

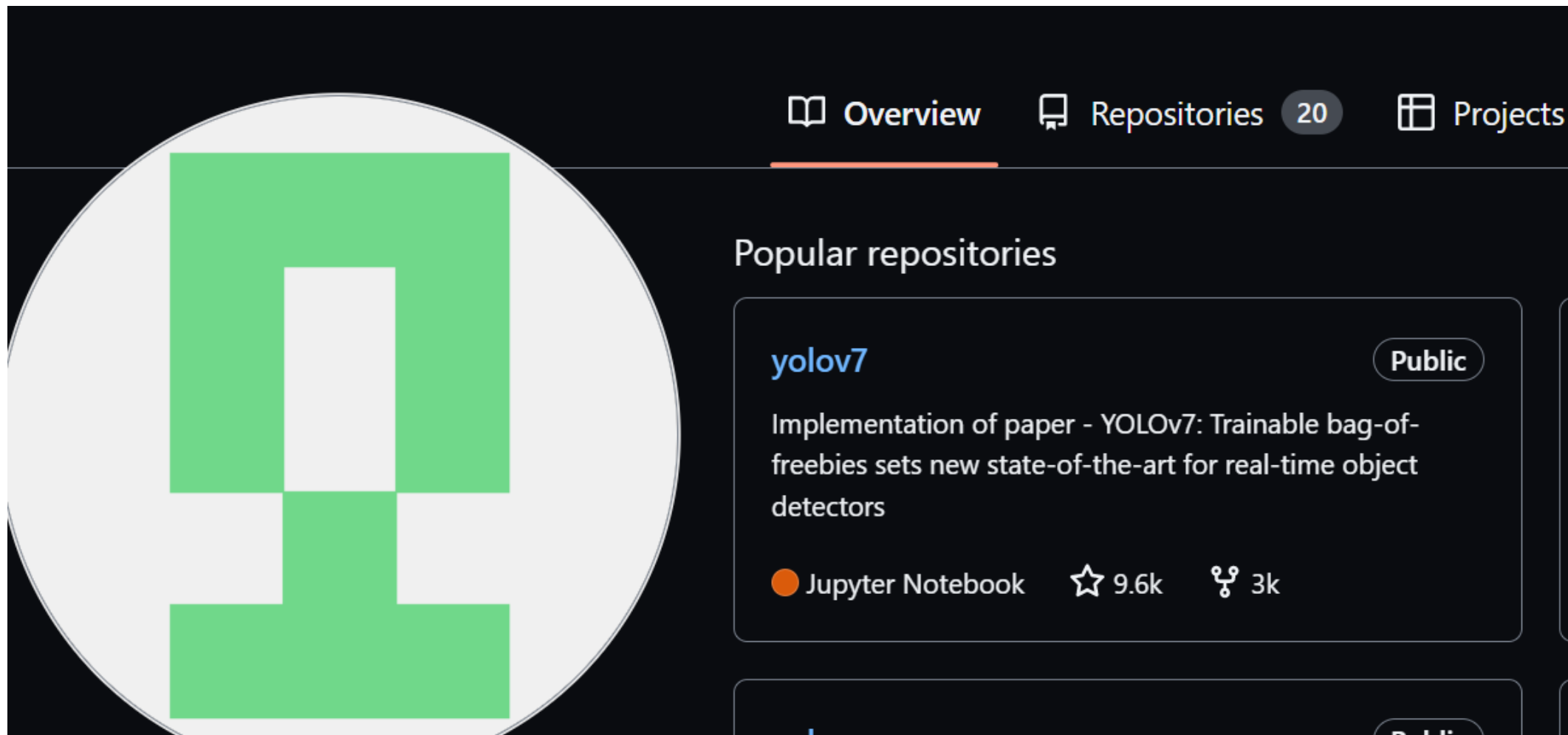
# YoloV7 GUI 步驟說明

1. 下載yolov7模組
2. 設置python虛擬環境
3. 安裝 yolov7 相關套件與 使用者介面的套件
4. 執行主程式
5. 介面功能說明

# Step1: 下載Yolov7封包

至作者的Github，點選yolov7 庫下載 yolov7 的封包

<https://github.com/WongKinYiu/yolov7>



進入 yolov7 庫，點擊Code，Download ZIP 下載

main

9 branches

1 tag

Go to file

Add file

Code

WongKinYiu Update README.md

cfgmain code

dataAdded para

deploy/triton-inference-s...Update REA

figuremain figure

inference/imagesSupport dy

modelsmain code

papermain code

Local

Codespaces

Clone

HTTPSSSHGitHub CLI

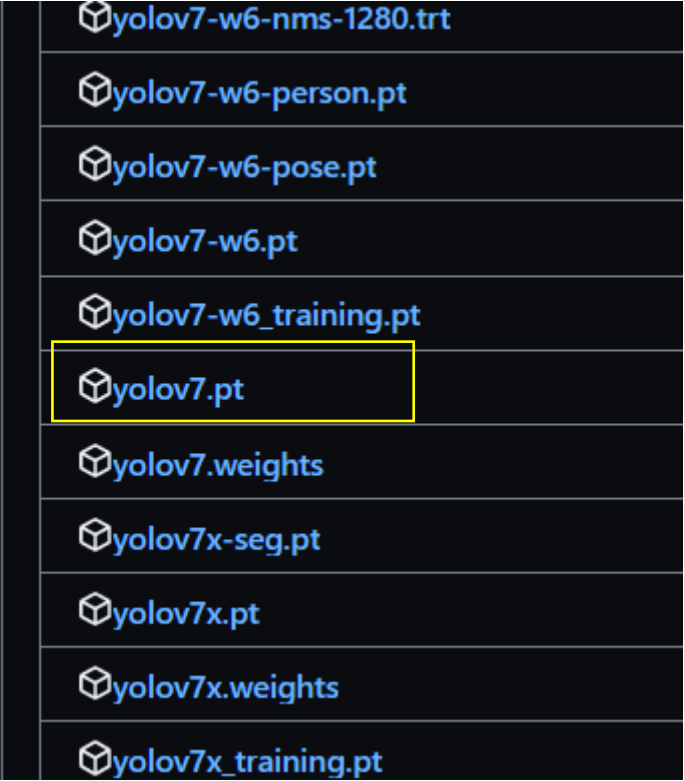
https://github.com/WongKinYiu/yolov7.git

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

在作者yolov7的releases中  
<https://github.com/WongKinYiu/yolov7/releases>，我選擇  
yolov7.pt來當訓練模型，一樣下載到yolov7路徑中



解壓縮作者的封包之後，  
點進去yolov7可以看到這樣的目錄

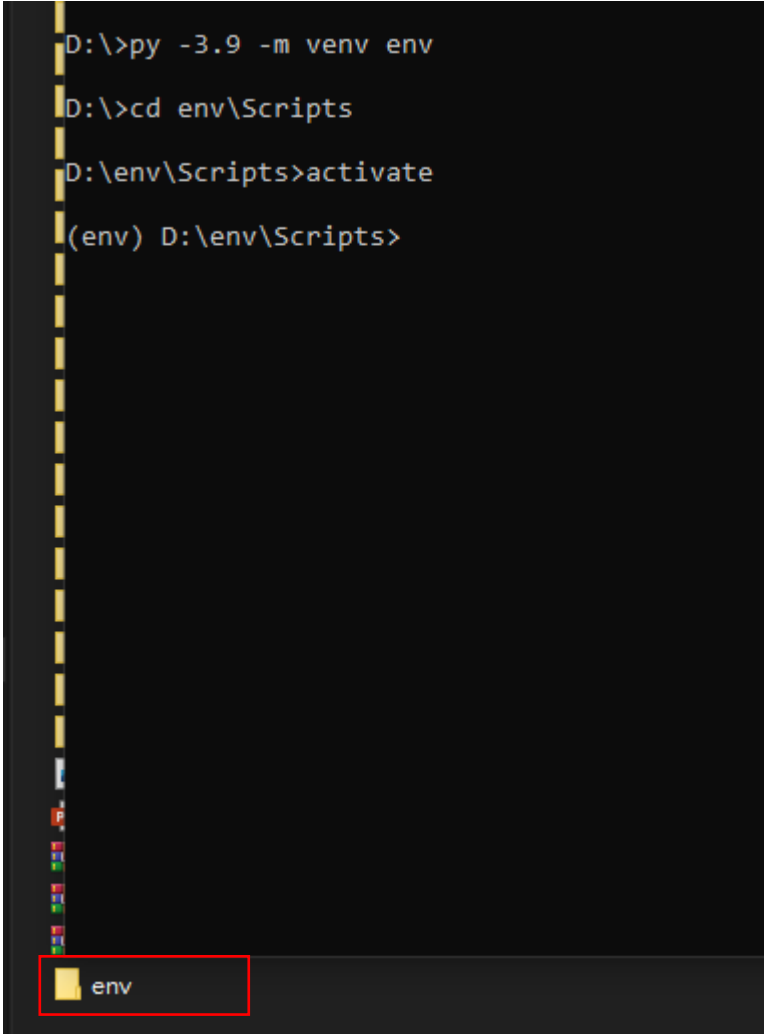
| 名稱               | 修改日期               | 類型              | 大小    |
|------------------|--------------------|-----------------|-------|
| inference        | 2023/5/25 下午 08:06 | 檔案資料夾           |       |
| models           | 2023/5/25 下午 08:06 | 檔案資料夾           |       |
| paper            | 2023/5/25 下午 08:06 | 檔案資料夾           |       |
| scripts          | 2023/5/25 下午 08:06 | 檔案資料夾           |       |
| tools            | 2023/5/25 下午 08:06 | 檔案資料夾           |       |
| utils            | 2023/5/25 下午 08:06 | 檔案資料夾           |       |
| .gitignore       | 2023/5/25 下午 08:06 | Git Ignore 來源檔案 | 4 KB  |
| detect.py        | 2023/5/25 下午 08:06 | Python 來源檔案     | 10 KB |
| export.py        | 2023/5/25 下午 08:06 | Python 來源檔案     | 9 KB  |
| hubconf.py       | 2023/5/25 下午 08:06 | Python 來源檔案     | 4 KB  |
| LICENSE.md       | 2023/5/25 下午 08:06 | MD 檔案           | 35 KB |
| README.md        | 2023/5/25 下午 08:06 | MD 檔案           | 14 KB |
| requirements.txt | 2023/5/25 下午 08:06 | 文字文件            | 1 KB  |
| test.py          | 2023/5/25 下午 08:06 | Python 來源檔案     | 17 KB |
| train.py         | 2023/5/25 下午 08:06 | Python 來源檔案     | 38 KB |
| train_aux.py     | 2023/5/25 下午 08:06 | Python 來源檔案     | 37 KB |

再將我提供的程式碼放在yolov7資料夾路徑內

| 本機 > Data (D:) > yolov7 |                    |             |            |  |
|-------------------------|--------------------|-------------|------------|--|
| 名稱                      | 修改日期               | 類型          | 大小         |  |
| .idea                   | 2023/5/31 上午 09:01 | 檔案資料夾       |            |  |
| __pycache__             | 2023/5/30 下午 04:42 | 檔案資料夾       |            |  |
| cfg                     | 2023/5/30 上午 10:39 | 檔案資料夾       |            |  |
| data                    | 2023/5/30 上午 10:39 | 檔案資料夾       |            |  |
| deploy                  | 2023/5/30 上午 10:39 | 檔案資料夾       |            |  |
| figure                  | 2023/5/30 上午 10:39 | 檔案資料夾       |            |  |
| inference               | 2023/5/30 上午 10:39 | 檔案資料夾       |            |  |
| models                  | 2023/5/30 上午 10:44 | 檔案資料夾       |            |  |
| paper                   | 2023/5/30 上午 10:39 | 檔案資料夾       |            |  |
| scripts                 | 2023/5/30 上午 10:39 | 檔案資料夾       |            |  |
| tools                   | 2023/5/30 上午 10:39 | 檔案資料夾       |            |  |
| utils                   | 2023/5/30 上午 10:44 | 檔案資料夾       |            |  |
| .gitignore              | 2023/1/2 上午 04:29  | 文字文件        | 4 KB       |  |
| detect.py               | 2023/1/2 上午 04:29  | Python File | 10 KB      |  |
| dialog.py               | 2023/5/31 上午 09:11 | Python File | 2 KB       |  |
| dialog_ui.py            | 2023/5/22 上午 10:04 | Python File | 4 KB       |  |
| dialog_ui.ui            | 2023/5/22 上午 10:04 | UI 檔案       | 4 KB       |  |
| error_dialog.py         | 2023/5/30 下午 12:14 | Python File | 1 KB       |  |
| error_dialog_ui.py      | 2023/3/31 上午 08:18 | Python File | 3 KB       |  |
| error_dialog_ui.ui      | 2023/3/31 上午 08:18 | UI 檔案       | 3 KB       |  |
| export.py               | 2023/1/2 上午 04:29  | Python File | 9 KB       |  |
| FrontEnd.py             | 2023/5/30 上午 10:24 | Python File | 1 KB       |  |
| globals.py              | 2023/5/22 下午 01:11 | Python File | 1 KB       |  |
| hubconf.py              | 2023/1/2 上午 04:29  | Python File | 4 KB       |  |
| LICENSE.md              | 2023/1/2 上午 04:29  | MD 檔案       | 35 KB      |  |
| main.py                 | 2023/5/31 上午 09:19 | Python File | 56 KB      |  |
| main_ui.py              | 2023/5/23 下午 01:07 | Python File | 12 KB      |  |
| main_ui.ui              | 2023/5/23 下午 01:07 | UI 檔案       | 13 KB      |  |
| Predict_view.py         | 2023/5/30 上午 11:07 | Python File | 21 KB      |  |
| Predict_view_ui.py      | 2023/5/16 上午 09:13 | Python File | 9 KB       |  |
| Predict_view_ui.ui      | 2023/5/18 下午 03:12 | UI 檔案       | 8 KB       |  |
| README.md               | 2023/1/2 上午 04:29  | MD 檔案       | 14 KB      |  |
| requirements.txt        | 2023/1/2 上午 04:29  | 文字文件        | 1 KB       |  |
| test.py                 | 2023/1/2 上午 04:29  | Python File | 17 KB      |  |
| thread_dialog.py        | 2023/5/30 上午 11:07 | Python File | 1 KB       |  |
| traced_model.pt         | 2023/5/30 上午 11:43 | PT 檔案       | 142,953 KB |  |
| train.py                | 2023/1/2 上午 04:29  | Python File | 38 KB      |  |
| train_aux.py            | 2023/1/2 上午 04:29  | Python File | 37 KB      |  |
| yolov7.pt               | 2023/1/30 下午 11:34 | PT 檔案       | 73,816 KB  |  |

## Step2.設置python虛擬環境

1. 使用高於 python3.9 的版本，這邊我使用的是 3.9.12 的版本
2. 開啟 cmd，這邊我在 D 槽創建虛擬環境，因為我有多個 python 版本，所以指令為 `py -3.9 -m venv env` 代表我是以 python 3.9 的版本來創建虛擬環境
  - 假設你的電腦只有一個版本，那麼指令為 `$ python -m venv env`
3. 然後可以在 D 槽看到一個 env 虛擬資料夾
4. 進入資料夾的 Scripts，使用 `activate` 指令將虛擬環境激活，在路徑的最前面看到 (env) 就是已經進入虛擬環境中了。



```
D:\>py -3.9 -m venv env
D:\>cd env\Scripts
D:\env\Scripts>activate
(env) D:\env\Scripts>
```

The screenshot shows a Windows command prompt window. The first three lines of commands are executed: `py -3.9 -m venv env`, `cd env\Scripts`, and `activate`. The fourth line shows the prompt has changed to `(env) D:\env\Scripts>`, indicating the virtual environment is active. At the bottom of the window, a taskbar shows a yellow icon labeled 'env' which is highlighted with a red rectangle.

## Step3.安裝 yolov7 相關套件與 使用者介面的套件


在虛擬環境內，進入yolov7資料夾，輸入 這個指令  
\$ pip install -r requirements.txt，安裝yolov7相關套件

```
(env) D:\env\Scripts>cd D:\yolov7  
(env) D:\yolov7>pip install -r requirements.txt
```

然後再安裝使用者介面的套件

\$ pip install pyqt5

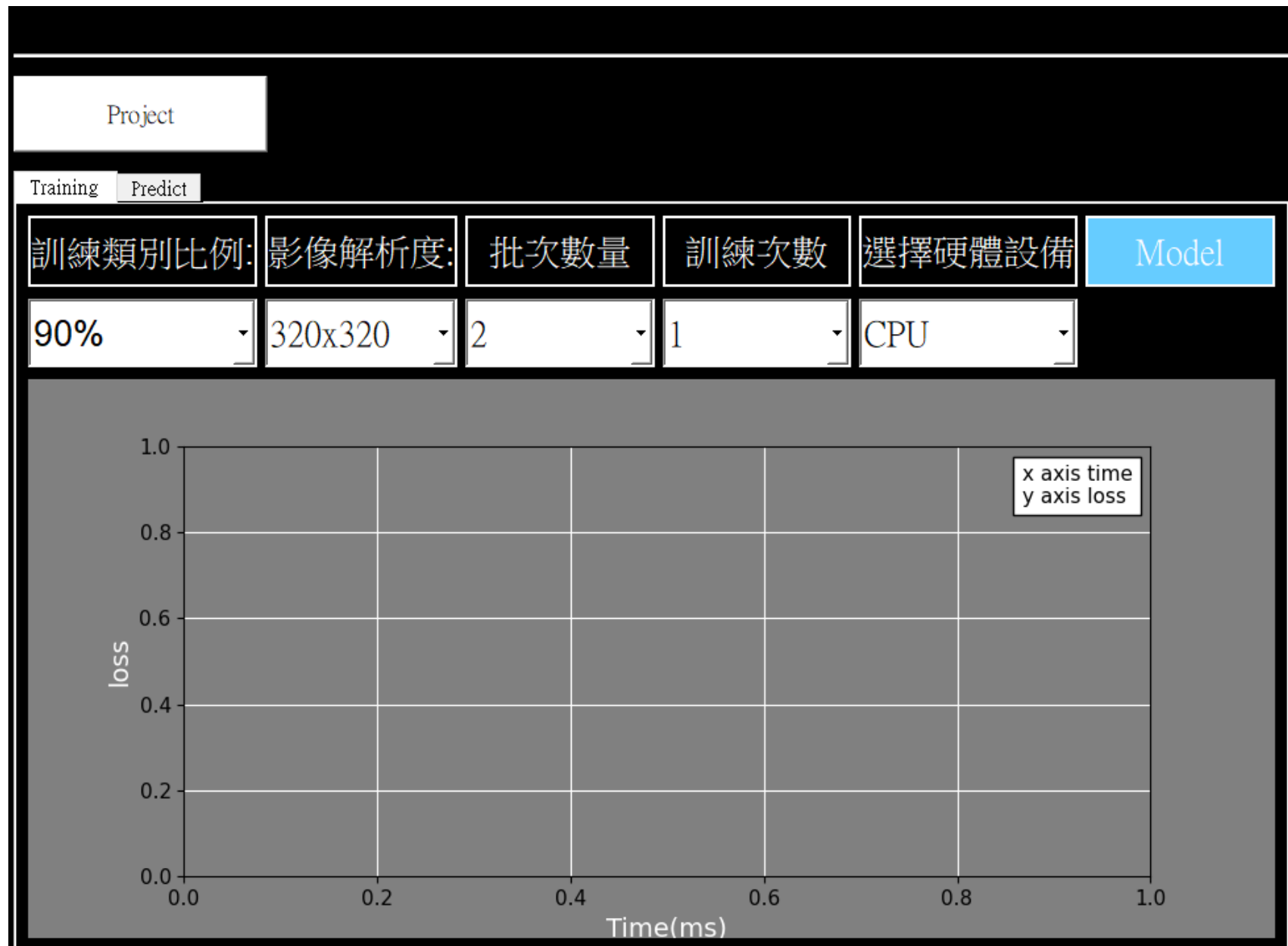
```
(env) D:\yolov7>pip install pyqt5  
Collecting pyqt5  
  Using cached PyQt5-5.15.9-cp37-abi3-win_amd64.whl (6.8 MB)  
Collecting PyQt5-Qt5>=5.15.2  
  Using cached PyQt5_Qt5-5.15.2-py3-none-win_amd64.whl (50.1 MB)  
Collecting PyQt5-sip<13,>=12.11  
  Using cached PyQt5_sip-12.12.1-cp39-cp39-win_amd64.whl (78 kB)  
Installing collected packages: PyQt5-Qt5, PyQt5-sip, pyqt5  
Successfully installed PyQt5-Qt5-5.15.2 PyQt5-sip-12.12.1 pyqt5-5.15.9  
WARNING: You are using pip version 22.0.4; however, version 23.1.2 is available.  
You should consider upgrading via the 'D:\env\Scripts\python.exe -m pip install --upgrade pip' command.  
(env) D:\yolov7>
```



黃色部分更新是pip版本，如果你的pip版本夠新可以不用理會

## Step4. 在完成前面步驟的安裝與設置，就可以開始執行主程式了

```
$ python main.py
```



開啟後的介面=>



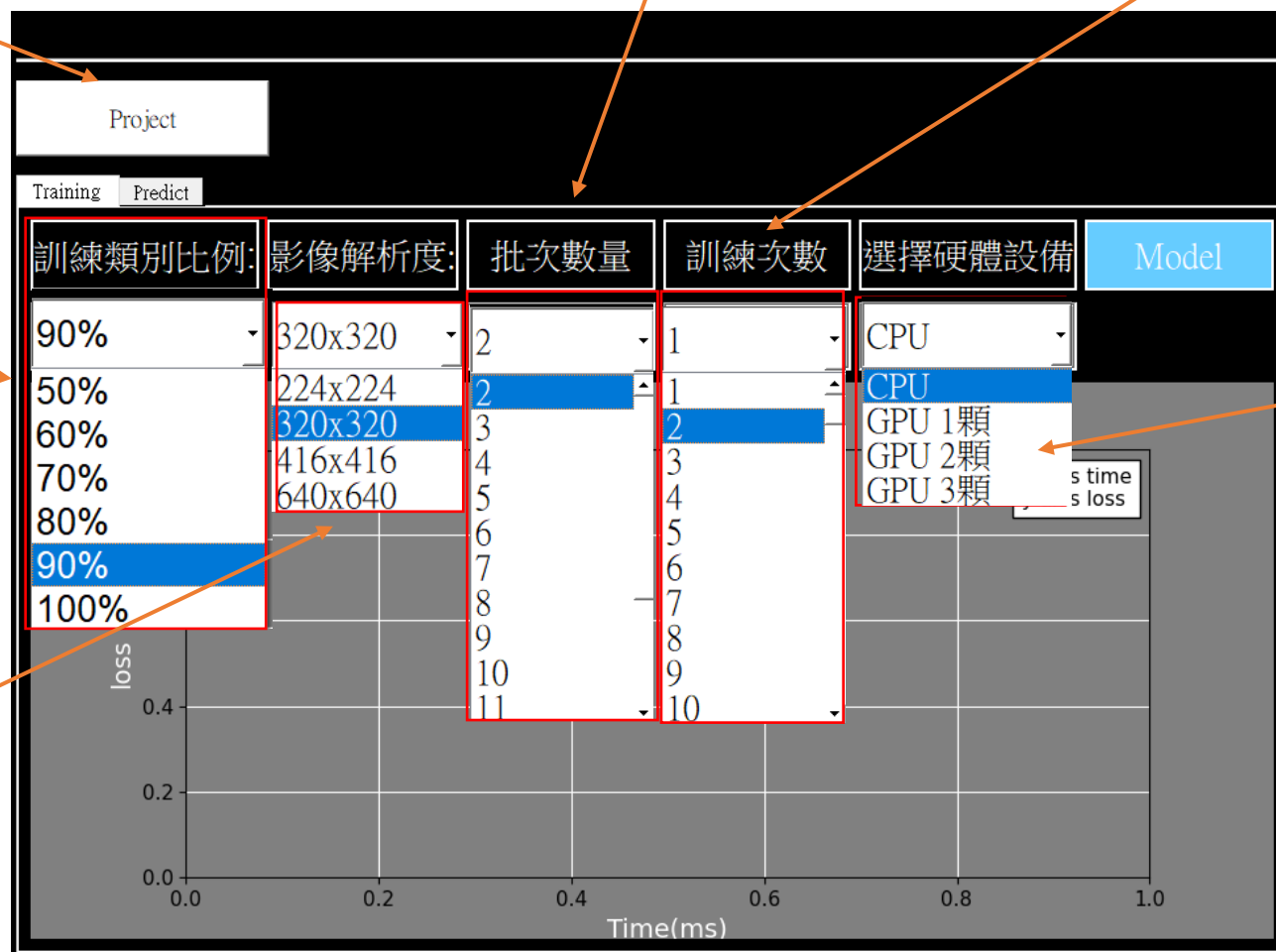
# Step5. 介面功能說明-Training

project按鈕可以選擇你的準備好的影像與label檔案的路徑資料夾

訓練數據集通常很大，無法一次將所有樣本放入模型進行訓練，因此我們將數據分為多個批次進行訓練。

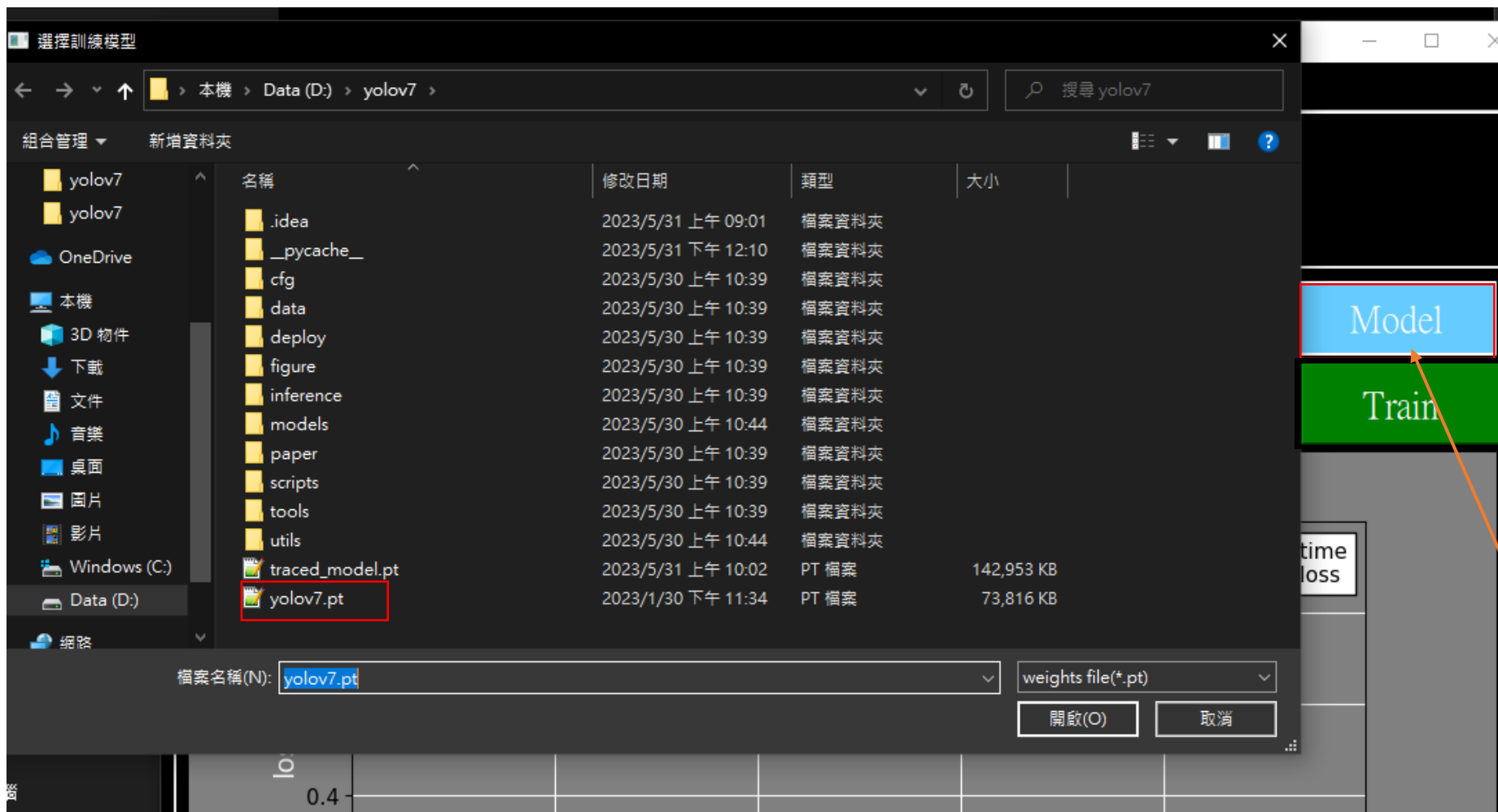
Epoch表示了訓練過程中數據集被使用了多少次。

訓練類別的比例: 會在你路徑資料夾中分類，將選擇的百分比分給訓練集，其餘的會分給驗證集



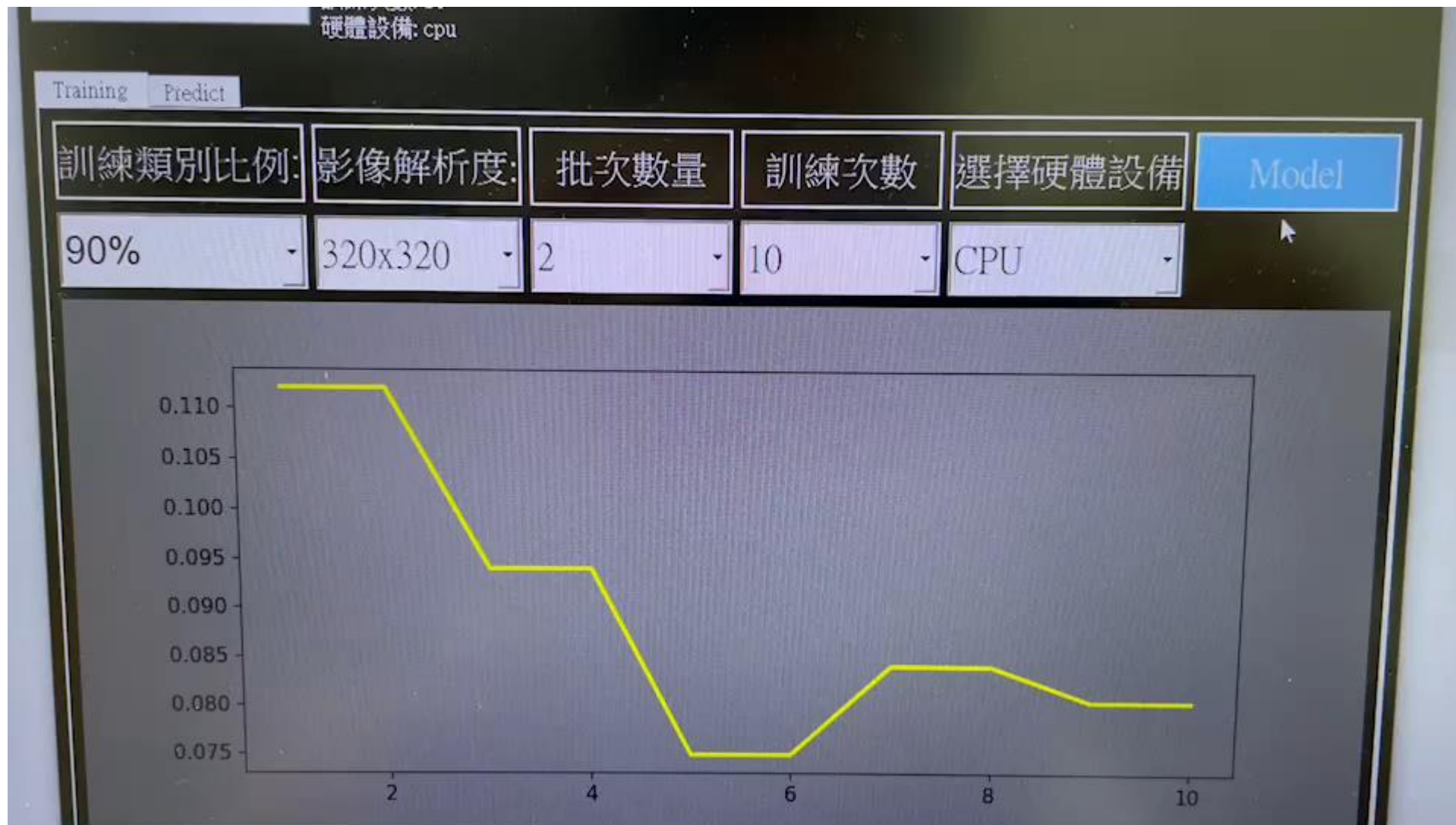
影響的解析度: 在訓練的過程，會將影像壓縮，避免記憶體太大

這邊只能選擇CPU，沒有使用conda的虛擬環境來執行，所以GPU無法被使用，(如果要選擇GPU，可以參閱我的另一個專案，有安裝說明 [https://github.com/kuan79223/SidePorject/blob/main/Yolov7\\_API/Yolov7%E5%9F%B7%E8%A1%8C%E6%93%8D%E4%BD%9C%E9%A0%86%E5%BA%8F.pdf](https://github.com/kuan79223/SidePorject/blob/main/Yolov7_API/Yolov7%E5%9F%B7%E8%A1%8C%E6%93%8D%E4%BD%9C%E9%A0%86%E5%BA%8F.pdf))

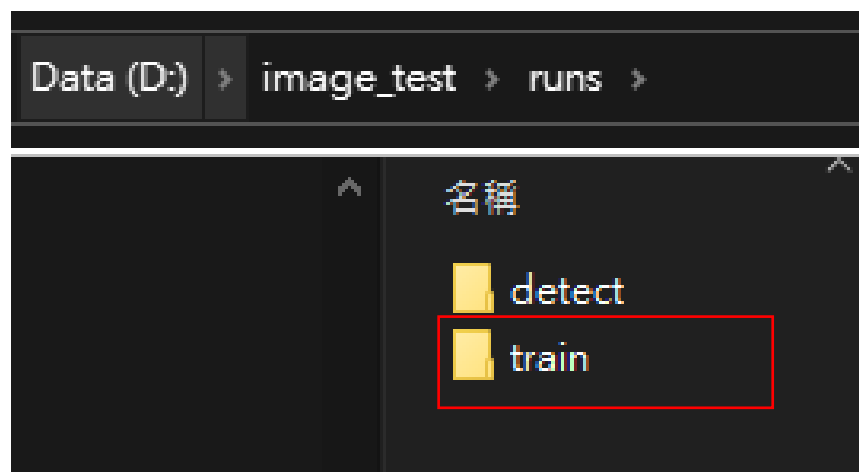
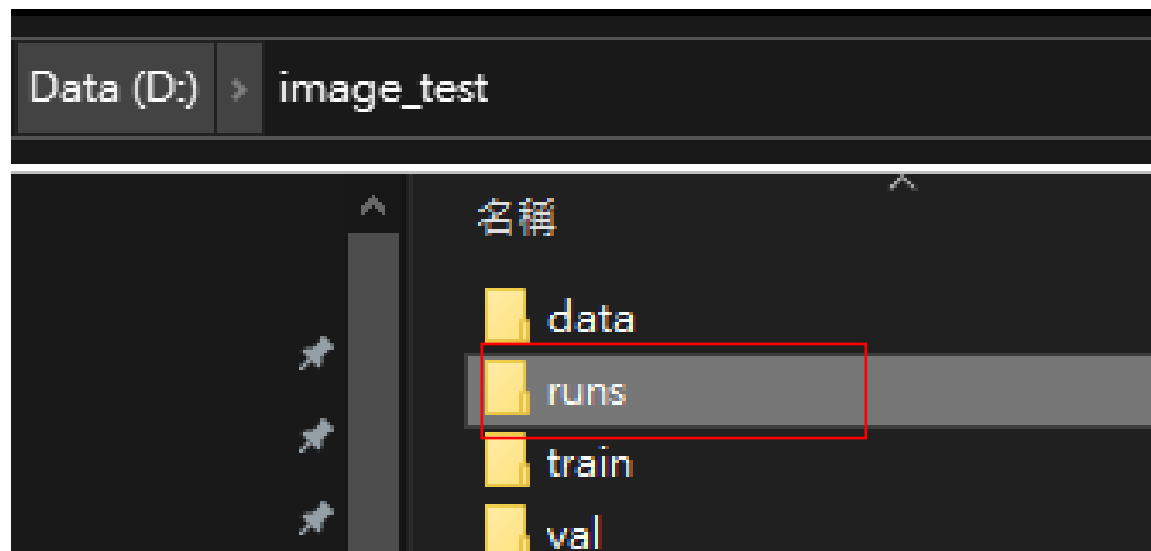


在參數都選擇完畢之後，就可以按model來選擇訓練的模型，步驟1中我們有下載 yolov7.pt 的模型，就可以拿來做訓練使用

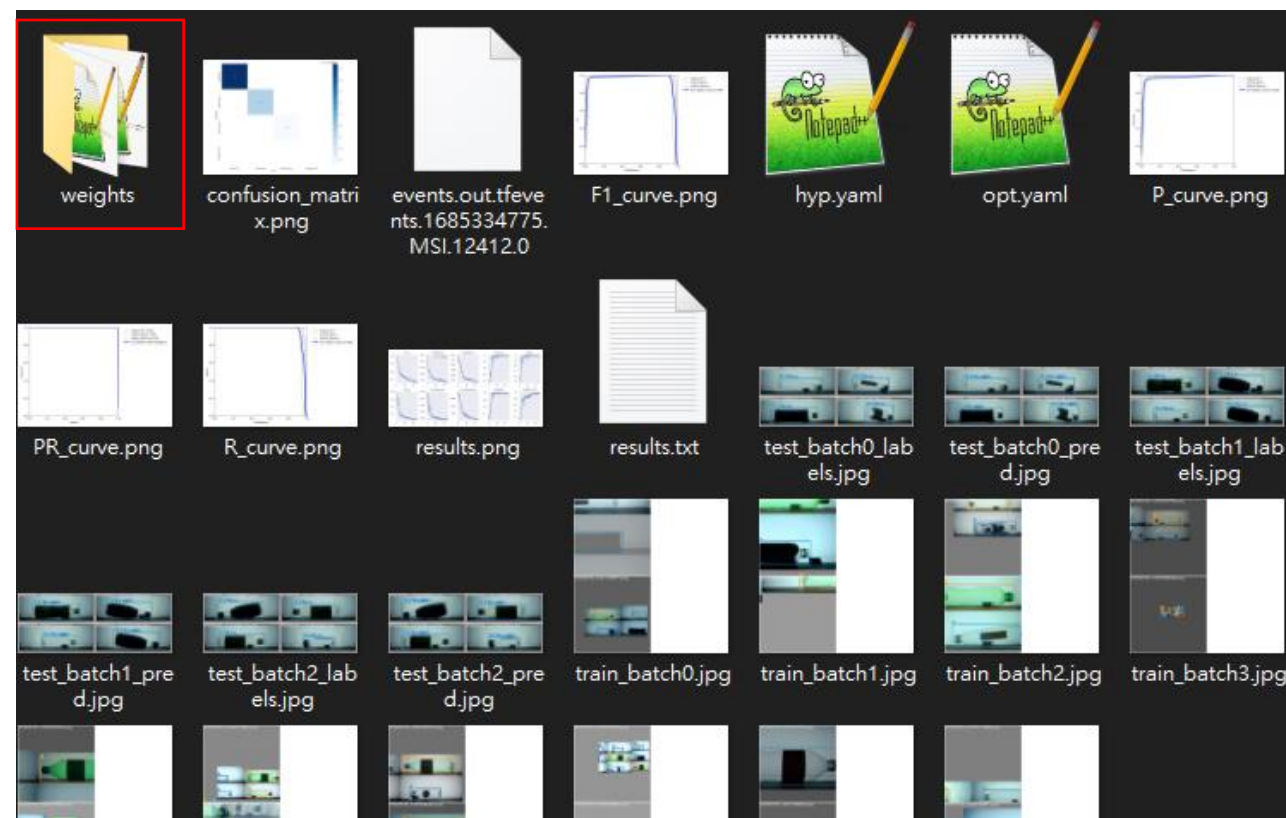
選擇之後，訓練按鈕會顯示出來，就可以點擊來做訓練了，訓練的過程，你可以即時的觀看訓練的分數，來評論訓練的好壞，也會在你專案中生成一份曲線圖的影像檔([figure.png](#))。Github有附上影片



訓練結束之後，專案路徑內會有一個 runs，在train 的資料夾內可以看到訓練後的一些參考資料，exp 資料夾為每一次訓練後產生



這個很重要，驗證影像的時候會用到



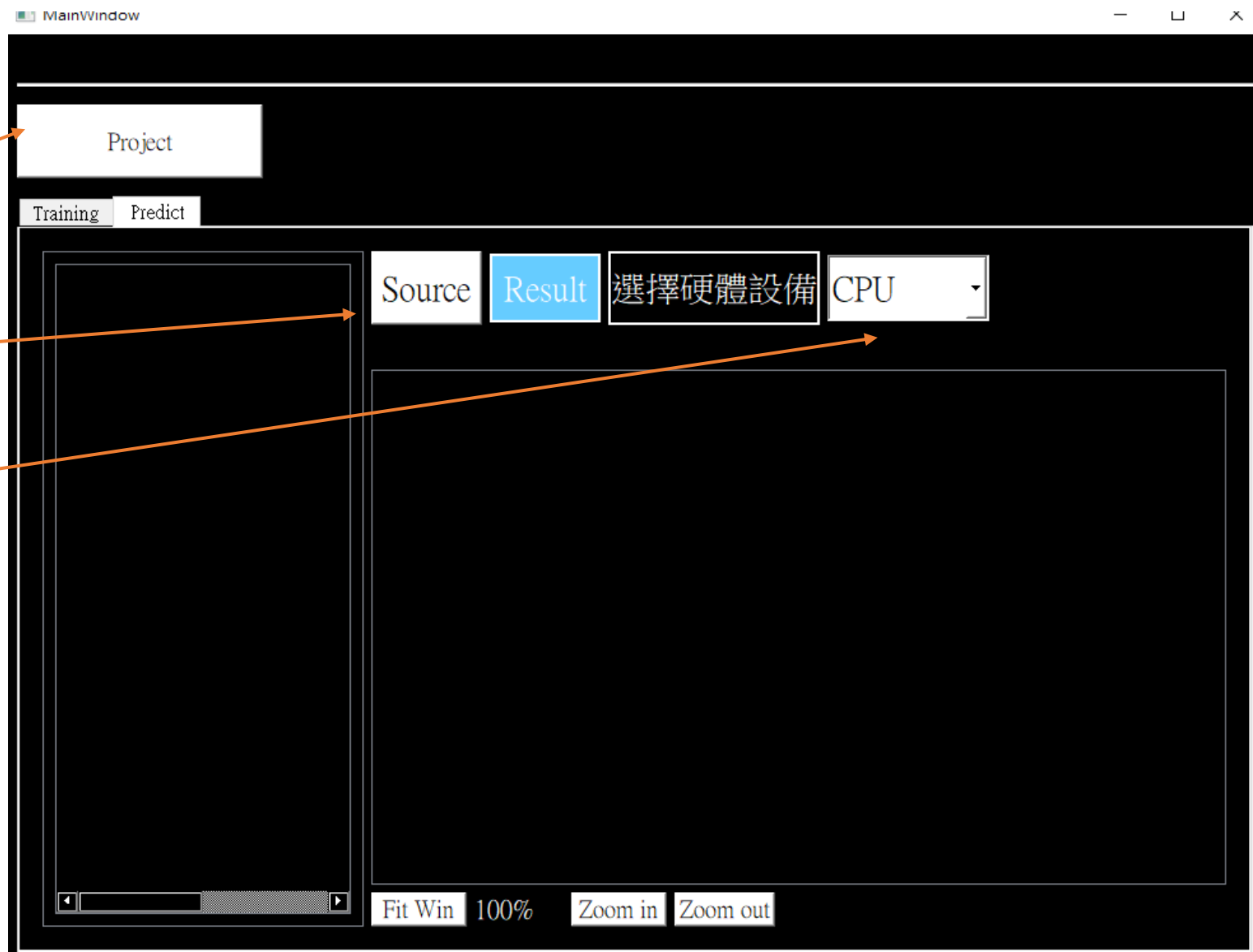
## Step5. 介面功能說明-Predict

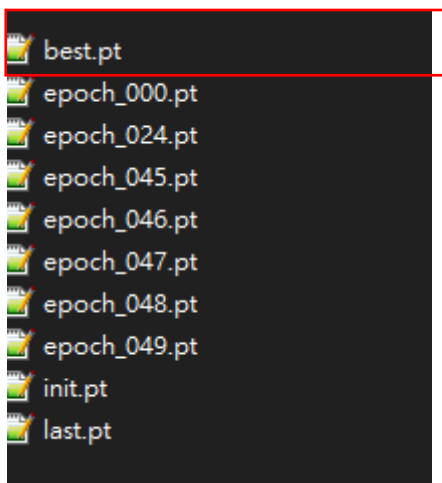
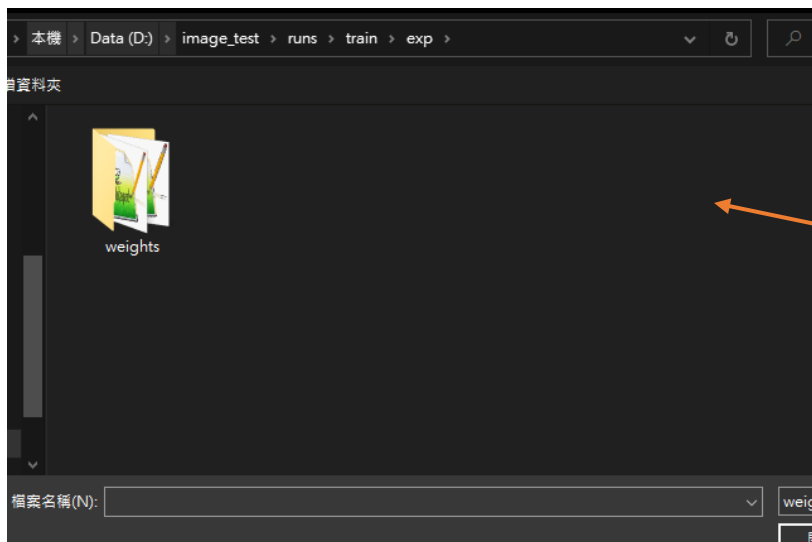
訓練之後就可以在predict 看到訓練的成果

project功能與train一樣

選擇你要預測影像的路徑

選CPU就可以了

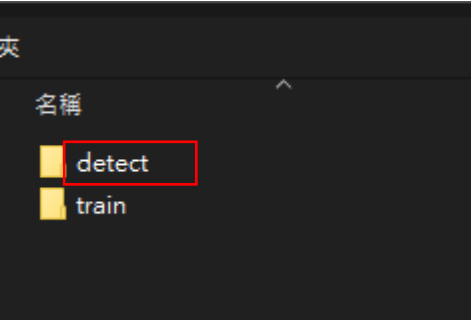




選擇在訓練後生成的  
runs/train 的 wieghts 有一個  
最佳的模型，Predict按鈕顯  
示出來後就可以做預測了



預測後點擊 **result** 按鈕，選擇預測後生成的 **detect/exp** 資料夾，載入後(影像很多可能需要等待一下)，可以在介面上直接看到物件偵測的結果





感謝你的觀看