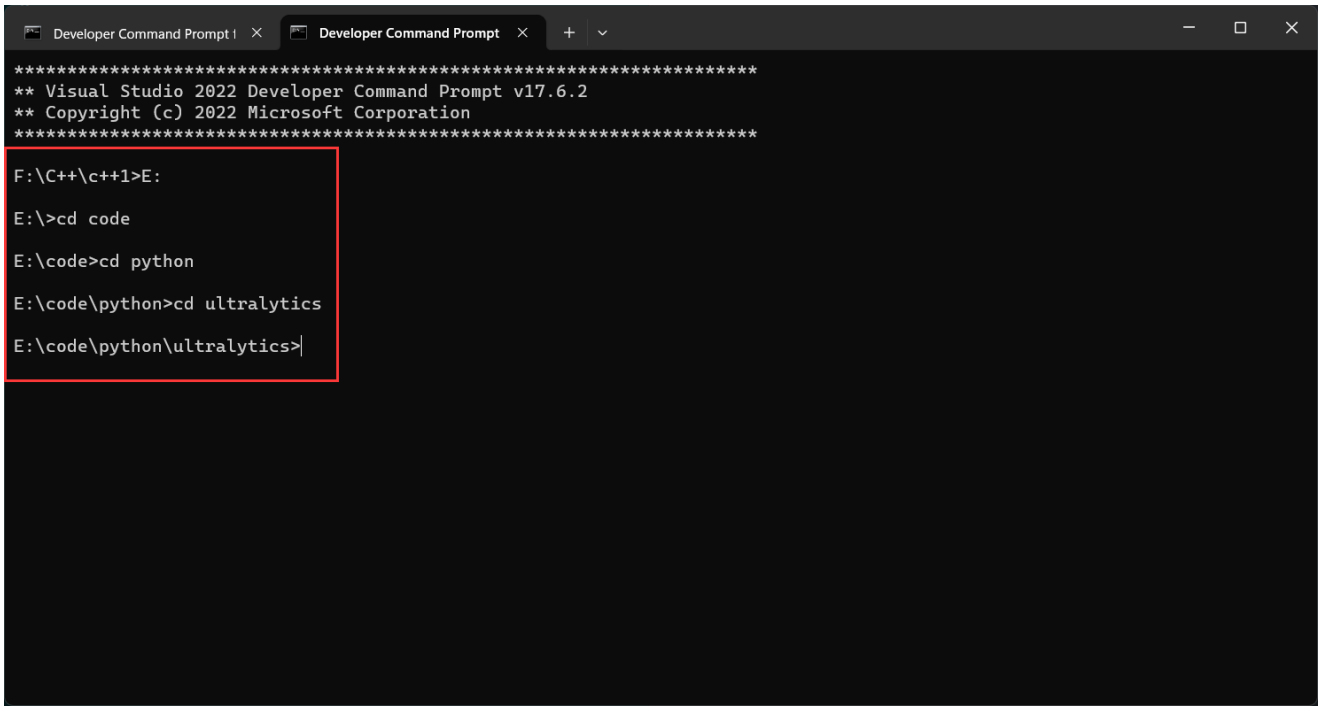
如果有魔法的就直接去网址 https://github.com/ultralytics/ultralytics 安装源码即可,没有的就用学长的网盘。
网盘链接：https://pan.baidu.com/s/1a2c3BKayXgQxHaq2qyrlMA
提取码：yolo

2.开始配置环境

将上述源码安装好后就解压缩，自己的路径要记好，下图只是一个示范



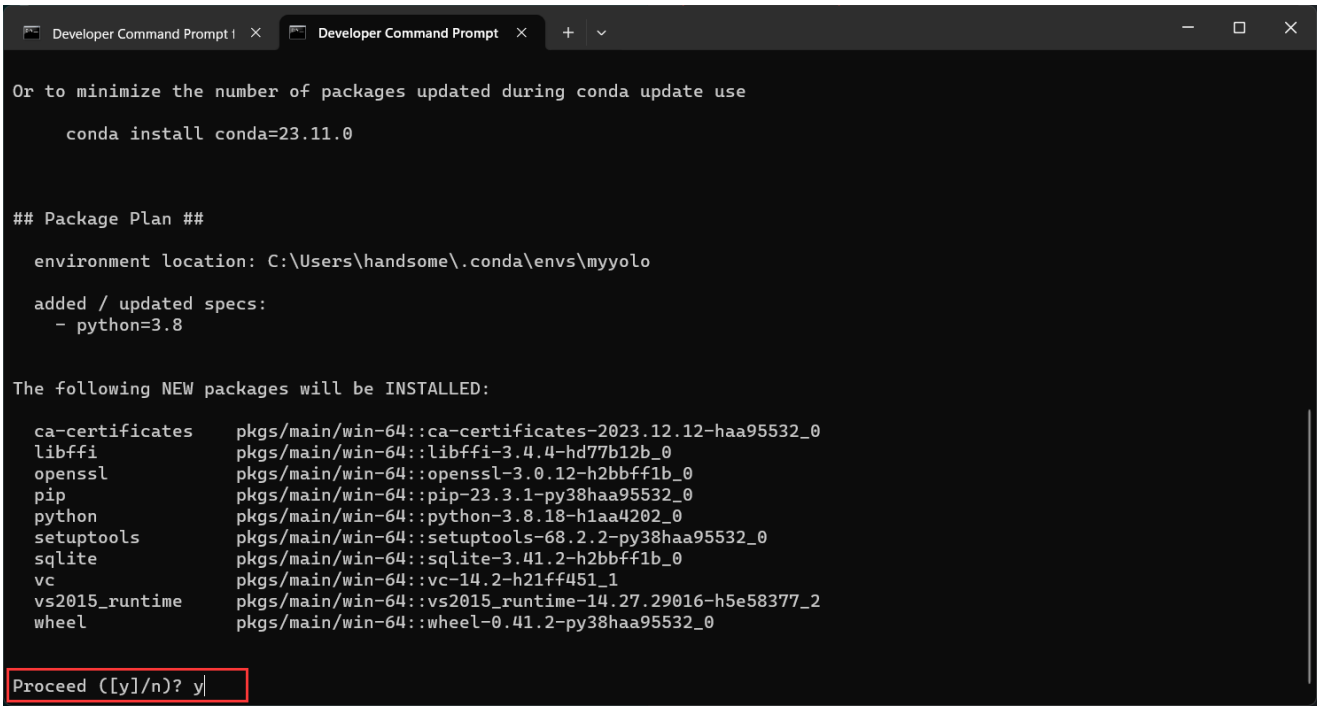
然后通过命令行转移到源码目录下



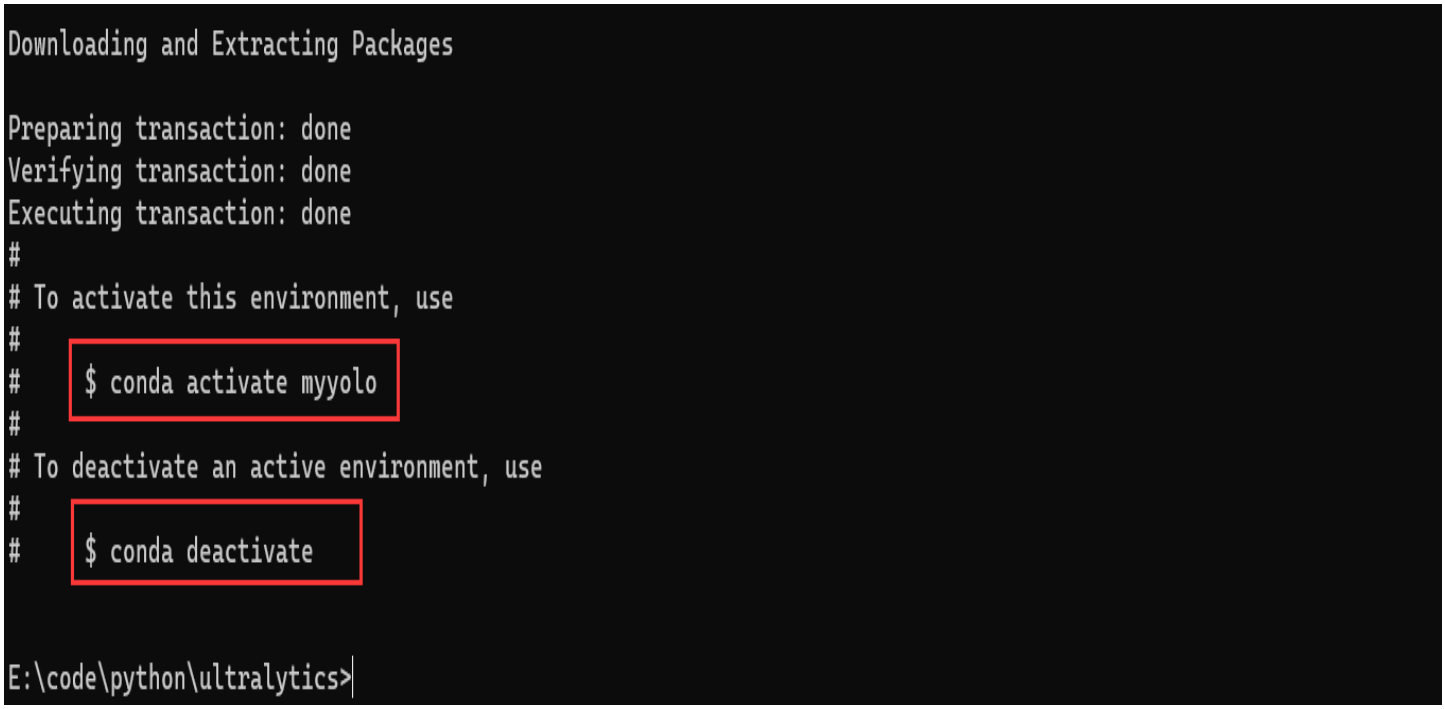
用 Anaconda 配置环境，先创建名称为 myyolo，python 版本为3.8的虚拟环境，

```
E:\code\python\ultralytics>
E:\code\python\ultralytics>conda create -n myyolo python=3.8
Retrieving notices: ...working... done
Collecting package metadata (current_repodata.json): done
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 23.7.4
  latest version: 23.11.0

Please update conda by running

    $ conda update -n base -c defaults conda

Or to minimize the number of packages updated during conda update use

     conda install conda=23.11.0



## Package Plan ##

  environment location: C:\Users\handsome\.conda\envs\myyolo

  added / updated specs:
    - python=3.8
```

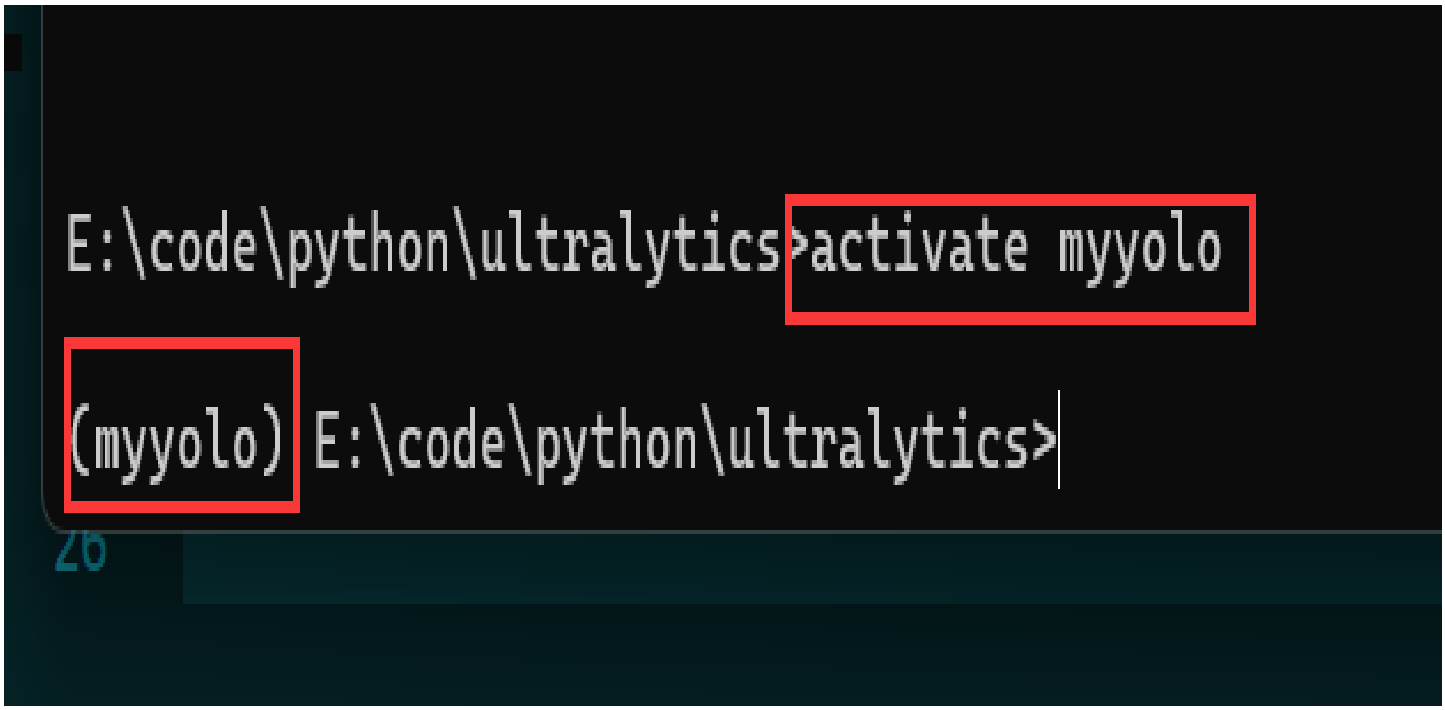
在 Proceed 这里输入 y (表达yes)即可,等待一会



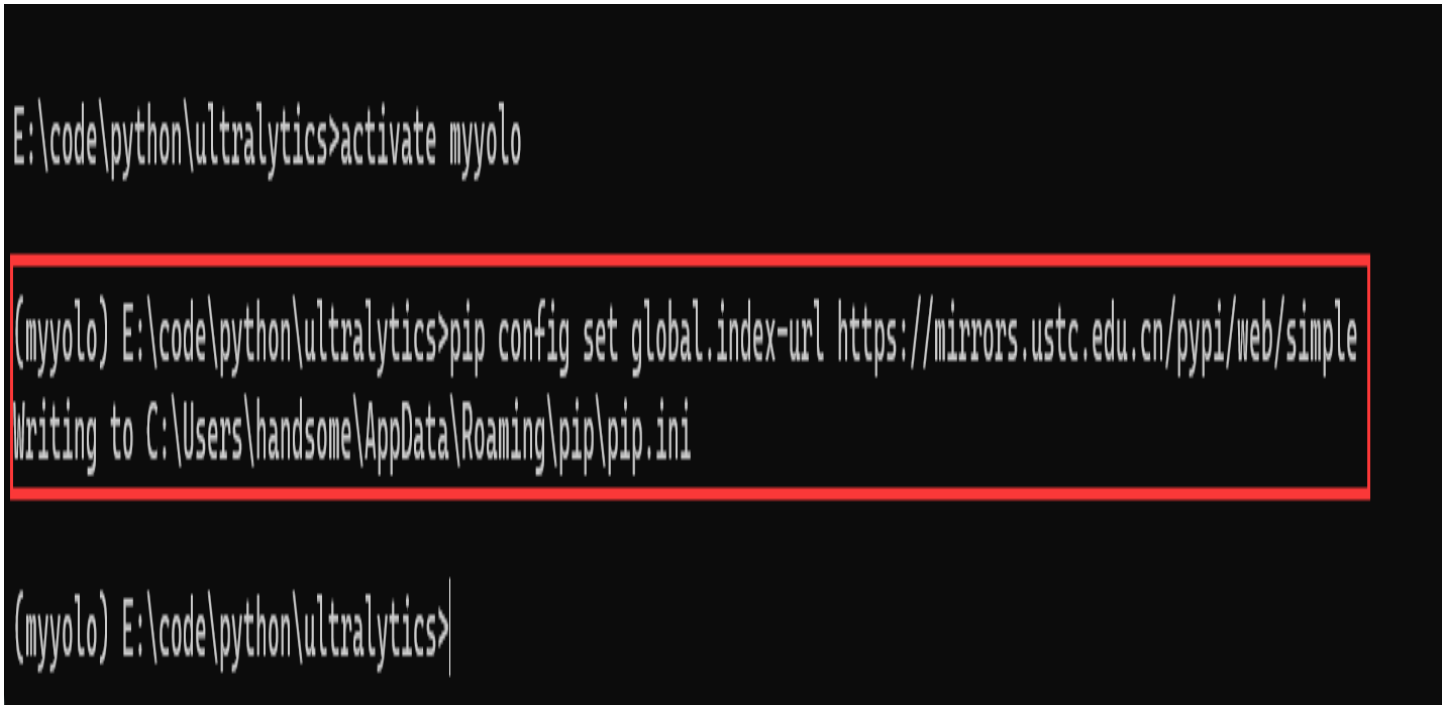
出现下图内容就表示虚拟环境创建成功



用命令 `activate myyolo` 如果不行就用 `conda activate myyolo` 打开虚拟环境,可以到看命令行右边就有 `myyolo` 标签



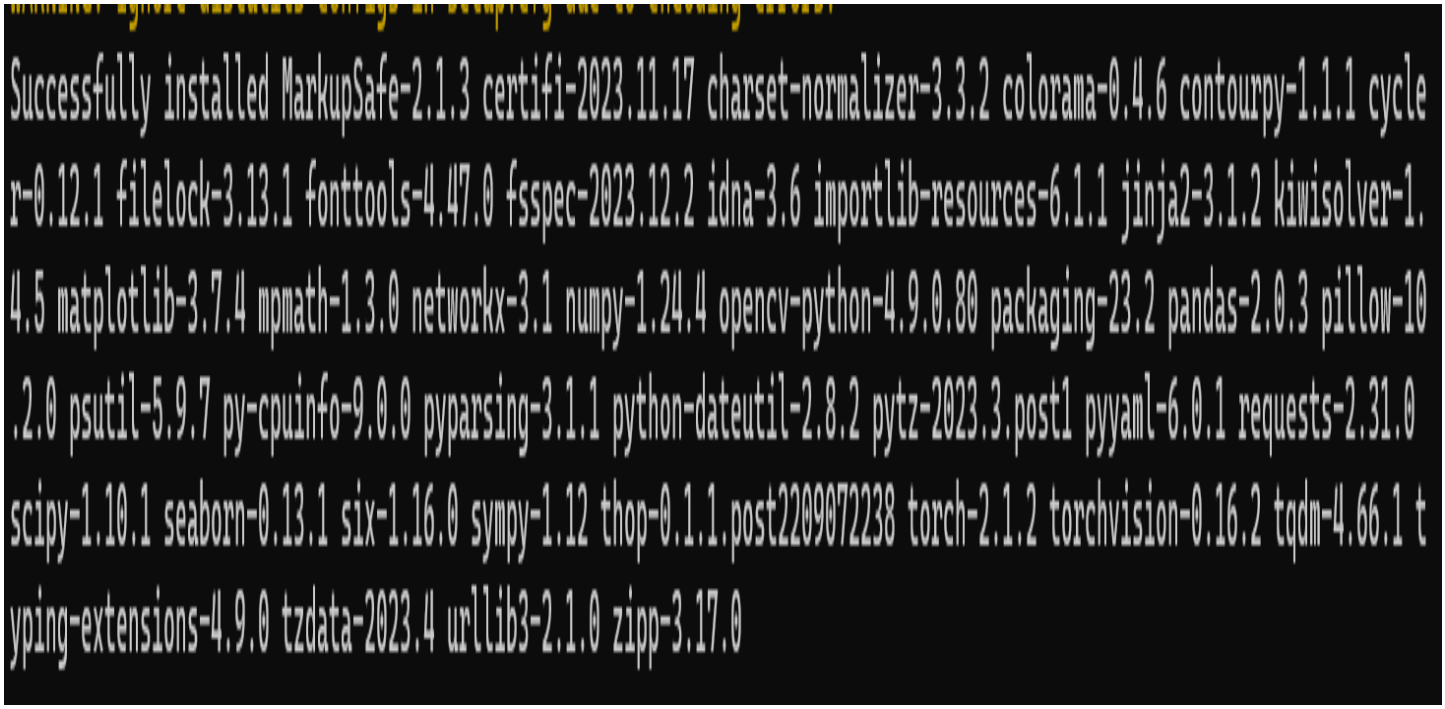
用 pip（一个 python 专用的依赖库下载工具)安装依赖,先设置好国内镜像源 pip config set global.index-url https://mirrors.ustc.edu.cn/pypi/web/simple



然后安装依赖:在当前目录下运行 pip install -r requirements.txt (该命令是让 pip 逐个安装 requirements.txt 文件中依赖库)

```
E:\code\python\ultralytics>activate myyolo

(myyolo) E:\code\python\ultralytics>pip config set global.index-url https://mirrors.ustc.edu.cn/pypi/web/simple
Writing to C:\Users\handsome\AppData\Roaming\pip\pip.ini

(myyolo) E:\code\python\ultralytics>pip install -r requirements.txt
```

之后就是漫长的等待,等待库安装成功,部分截图



```
  Downloading https://mirrors.bfsu.edu.cn/pypi/web/packages/93/e8/facde510585869b5ec694e8e0363ffe4eba067cb357a8398a55f6a
1f8023/importlib_resources-6.1.1-py3-none-any.whl (33 kB)
Collecting charset-normalizer<4,>=2 (from requests>=2.23.0->-r requirements.txt (line 10))
  Downloading https://mirrors.bfsu.edu.cn/pypi/web/packages/db/fb/d29e343e7c57bbf1231275939f6e75eb740cd47a9d7cb2c52ffeb6
2ef869/charset_normalizer-3.3.2-cp38-cp38-win_amd64.whl (99 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 99.6/99.6 kB 1.1 MB/s eta 0:00:00
Collecting idna<4,>=2.5 (from requests>=2.23.0->-r requirements.txt (line 10))
  Downloading https://mirrors.bfsu.edu.cn/pypi/web/packages/c2/e7/a82b05cf63a603df6e68d59ae6a68bf5064484a0718ea5033660af
4b54a9/idna-3.6-py3-none-any.whl (61 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 61.6/61.6 kB 1.1 MB/s eta 0:00:00
Collecting urllib3<3,>=1.21.1 (from requests>=2.23.0->-r requirements.txt (line 10))
  Downloading https://mirrors.bfsu.edu.cn/pypi/web/packages/96/94/c31f58c7a7f470d5665935262ebd7455c7e4c7782eb525658d3dbf
4b9403/urllib3-2.1.0-py3-none-any.whl (104 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 104.6/104.6 kB 2.0 MB/s eta 0:00:00
Collecting certifi>=2017.4.17 (from requests>=2.23.0->-r requirements.txt (line 10))
  Downloading https://mirrors.bfsu.edu.cn/pypi/web/packages/64/62/428ef076be88fa93716b576e4a01f919d25968913e817077a386fc
be4f42/certifi-2023.11.17-py3-none-any.whl (162 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 162.5/162.5 kB 1.2 MB/s eta 0:00:00
Collecting filelock (from torch>=1.8.0->-r requirements.txt (line 12))
```

然后就是安装必要依赖

```
(myyolo) E:\code\python\ultralytics>pip install ultralytics
WARNING: Ignore distutils configs in setup.cfg due to encoding errors.
Looking in indexes: https://mirrors.ustc.edu.cn/pypi/web/simple
Collecting ultralytics
  Downloading https://mirrors.bfsu.edu.cn/pypi/web/packages/2f/c7/ef29b1cfef781279ba46c6892440f08860822bf64798f348655298
85b99b/ultralytics-8.0.237-py3-none-any.whl (691 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 691.9/691.9 kB 908.9 kB/s eta 0:00:00
```

环境就安装完毕

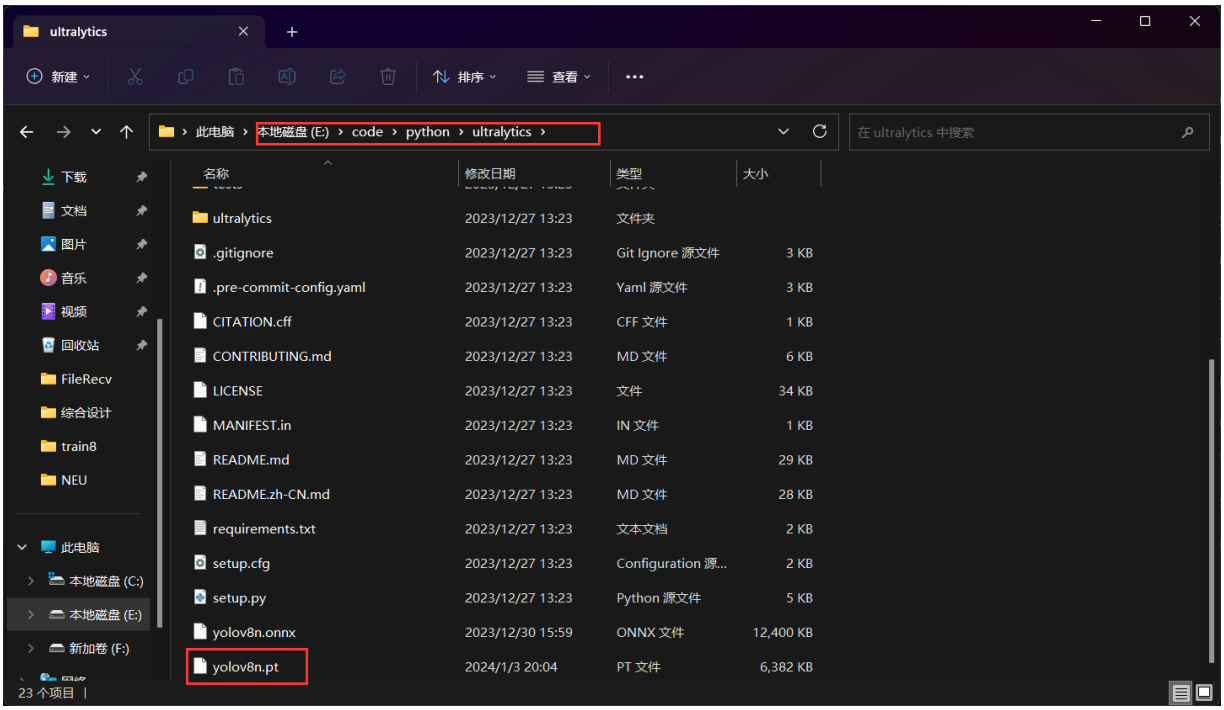
3.官方预训练模型测试

安装一个预训练模型权重,

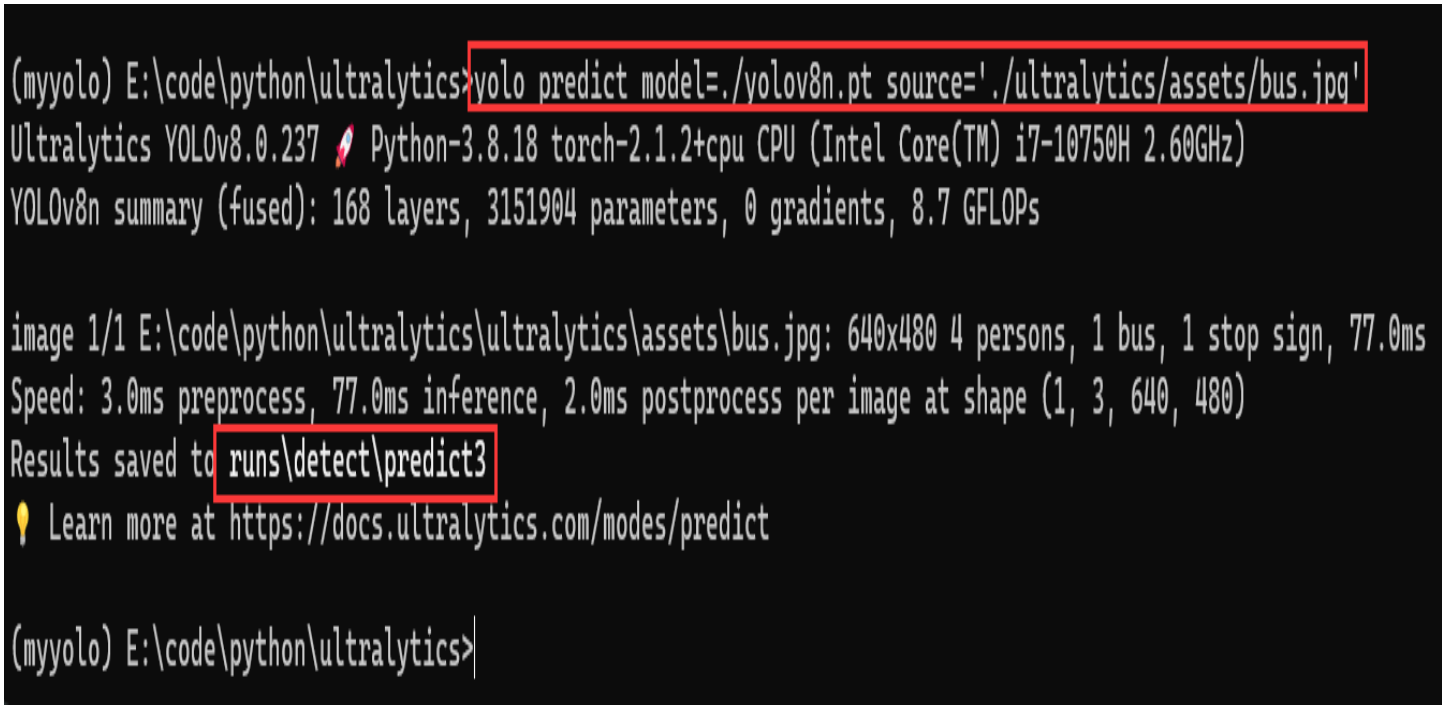
`yolov8n.pt`：https://github.com/ultralytics/assets/releases/download/v0.0.0/yolov8n.pt

`yolov8s.pt`：https://github.com/ultralytics/assets/releases/download/v0.0.0/yolov8s.pt

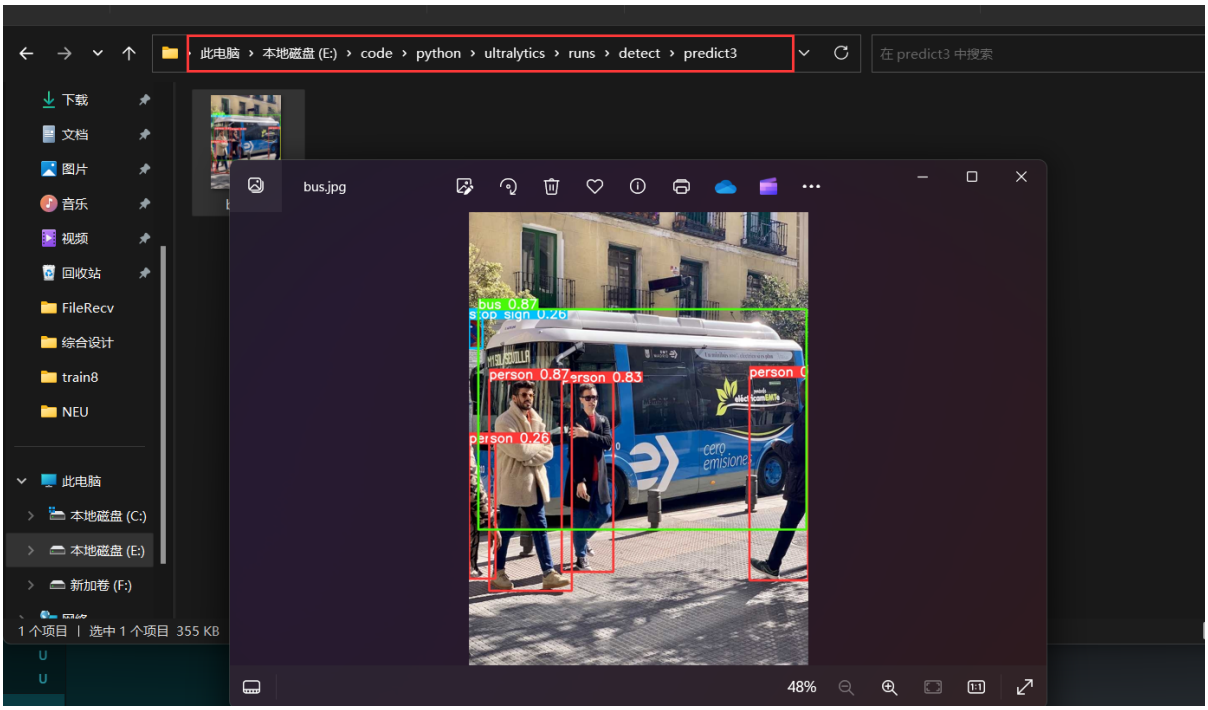
学长的百度网盘里面有 `yolov8n.pt` 文件,把它转移到源码目录下即可



通过命令行运行 `yolo predict model=./yolov8n.pt source='./ultralytics/assets/bus.jpg'` , `./ultralytics/assets/bus.jpg` 是在 yolov8 源码自带的图像



发现结果图像在当前目录下的 `runs/detect/predict3` (注意看目录,第一次生成应该是 `runs/detect/predict` ,学长的是因为已经生成过多次了),查看结果图



3.运行自己拍摄的图片

3.1 命令行

用自己的手机拍照,然后上传到电脑端,用命令

```
yolo predict model=./yolov8n.pt source=图像路径
```

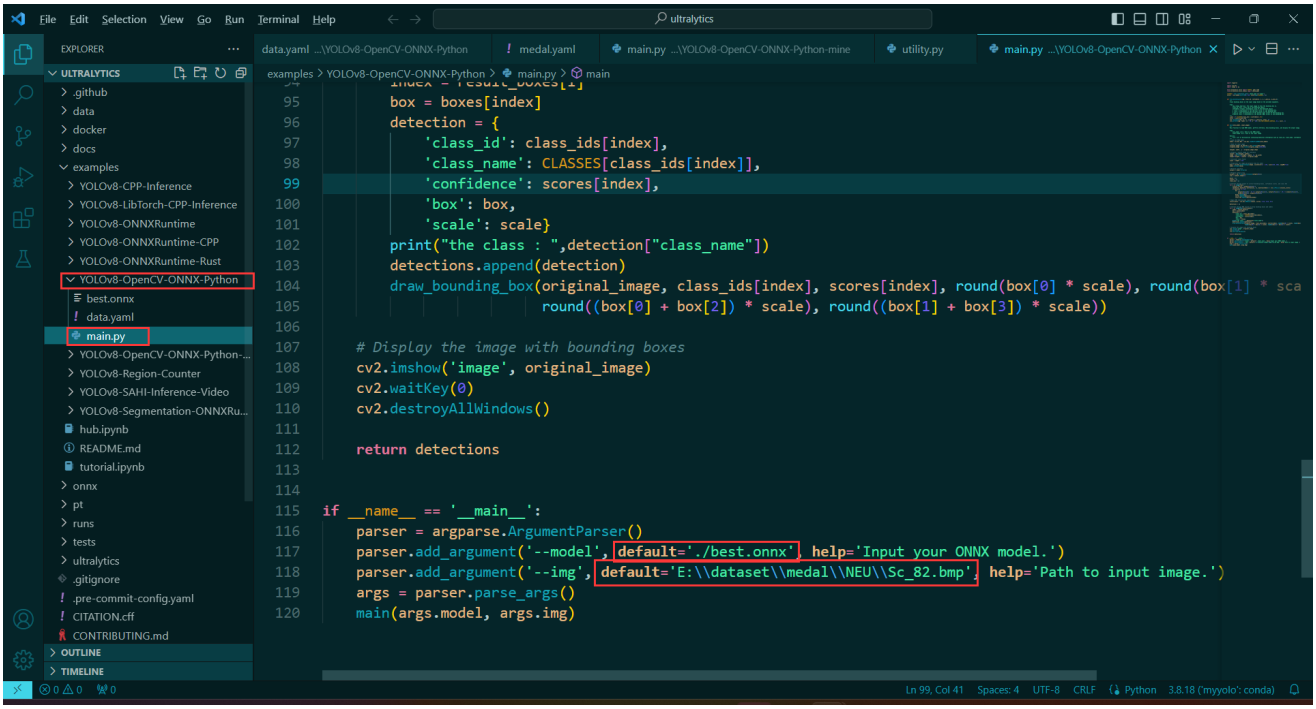
即可使用模型推理自己的数据。注意:要在源码目录下运行代码。

3.2 代码

在源码路径有 `example` 文件夹,而 `./examples/YOLOv8-OpenCV-ONNX-Python` 使用 `opencv-python` 和 `.onnx` 文件推理部署的;我们可以用该代码进行推理。
首先把 `.pt` 权重转换为 `.onnx` ,如下图

```
(myyolo) E:\code\python\ultralytics>yolo export model=yolov8n.pt format=onnx
Ultralytics YOLOv8.0.237 🚀 Python-3.8.18 torch-2.1.2+cpu CPU (Intel Core(TM) i7-10750H 2.60GHz)
YOLOv8n summary (fused): 168 layers, 3151904 parameters, 0 gradients, 8.7 GFLOPs

PyTorch: starting from 'yolov8n.pt' with input shape (1, 3, 640, 640) BCHW and output shape(s) (1, 84, 8400) (6.2 MB)
requirements: Ultralytics requirement ['onnx>=1.12.0'] not found, attempting AutoUpdate...
WARNING: Ignore distutils configs in setup.cfg due to encoding errors.
  WARNING: Ignore distutils configs in setup.cfg due to encoding errors.
  WARNING: Ignore distutils configs in setup.cfg due to encoding errors.
WARNING: Ignore distutils configs in setup.cfg due to encoding errors.
Looking in indexes: https://mirrors.ustc.edu.cn/pypi/web/simple
Collecting onnx>=1.12.0
  Downloading https://mirrors.bfsu.edu.cn/pypi/web/packages/98/b7/514a90cea77a4e80691f52e35c4486d31fe205771b9785f57535fb5e0
7c4/onnx-1.15.0-cp38-cp38-win_amd64.whl (14.3 MB)
     ---------------------------------------- 14.3/14.3 MB 892.7 kB/s eta 0:00:00
Requirement already satisfied: numpy in c:\users\handsome\.conda\envs\myyolo\lib\site-packages (from onnx>=1.12.0) (1.24.4)

Collecting protobuf>=3.20.2 (from onnx>=1.12.0)
  Downloading https://mirrors.bfsu.edu.cn/pypi/web/packages/dd/13/f8262eac2f44b643794d3783108ccefc8f87ef3929da45c474277a6e0
e95/protobuf-4.25.1-cp38-cp38-win_amd64.whl (413 kB)
     ---------------------------------------- 413.4/413.4 kB 2.2 MB/s eta 0:00:00
Installing collected packages: protobuf, onnx
Successfully installed onnx-1.15.0 protobuf-4.25.1

requirements: AutoUpdate success ✅ 31.8s, installed 1 package: ['onnx>=1.12.0']
requirements: ⚠️ Restart runtime or rerun command for updates to take effect


ONNX: starting export with onnx 1.15.0 opset 17...
ONNX: export success ✅ 32.8s, saved as 'yolov8n.onnx' (12.2 MB)

Export complete (35.4s)
Results saved to E:\code\python\ultralytics
Predict:         yolo predict task=detect model=yolov8n.onnx imgsz=640
Validate:        yolo val task=detect model=yolov8n.onnx imgsz=640 data=coco.yaml
Visualize:       https://netron.app
💡 Learn more at https://docs.ultralytics.com/modes/export
```

在当前目录下得到 .onnx 文件然后把该 .onnx 文件转移到 ./examples/YOLOv8-OpenCV-ONNX-Python 中,再用vscode或者直接打开 ./examples/YOLOv8-OpenCV-ONNX-Python 的 main.py 文件



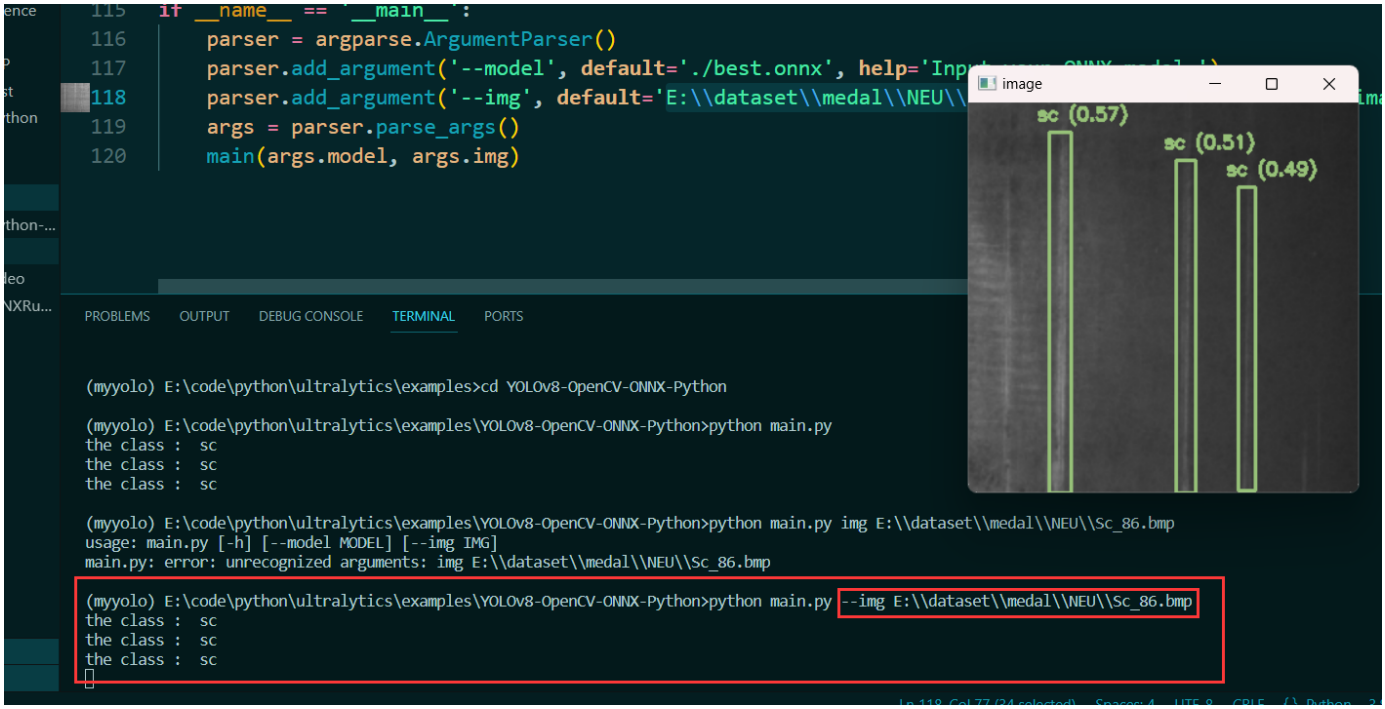
其中主要的代码如下,将主要代码改一下即可

```
parser = argparse.ArgumentParser()
## 设置你的onnx文件路径,这一行要改
parser.add_argument('--model', default='./best.onnx', help='Input your ONNX model.')
## 设置检测的图像,这一行要改
parser.add_argument('--img', default='E:\\dataset\\medal\\NEU\\Sc_82.bmp', help='Path to input
args = parser.parse_args()
main(args.model, args.img)
```

然后在命令行中运行 python main.py ,用默认参数运行

也可以在命令行中替换参数

```
## 修改图像路径
python main.py --img E:\\dataset\\medal\\NEU\\Sc_86.bmp
```



模板:

```
python main.py --model .onnx模型路径 --img 图像路径
```

3.3 命令行方法和代码方法比较

命令行方法比较快捷简便但是不能进行魔改，代码方法比较复杂但是你可以把内部代码进行魔改，添加pyqt模块和其他模型做成一个软件或者视频流算法。