

醫學影像專題 專題二 影像分割

任務：設計一個卷積神經網路進行影像分割。

1. 框架**僅限 PyTorch**並以 Python 實作。
2. 請記得安裝 CUDA 及 cuDNN。
3. 提供之影像資料請自行分成訓練/測試集，與分類較不同，分割的影像及 Ground truth 皆為影像，分別放在 data/imgs 及 /masks 中，相對應的檔案檔名會相同，僅路徑不同。訓練時會讀入 data/ 中的所有資料，大家可以自行取出幾組照片留做測試。
4. 範例程式使用 Unet，可使用其他網路架構或自行設計。
5. 繳交時請將以下放在一個資料夾中壓縮並填上組別：
 - a. 訓練及測試程式
 - b. 測試圖片之**結果圖**與**預測機率圖**(後續說明，預計 10 張組)
 - c. 口頭報告 power point
 - d. 書面報告
6. 黑箱測試方式為由我們提供影像(無 Ground truth)，大家測試後再回傳結果圖與機率圖，我們會計算 AUC，以此做評估依據。

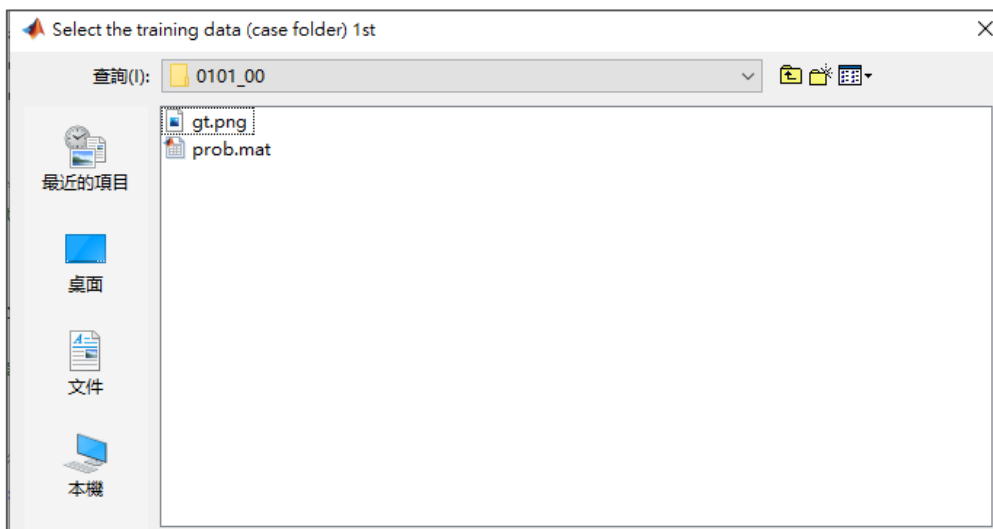
建議使用提供的 predict.py 進行測試，以避免輸出的機率圖格式與我們的不同而無法測試，會對您的成績造成影響。

實驗規定及資料說明：

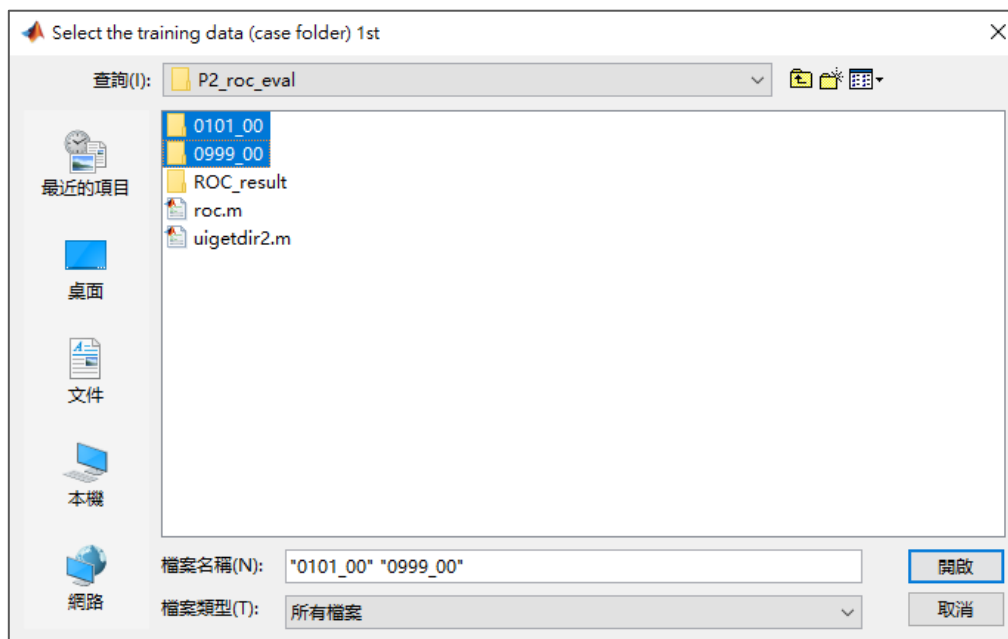
1. 實驗資料共 40 組(一組 5 張)，同一組的前綴檔名一樣(如 0101_02、0101_03)，後綴則統一有 ”_1” 到 ”_5” ，其代表意義為 5 個時間點拍攝之照片，因此 Ground truth 都一樣(已帮大家處理好)。
訓練時設定各組可選擇：
 - a. 只挑選某一個時間點進行訓練；
 - b. 5 個時間點個別訓練(意即你會有 5 個模型 for 5 個時間點)
 - c. 5 個時間點一起訓練。
2. 需進行 5 次的交叉驗證。
3. 效能評估：於每組圖片中(5 張)選一張最佳結果計算 AUC(範例為 matlab 程式)，共 40 組所以應要有 40 張照片的 AUC 結果。

範例程式說明：

1. 使用 train.py 訓練模型，網路架構定義在/unet 中。
 2. 產生模型後再在 predict.py 中修改訓練好的模型名稱並測試，會產生預測結果和機率圖，原始程式一次測試一張圖片，若要批量測試可以自行撰寫腳本。
 3. ROC 程式於/ROC 中的 roc.m，計算時請將兩張圖(預測機率圖、/masks 中相對應 ground truth)放在同一個資料夾中(圖一，路徑可以取為該影像檔名，如 0101_02_1)，接著選取要計算的資料夾(可批量取，如圖二)。
- 各組照片的資料夾已分好在/ROC_input 中。



圖一



圖二

4. 執行後結果會存在/ROC_result 並在 Console 印出各張圖的 AUC，如圖三。

```

Total cases : 2
0101_00
AUC_ROC =

    0.6187

-----
0999_00
AUC_ROC =

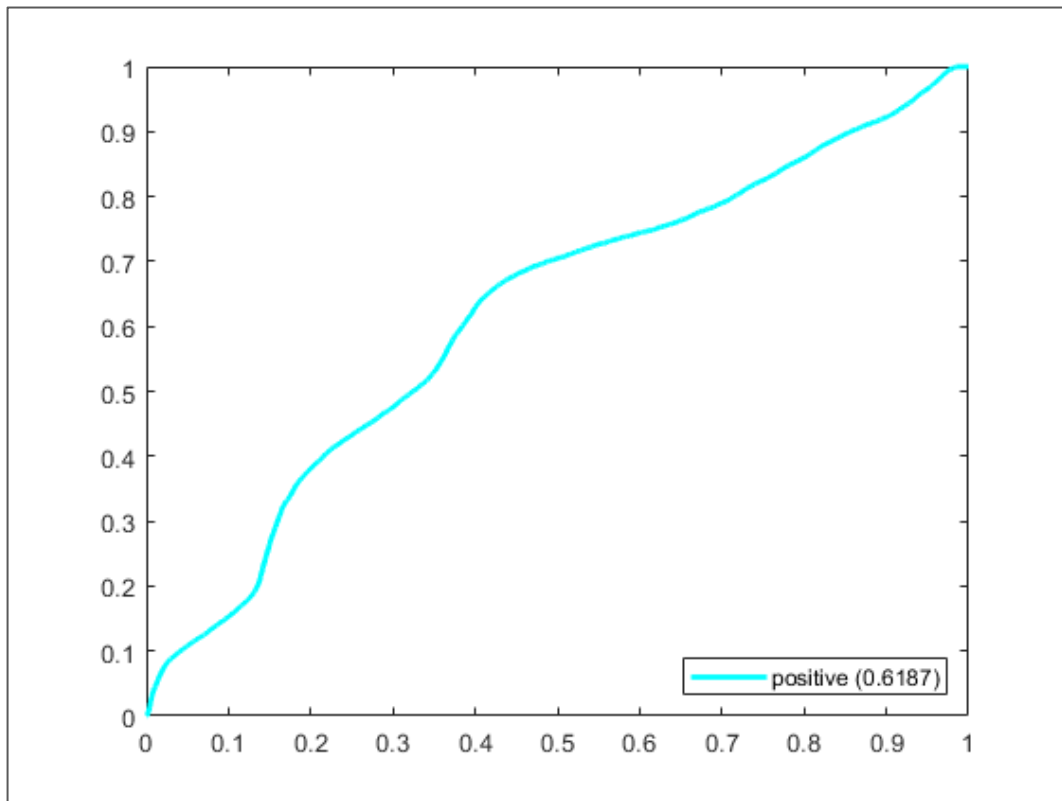
    0.6187

-----

```

圖三

5. ROC 曲線圖不會每張都儲存，大家可以自行修改程式，若不想修改程式則在選取時只選擇一個資料夾，並把印出來的圖儲存下來，ROC 圖如圖四。



圖四

6. 關於任務第 5 點 “b. 測試圖片之結果圖與預測機率圖”，我們在之後會公佈 10 組影像(共 50 張，無 GT)，大家使用自己的模型測試完後應會產生結果圖以及機率圖，請大家繳回這 10 組的**結果圖及機率圖**(每組照片選一張進行測試並回傳該張的結果圖及機率圖即可，共 **10 張結果圖及 10 張機率圖**)，我們會帮大家畫出 ROC 以及計算 AUC，並以此作為黑箱測試之結果。