YOLOv8環境的建置

1. 套件安裝
   1. anaconda軟體的下載網址，然後安裝anaconda軟體。

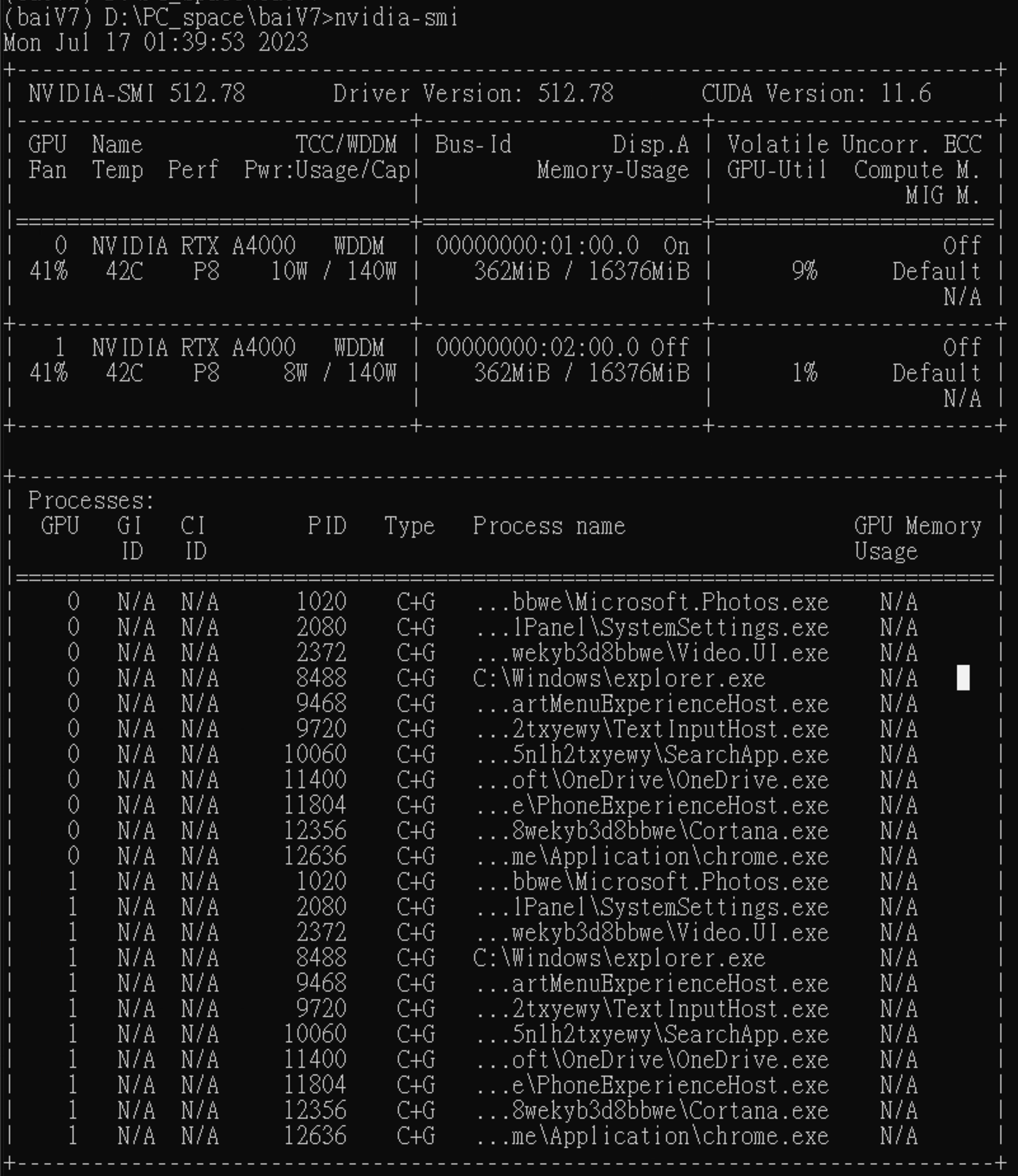
**https://www.anaconda.com/download**

* 1. conda 環境建立，使用python 3.8

**conda create --name baiV9 python=3.8**

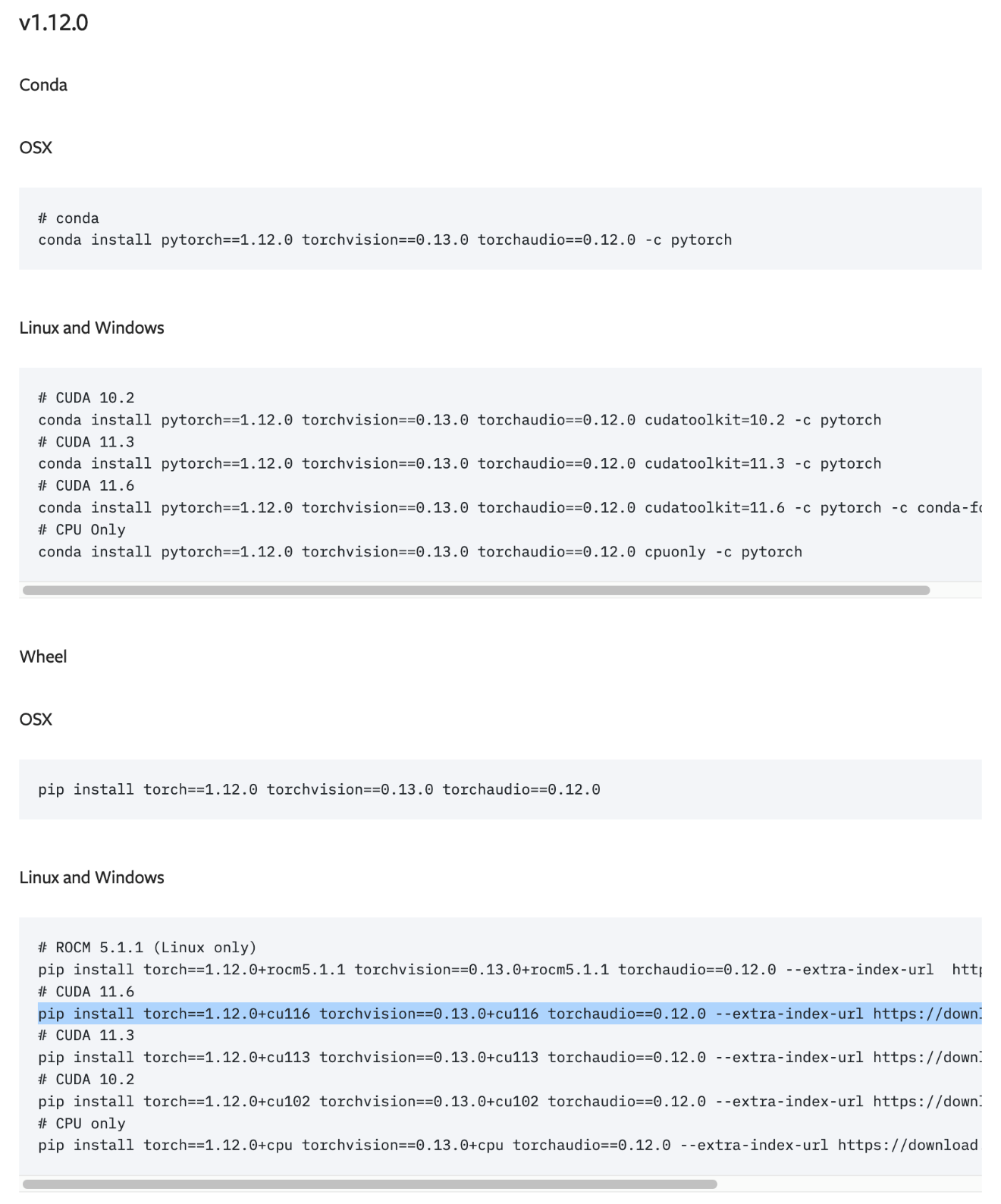
* 1. 取得nvidia版本的指，cmd -> **nvidia-smi**

**(感覺以下要去裝的pytorch版本跟下面框框這11.6沒有關係)**



* 1. 依序安裝以下的套件：
     + **~~pip install opencv-python pandas ipython psutil~~**
     + **~~pip install PyYAML PyQt5~~**
     + **pip install tqdm matplotlib seaborn tensorboard==2.12.0**
     + **pip install ultralytics**
     + Torch與cuXXX: 進入此網站，找到對應的版本 ->

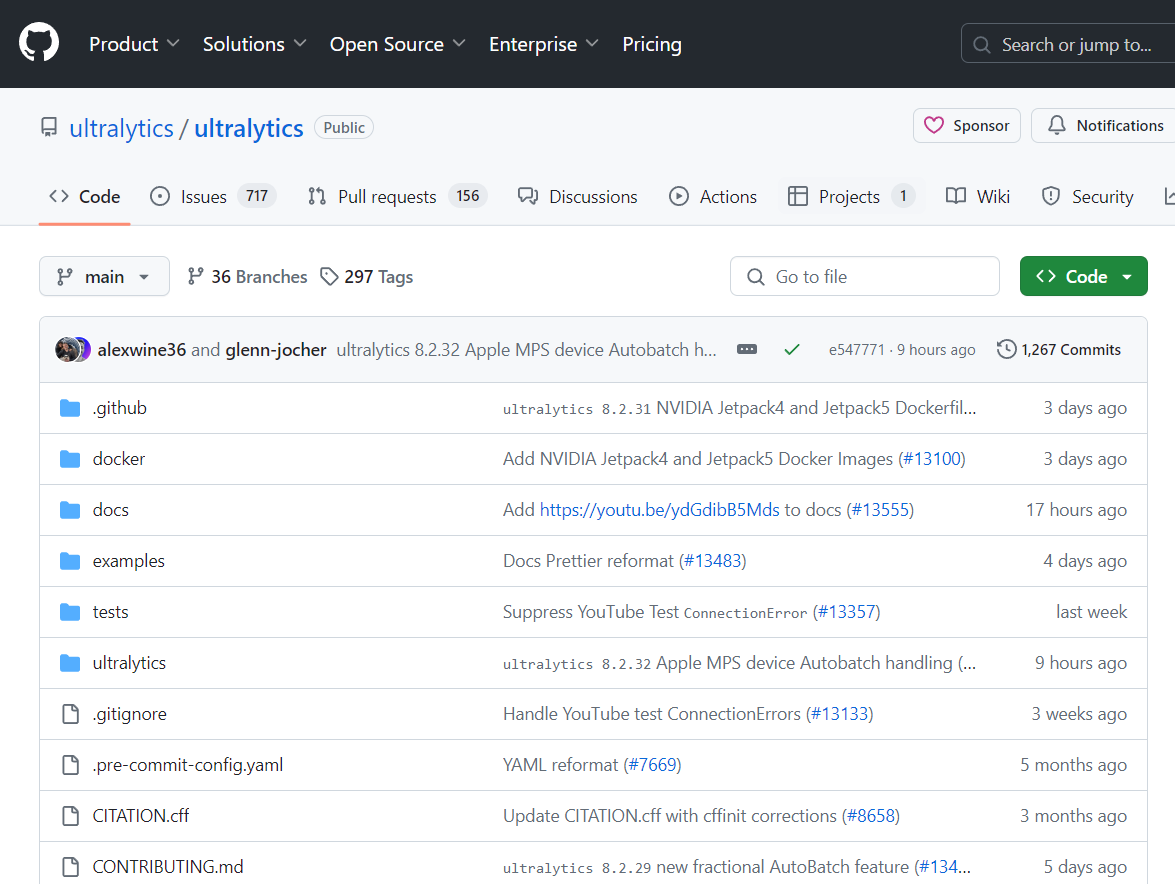
**https://pytorch.org/get-started/previous-versions/**



**pip install torch==1.12.0+cu116 torchvision==0.13.0+cu116 torchaudio==0.12.0 --extra-index-url https://download.pytorch.org/whl/cu116**

1. YOLOv8下載的地方

**https://github.com/ultralytics/ultralytics**



1. YOLOv7目錄結構 (**這部分還未修改**)

**yolov7-main folder**

**|-- cfg**

**| |-- baseline**

**| |-- deploy**

**| |-- training**

**| | |-- fruits.yaml**

**|-- data**

**| |-- fruits.yaml**

**| |-- hyp.scratch.p5.paml**

**| |-- hyp.scratch.p6.paml**

**|-- datasets(EX: fruits)**

**| |-- images**

**| | |-- train**

**| | |-- \*.jpg**

**| | |-- \*.jpg**

**| | | … …**

**| | | -- val**

**| | |-- \*.jpg**

**| | |-- \*.jpg**

**| | | … …**

**| |-- labels**

**| | |-- train**

**| | |-- \*.txt**

**| | |-- \*.txt**

**| | | … …**

**| | |-- val**

**| | |-- \*.txt**

**| | |-- \*.txt**

**| | | … …**

**| |-- train.txt**

**| |-- val.txt**

**|-- runs**

**| |-- detect**

**| |-- train**

**|-- detect.py**

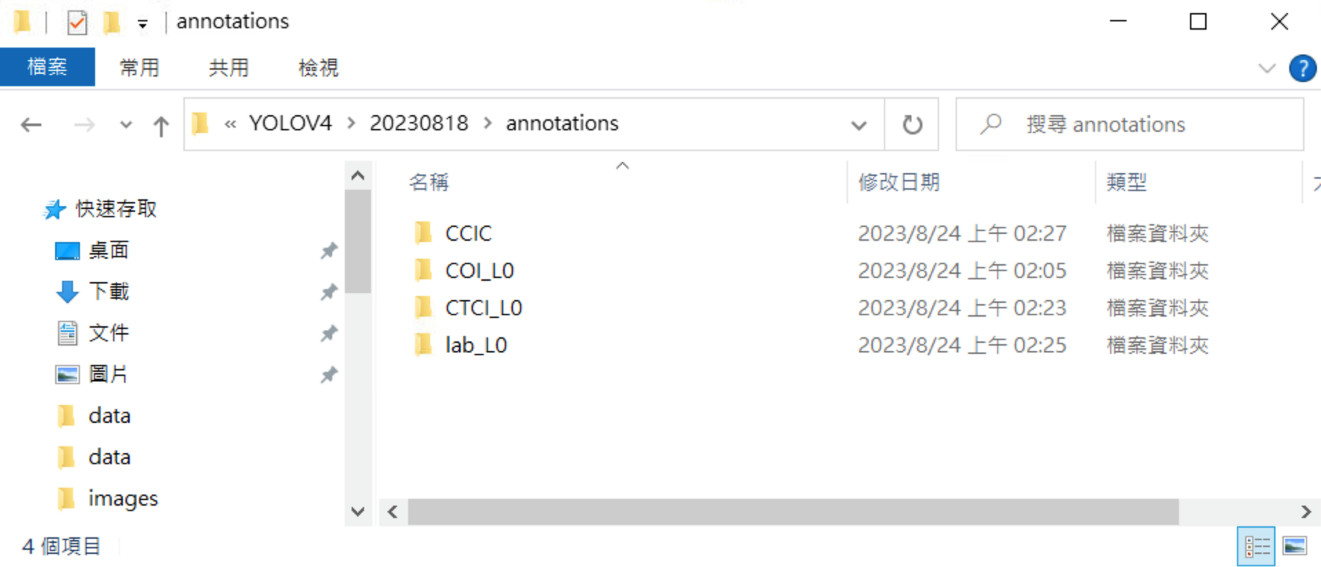
**|-- train.py**

**|-- test.py**

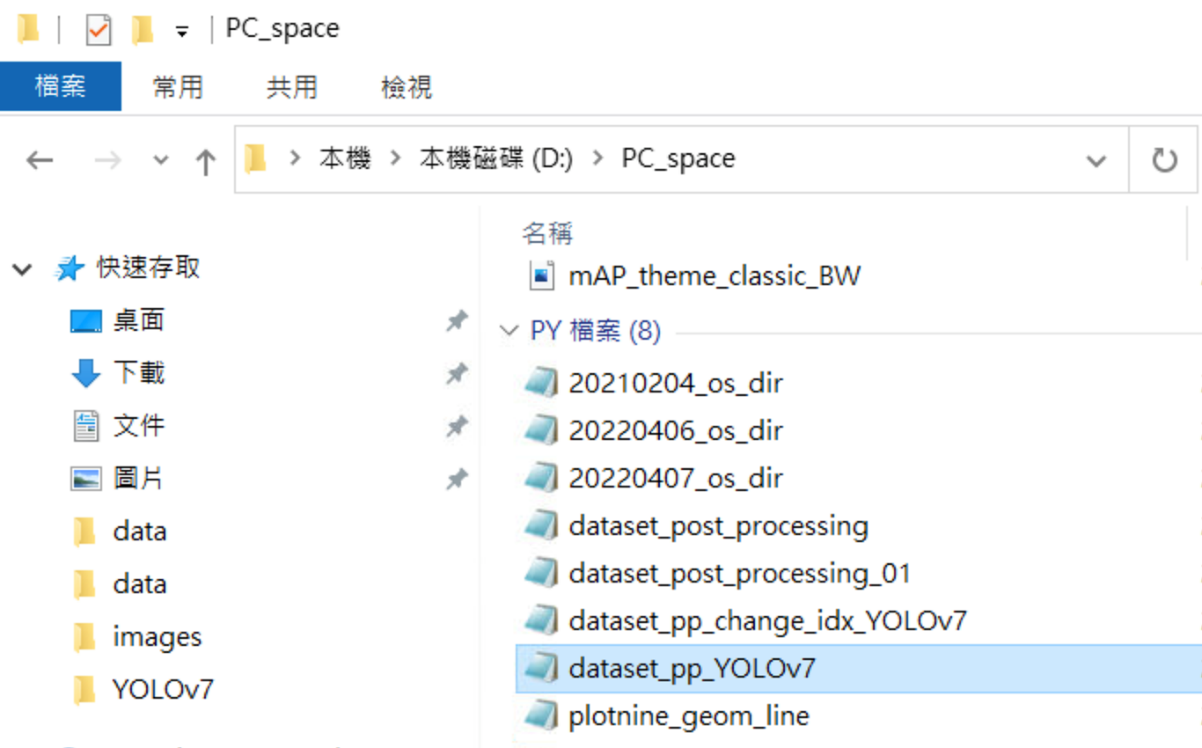
1. 訓練前的準備步驟：(**這部分還未修改**)

流程圖

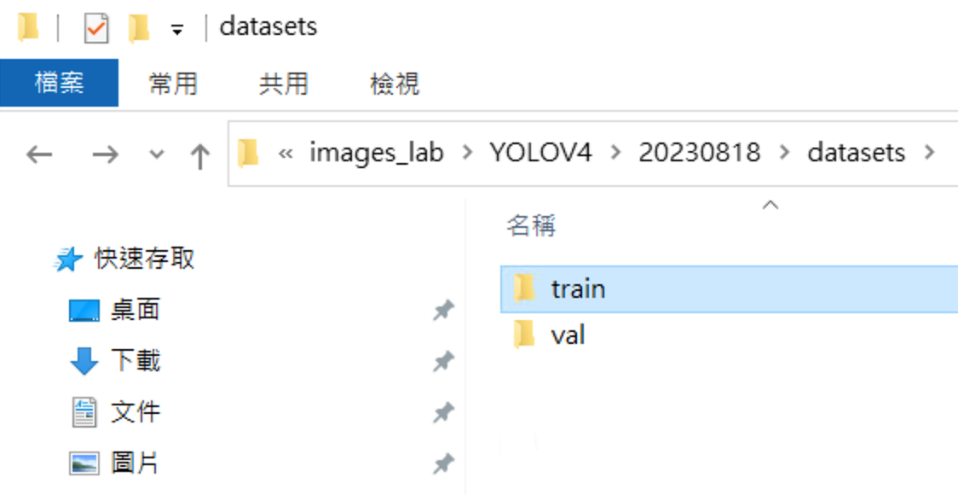
* 1. 在D槽作業，標註完成的資料夾內容複製至新創的資料夾。
     + 複製資料夾來源位置 -> D:\PC\_space\images\_lab\YOLOV4\20220615\**annotations**
     + 新創的資料夾範例 ->



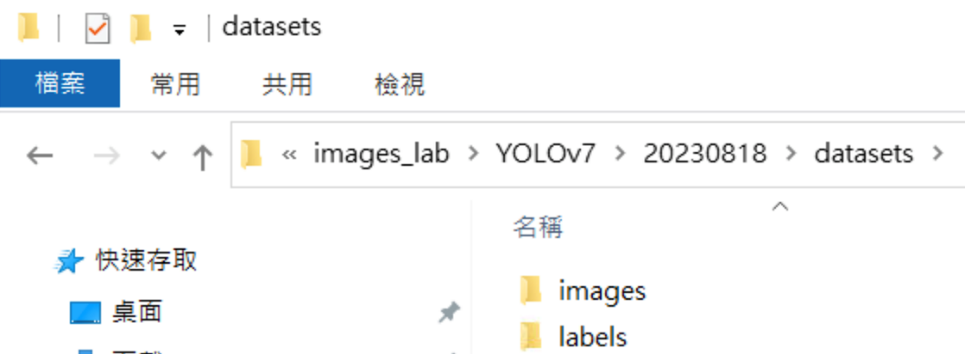
* + - continue
  1. 在D槽作業，建立datasets資料夾，包含train與val資料夾。
     + datasets資料夾需要手動先建立。
     + 執行程式 -> **dataset\_pp\_YOLOv7.py**



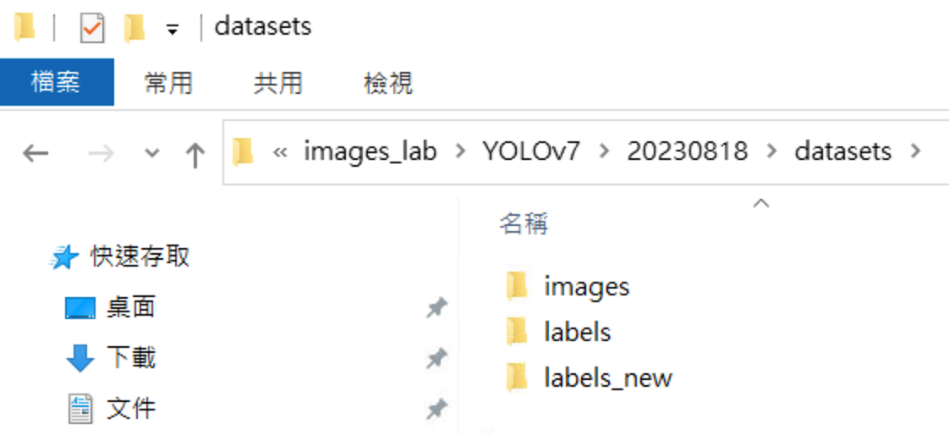
* + - 在D槽作業，產生的結果範例：



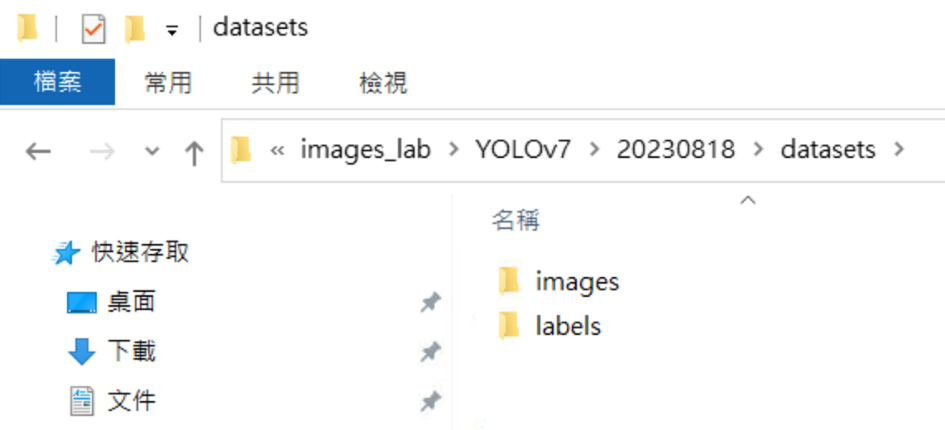
* + - continue
  1. 在D槽作業，複製產生的**datasets**資料夾內容至YOLOv7資料夾。
     + 目標位置 -> D:\PC\_space\images\_lab\YOLOv7
     + 結果範例：



* + - continue
  1. 在D槽作業，執行dataset\_pp\_change\_idx\_YOLOv7程式，將class index重新排列，並且產生程式的執行log檔案。
     + 程式位置 -> D:\PC\_space
     + 執行結果：



* + - Log檔案的產生位置在D:\PC\_space，名稱範例為20230818\_lable\_new\_log。
  1. 在E槽作業，將**images**與**label\_new**資料夾內容複製到欲訓練的位置。
     + 目標位置 -> E:\images\_lab\YOLOv7
     + 複製的內容為**images**與**label\_new**，其中label\_new複製完成後，將其改名為label，結果範例：



* + - continue
  1. 從D槽複製以下內容至E槽。
     + cfg，來源位置 -> D:\PC\_space\baiV7\yolov7\cfg\training，內容範例為20230818\_cfg\_dd\_0.yaml。
     + cfg，目標位置 -> E:\baiV7\yolov7\cfg\training
     + data，來源位置 -> D:\PC\_space\baiV7\yolov7\data，內容範例為20230818\_data\_dd\_0.yaml。
     + data，目標位置 -> E:\baiV7\yolov7\data
     + 訓練圖片的txt紀錄檔案，來源位置，範例為20230818\_train.txt與20230818\_val.txt -> D:\PC\_space\baiV7\yolov7\deterioration\_detection
     + 訓練圖片的txt紀錄檔案，目標位置 -> E:\baiV7\yolov7\**deterioration\_detection**
  2. 刪除既有的**\*.cache**檔案，範例為20230818\_train.cache與20230818\_val.cache，這樣才不會接續上次的訓練。
  3. 以上為訓練前的準備步驟。

1. Command for detect

**yolo predict model=yolov8n.pt source='https://ultralytics.com/images/bus.jpg'**

1. Command for training

**yolo detect train device=0 data=./ultralytics/cfg/datasets/20240613\_data\_dd\_04578.yaml model=./ultralytics/cfg/models/v8/yolov8l.yaml epochs=200 imgsz=640 project=runs/train name=yolov8\_l\_dd\_dataset2**

1. 訓練時用tensorboard看狀態，指令以及輸入到瀏覽器的網址：

**tensorboard --logdir runs/train -> dos cmd，要到YOLOv8的資料夾位置。**

**EX: (baiV8) E:\baiV8\ultralytics> tensorboard --logdir ./runs/train**

[**http://localhost:6006**](http://localhost:6006) **-> chrome**

1. Continue
2. Continue